Theoretical Computer Science 82 (1991) 177-214
Elsevier

Fundamental Study

Infinite hypergraphs I. Basic properties

Michel Bauderon*

Laboratoire Bordelais de Recherche en Informatique, Unité associée au C.N.R.S. no 1304, Université de Bordeaux I, Département Informatique, I.U.T. 'A', F-33405 Talence Cedex, France

Communicated by M. Nivat
Received February 1989
Revised February 1990

Abstract

Bauderon, M., Infinite hypergraphs I. Basic properties (*Fundamental Study*), Theoretical Computer Science 82 (1991) 177-214.

Basic properties of the category of infinite directed hyperedge-labelled hypergraphs are studied. An algebraic structure is given which enables us to describe such hypergraphs by means of infinite expressions. It is then shown that two expressions define the same hypergraph if and only if they are congruent with respect to some rewriting system. These results will be used in the second part of this paper to solve systems of recursive equations on hypergraphs and characterize their solutions.

Résumé

On étudie les propriétés fondamentales des hypergraphes infinis à hyperarêtes étiquetées. Une structure algébrique est fournie qui permet de décrire de tels graphes à l'aide d'expressions infinies. On prouve que deux telles expressions définissent le même graphe si et seulement si elles sont congruentes modulo un certain système de réécriture. Ces résultats seront utilisés dans la seconde partie de cet article pour étudier certains systèmes d'équations récursives sur les hypergraphes et pour caractériser leurs solutions.

1. Introduction

Infinite trees have proved to be a very essential tool for the study of the algebraic semantics of program schemes, where they usually arise as (components of) solutions

* This work has been partially supported by the C.N.R.S. PRC "Mathématiques et Informatique" and by the ESPRIT BRA 3299 "Computing with graphs transformations".

of systems of equations on trees. Numerous works have been devoted to them (a comprehensive treatment of the theory of infinite trees may be found in [9]).

However, a program scheme should be more naturally represented as a graph (in fact a directed graph), in order to take into account the natural sharing of some subterms, and the real objects of interest in semantics should then be the infinite graphs arising as solutions of certain systems of recursive equations on graphs.

As might be expected, such systems of equations generate some very "regular" graphs (in a sense that we shall make more precise later on). Similar graphs have already appeared in various works such as [5] (the so-called pattern graphs), [25] (map sequences) or [29] (context-free graphs), which is a further reason for the interest we have in their study.

Intuitively, solving such systems of equations on graphs appears to be quite straightforward, through the method of iterated substitution which is commonly used on trees. This method relies on the algebraic properties of the sets of trees and infinite trees (practically, upon the existence of a notion of substitution), and its validity when used for the resolution of equations on trees is insured by the availability of a metric or order structure which turns the set of infinite trees either into a complete metric space or an ω-complete many-sorted algebra. For both structures, there exists a well known fixpoint theorem which one can readily apply (see [9]).

The first requirement is easily fulfilled since an algebraic framework has been introduced in [4] to describe finite oriented hyperedge-labelled hypergraphs with a sequence of distinguished vertices called sources, which can be readily extended to countable hypergraphs (a fairly similar model for finite graphs has been introduced independently by Habel and Kreowski (cf. [18], or [17] for a detailed study)). Informally, an example of such a graph might look something like (a precise description will be given in Section 3) that shown in Fig. 1 (the vertices and

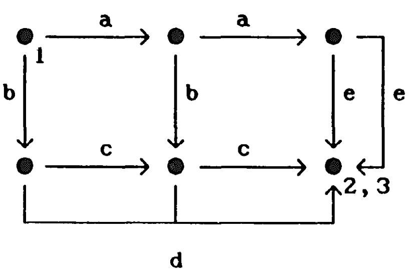


Fig. 1.

hyperedges are not named on this figure, the hyperedges are labelled, most hyperedges are binary, only the one labelled by d is ternary, the integers 1, 2 and 3 designate the sources).

Unfortunately, the second requirement is much harder to satisfy, i.e. no classical notion of limit is easily available on graphs (either metric or order-theoretic). We did not succeed in defining one.

In our own setting, the difficulty in defining an order relation is (intuitively) the following: to be natural, such an order relation should respect graph inclusion. Namely, if G is a subgraph of G' in some intuitive sense, one expects G to be smaller than G' and a reasonable definition of such an order should be something like:

$$G \leqslant G' \Leftrightarrow G \text{ is a subgraph of } G'.$$

Reflexivity and transitivity are clearly satisfied. Now, to be an order relation, this relation should be asymmetric as well, i.e., one should have

$$G \leqslant G', G' \leqslant G \Rightarrow G \simeq G'.$$

This property is satisfied in the finite case, i.e. one can check that the relation we have just defined is actually an order relation on the set of finite graphs but this is not necessarily true in the case of infinite graphs as will be shown by the following counter-example (Courcelle).

Let G and G' be the two infinite (hyper-)graphs with one single source shown in Fig. 2. It is fairly clear in this example that $G \leqslant G'$ and that $G' \leqslant G$ but that

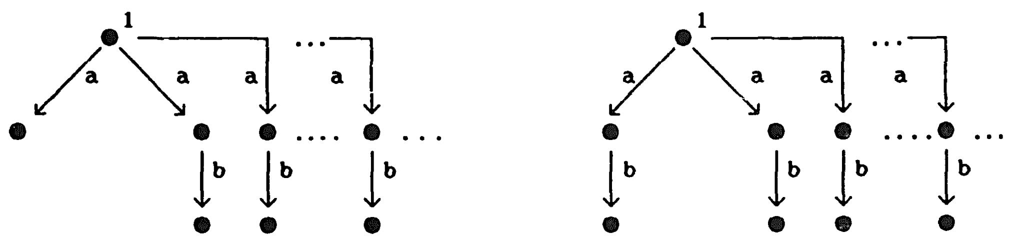


Fig. 2.

these two graphs are not isomorphic. It must be noted that this counter-example involves two graphs with non-finite degree. In fact, one has the following result [5]. If we restrict ourselves to some finite graphs, the relation $\leqslant$ defined above is an order relation (we shall not be more precise about "some" since it is not really relevant in our context).

However, since graphs with non-locally finite degree occur very naturally when solving regular equations on hypergraphs (some examples will appear in the second part [3]) we cannot restrict ourselves to the hypotheses of the previous result and have to use different tools.

This kind of difficulty is well known since it has been met in a wide range of problems (study of infinite words [8] where an ad hoc approach is used, theory of domains [23, 30] abstract data types [24] etc.) and a method has been developed by generalizing the usual order-theoretic concepts and results of algebraic cpos to the more general setting of (algebroidal) ω-categories (the topological approach might be generalized as well, defining some kind of Scott topology on algebroidal categories; to our knowledge this has not been done yet).

Let us simply add that, in a number of situations where a cpo structure is available, the order relation is defined in terms of inclusion (or substitution) and that one has to alternatively use the order structure or its actual implementation. We therefore advocate the use of categorical concepts even in those cases, since they provide a very simple and elegant framework which unifies both approaches, although they are not yet familiar enough to most computer scientists.

With no further justification, we shall first develop the basic framework which is necessary, including some elements of category theory, basic definitions on sourced hypergraphs and hypergraph expressions before going into the resolution of equations.

The main results of this work are:

- it is correct to use the method of iterated substitution to solve systems of regular equations on hypergraphs [3], and their solutions can be completely described;
- hypergraphs may be described by hypergraphs expressions, in such a way that
  - (i) two expressions denote the same hypergraph if and only if they are congruent with respect to some rewriting system (Section 5),
  - (ii) a system of regular equations may be solved either directly or through a corresponding system of regular equations on expressions, yielding the same solutions [3];
- equational hypergraphs (i.e. components of the initial solution of some system of regular equations) coincide with the context-free graphs in [29] (provided some slight restrictions are made [3]);
- most of the constructions given in the paper are effective (for some reasonable definition of effectivity which shall be made precise in Section 2).

The first part of this paper is organised as follows:

- Section 2 is devoted to a number of definitions and results from category theory,
- Section 3 presents the basic definitions and results on sourced hypergraphs,
- Section 4 develops the notion of hypergraph expressions and shows how it is related to sourced hypergraphs,
- Section 5 contains the proof of the fact that two expressions denote the same hypergraph if and only if they are congruent with respect to a certain rewriting system.

The study of systems or recursive equations will be done in the second part of this paper [3].

Early results of this work have been presented in a more informal way at the 2nd Conference on Automata, Languages and Programming systems, held in Salgótarján (Hungary), May 1988, and at the 14th Workshop on Graph-Theoretic Concepts in Computer Science, held in Amsterdam, June 1988 [2]. A tutorial survey of research about finite and infinite hypergraphs may be found in [12].

Notations. As usual we let $\mathbb{N}$ denote the set of non-negative integers and for any $n \in \mathbb{N}$, we let $[n] = \{p \in \mathbb{N} / 1 \leqslant p \leqslant n\}$, with $[0] = \emptyset$.

For any set E, $\# E$ denotes the cardinality of E and $\mathcal{P}(E)$ denotes the set of all its subsets. Equiv$[n]$ denotes the set of equivalence relations in $[n]$. Most often, such an equivalence relation will be specified by some generating subset (in $[n]^2$).

If $A = \bigcup_{n\in\mathbb{N}} A_n$ is a ranked alphabet, with rank function $\tau : A \to \mathbb{N}$, the free monoid generated by this alphabet will be denoted by A^*.

2. Some elements of category theory

This rather lengthy section is intended to bring altogether all the notions that we shall need later on and make this paper almost self-contained (without indulging oneself in too intricate considerations). It may be skipped and referred to only when necessary.

Most of the notions and results are standard but some might be rather difficult to find in the available textbooks (such as [28] or [26]; see however [27] for some interesting discussions of fixed points). Most proofs are not included.

2.1. Categories

A *category* $\mathbf{C}$ is a pair $(\mathrm{Ob}_{\mathbf{C}}, \mathrm{Hom}_{\mathbf{C}})$ where $\mathrm{Ob}_{\mathbf{C}}$ is the class of objects and for any two objects A, B of $\mathrm{Ob}_{\mathbf{C}}$, $\mathrm{Hom}_{\mathbf{C}}(A, B)$ is the set of arrows from A to B, subject to the following conditions:

- for any object A, there exists a unique identity arrow $\mathbf{1}_A$,
- for any three objects, A, B, C, the composition of arrows is defined:

$$\mathrm{Hom}_{\mathbf{C}}(A, B) \times \mathrm{Hom}_{\mathbf{C}}(B, C) \to \mathrm{Hom}_{\mathbf{C}}(A, C)$$

The most usual category is of course **Set**, the category whose objects are sets and whose arrows are the usual mapping between sets. We shall let $\mathbf{1}$ denote the one object one arrow category.

Another category of frequent use in our context is the category ω, associated with the first infinite ordinal ω. It is usually represented in the following way (omitting the identities and the composed arrows), which describes the natural order on $\mathbb{N}$:

$$0 \to 1 \to 2 \to 3 \to \cdots \to n \to \cdots.$$

It is well known, that in a similar way, a category may be associated with every preordered set: its objects are the elements of the preorder and there is an arrow $a \to b$ if and only if $a \leq b$.

It is quite frequent in theoretical computer science, that the object of concern is a set of structured objects endowed with a (pre)order. Usually, this (pre)order is defined in terms of some sort of inclusion morphisms (this is the case for trees for instance). In those cases, these morphisms are exactly the arrows of the category canonically associated with the (pre)order, and the proofs make alternatively use of the (pre)order or of what we would like to call its implementation. This simply makes them more awkward when categorical concepts would render them more elegant. This fact will be used in Section 4.4.

An arrow $f: a \to b$ in $\mathbf{C}$ is a *mono(morphism)* if it is left cancellable, an *epi(morphism)* if it is right cancellable, an *isomorphism* if there is an arrow $f': b \to a$ such that $ff' = 1_b$ and $f'f = 1_a$. We shall sometimes write $A \simeq B$ to mean that A and B are isomorphic. Note that isomorphic implies mono and epi while the converse is not necessarily true, though it will hold in our particular case.

With a category $\mathbf{C}$, one associates a new category called the *category of arrows* of $\mathbf{C}$, whose objects are the arrows of $\mathbf{C}$ and whose arrows from an object $f: a \to b$ to an object $f': a' \to b'$ are pairs of arrows $(g: a \to a', h: b \to b')$ such that $f' \circ g = h \circ f$, composition of arrows being defined componentwise.

An object $\perp$ in a category $\mathbf{C}$ is an *initial object* if for any object C of $\mathbf{C}$, there is a unique arrow $\perp_C: \perp \to C$. Clearly, an initial object in a category is unique up to an isomorphism. A category is an *initial category* if it has an initial object.

If $\mathbf{C}$ and $\mathbf{D}$ are two categories, a product category $\mathbf{C} \times \mathbf{D}$ is defined which has as objects and arrows, the pairs of objects and arrows of $\mathbf{C}$ and $\mathbf{D}$. If $\mathbf{C}$ and $\mathbf{D}$ are initial categories, $\mathbf{C} \times \mathbf{D}$ is also initial. This is easily extended to an arbitrary finite number of categories.

2.2. *Functors and colimits*

Let $\mathbf{C}$ and $\mathbf{D}$ be two categories. A *(covariant) functor* $\mathscr{F}$ from $\mathbf{C}$ to $\mathbf{D}$ assigns to each object C of $\mathbf{C}$ an object $\mathscr{F}C$ of $\mathbf{D}$ and to each arrow $f: C \to D$, an arrow $\mathscr{F}f: \mathscr{F}C \to \mathscr{F}D$ such that $\mathscr{F}\mathbf{1}_A = \mathbf{1}_{\mathscr{F}A}$ and $\mathscr{F}(f \circ g) = \mathscr{F}f \circ \mathscr{F}g$.

Let $\mathbf{J}$ be a category. A $\mathbf{J}$-diagram in the category $\mathbf{C}$ is a functor $\mathscr{D}$ from $\mathbf{J}$ to $\mathbf{C}$. In particular, an *ω-diagram* in $\mathbf{C}$ is a functor from ω into $\mathbf{C}$, or, in other words, a countable sequence of objects and arrows which we shall draw as

$$C_0 \to C_1 \to C_2 \to C_3 \to \cdots \to C_n \to \cdots.$$

A *cocone* over a $\mathbf{J}$-diagram $\mathscr{D}$ in $\mathbf{C}$ is an object K of $\mathbf{C}$ (the vertex of the cocone) and a family of arrows $KJ: \mathscr{D}J \to K$ for any J in $\mathbf{J}$ such that, for any arrow $u: I \to J$ in $\mathbf{J}$ one has $KI = \mathscr{D}u \circ KJ$.

A *colimit* for $\mathscr{D}$ is a cocone over $\mathscr{D}$ with vertex C such that, for any cocone with vertex K over $\mathscr{D}$, there is a unique arrow $U: C \to K$ which satisfies $KJ = U \circ CJ$ for every object J in $\mathbf{J}$. The colimit of a $\mathbf{J}$-diagram is unique up to an isomorphism. The *ω-limit* of an ω-diagram in $\mathbf{C}$ is the colimit of this diagram.

The *coproduct* of a family of objects $C_{I \in \mathbf{I}}$ in $\mathbf{C}$ indexed by a category $\mathbf{I}$ is the colimit of the diagram $\mathscr{G}: I \to C_I$, with no arrow component.

Let $\mathbf{2}$ be the category $1 \rightrightarrows 2$ (two distinct objects, two distinct arrows and identities). A colimit of a $\mathbf{2}$-diagram is called a *coequalizer*.

A *pushout* is a colimit for a diagram over the category $\mathbf{W}$ with three objects and only two arrows different from the identities

$$\begin{array}{ccc} a & \longrightarrow & b \\ \downarrow & & \\ c & & \end{array}$$

2.3. ω-completeness and ω-continuity

A category is called *cocomplete* if it has all colimits. The standard technique to prove this property is to use the following theorem.

2.3.1. Theorem (McLane [28]). *A category has colimits if and only if it has coproducts and coequalizers.*

However this usually leads to fairly long and awkward proofs. In many cases, the cocompleteness can be obtained thanks to Theorem 2.6.2. (whose proof of course, makes use of Theorem 2.3.1).

2.3.2. Proposition. *If* $\mathbf{C}$ *and* $\mathbf{D}$ *are cocomplete categories, the product category* $\mathbf{C}\times\mathbf{D}$ *is cocomplete. If* $\mathbf{C}$ *and* $\mathbf{D}$ *are initial with initial objects* $\perp_{\mathbf{C}}$ *and* $\perp_{\mathbf{D}}$, *the product category is initial with initial object the pair* $(\perp_{\mathbf{C}}, \perp_{\mathbf{D}})$.

A category is an *ω-category* if it has all ω-limits, i.e. colimits of all ω-diagrams. Clearly, a cocomplete category is an ω-category and has coproducts and pushouts as well.

We say that a subcategory $\mathbf{C}$ of a category $\mathbf{D}$ is *codense* (resp. *ω-dense*) in $\mathbf{D}$ if any object D of $\mathbf{D}$ is the colimit (resp. ω-limit) of a diagram (resp. ω-diagram) in $\mathbf{C}$ (in other words, ω-density means that any object in $\mathbf{D}$ may be approximated by a countable sequence of objects in $\mathbf{C}$).

A functor $\mathscr{F}$ from $\mathbf{C}$ to $\mathbf{D}$ is *ω-continuous* if it preserves the ω-limits and the related ω-limiting cocones (ω-cones). A bifunctor $\mathscr{F}$ from $\mathbf{B}\times\mathbf{C}$ into $\mathbf{D}$ is ω-continuous if its two components are. Clearly, ω-continuity is preserved by composition of functors (or bifunctors). All through this paragraph, ω may be replaced with co for a greater generality.

2.4. Definition. Let $\mathscr{F}$ be an endofunctor of a category $\mathbf{C}$. A *fixpoint* of $\mathscr{F}$ is a pair (C, u) where C is an object of $\mathbf{C}$ and $u : \mathscr{F}C \to C$ is an isomorphism. Such a fixpoint is sometimes called an $\mathscr{F}$-algebra.

A detailed study of the following theorem will be found in Adamek and Koubek [1] where it is dated back to Lambek [21]. Related works and results are those of Lehmann [23], Lehmann and Smyth [24] and more recently, Lambek [22].

2.5. Theorem (existence of a fixpoint). *Let* $\mathbf{C}$ *be an initial ω-category and* $\mathscr{F}$ *be an ω-continuous endofunctor of* $\mathbf{C}$. *Then the fixpoints of* $\mathscr{F}$ *form an initial category, whose initial object is the colimit of the following ω-diagram*:

$$\perp \to \mathscr{F}\perp \to \mathscr{F}^2\perp \to \mathscr{F}^3\perp \to \cdots \to \mathscr{F}^n\perp \to \cdots.$$

Proof. The proof of this theorem is fairly standard but we give it since it is constructive and a good understanding of it will enlighten the rest of this paper.

As usual, the construction of a fixpoint is made by successive approximations of the solution. Indeed, from any object C such that there is an arrow $u: C \to \mathcal{F}C$ in $\mathbf{C}$ we can generate an ω-diagram

$$C \xrightarrow{u} \mathcal{F}C \xrightarrow{\mathcal{F}u} \mathcal{F}^2C \xrightarrow{\mathcal{F}^2u} \mathcal{F}^3C \xrightarrow{\mathcal{F}^3u} \cdots \to \mathcal{F}^nC \xrightarrow{\mathcal{F}^nu} \cdots.$$

Since $\mathbf{C}$ is an ω-category, this ω-diagram has an ω-limit Ξ corresponding to the inner colimiting cocone in the commutative diagram shown in Fig. 3. Now since $\mathcal{F}$

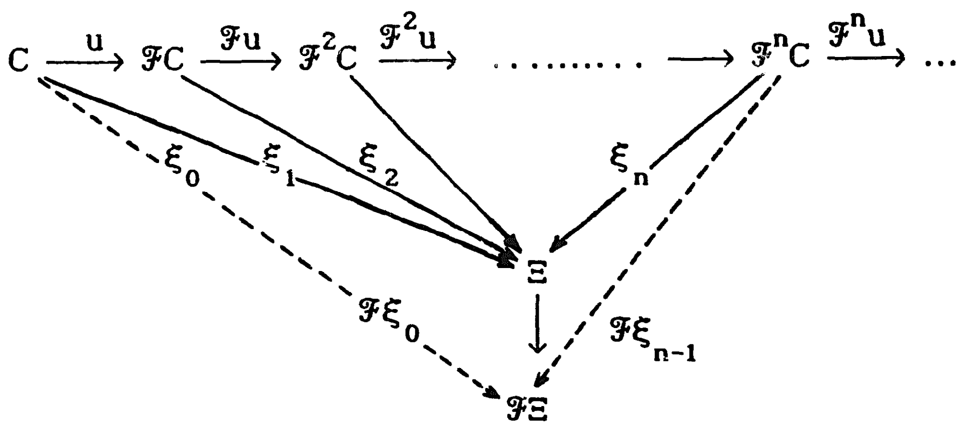


Fig. 3.

is an ω-continuous functor, when applied to the inner cocone, it generates the outer cocone (dotted lines). It follows that these two cocones are colimiting for the following ω-diagram:

$$\mathcal{F}C \xrightarrow{\mathcal{F}u} \mathcal{F}^2C \xrightarrow{\mathcal{F}^2u} \mathcal{F}^3C \xrightarrow{\mathcal{F}^3u} \cdots \longrightarrow \mathcal{F}^nC \xrightarrow{\mathcal{F}^nu} \cdots,$$

hence that the arrow $\Xi \to \mathcal{F}\Xi$ is an isomorphism.

Of course, the previous construction applies to the initial object $\perp_C$ of $\mathbf{C}$, since (by initiality of $\perp_C$) there always exists a unique arrow $\perp_C \to \mathcal{F}\perp_C$. Hence, the initial object in a category generates in all cases a fixpoint $\Omega_{\mathcal{F}}$ of the functor.

It is then easily checked that $\Omega_{\mathcal{F}}$ is initial in the category of fixpoints of $\mathcal{F}$, or, in other words, that for any other fixpoint Ξ of $\mathcal{F}$ there exists a unique arrow $\Omega_\Xi : \Omega \to \Xi$ such that the following diagram is commutative:

$$\begin{tikzcd} \Omega_{\mathcal{F}} \arrow[r, "\Omega_\Xi"] \arrow[d] & \Xi \arrow[d, "\mathrm{I}"] \\ \Omega_{\mathcal{F}} \arrow[r, "\mathcal{F}\Omega_\Xi"] & \mathcal{F}\Xi \end{tikzcd}$$

Indeed, since there exists a unique arrow $\perp_\Xi : \perp_C \to \Xi$, iterated application of the functor $\mathcal{F}$ generates a cocone $\mathcal{F}^i\perp_\Xi : \mathcal{F}^i\perp_C \to \mathcal{F}\Xi \simeq \Xi$. The existence of a unique arrow Ω_Ξ is ensured by the definition of a colimit. This concludes the proof. □

2.6. Comma categories

2.6.1. Definition. Let $\mathcal{F}: B \to A$ and $\mathcal{G}: C \to A$ be two functors. The comma category $(\mathcal{F}\downarrow\mathcal{G})$ has:

- as objects, the triples (b, c, f) where b is in B, c in C and $f: \mathcal{F}b \to \mathcal{G}c$,
- as arrows from (b, c, f) to (b', c', f') the pairs of arrows $(k: b \to b', h: c \to c')$ such that $f' \circ \mathcal{F}k = \mathcal{G}h \circ f$, i.e. such that the following diagram commutes:

$$\begin{tikzcd} \mathcal{F}b \arrow[r, "f"] \arrow[d, "\mathcal{F}k"'] & \mathcal{G}c \arrow[d, "\mathcal{G}h"] \\ \mathcal{F}b' \arrow[r, "f'"'] & \mathcal{G}c' \end{tikzcd}$$

As special cases, one writes $(\mathbf{B}\downarrow\mathcal{G})$ when $\mathbf{B} = \mathbf{A}$ and $\mathcal{F} = \mathbf{1}_\mathbf{A}$, and $(\mathcal{F}\downarrow A)$ when $\mathbf{C}$ is the category $\mathbf{1}$ and $\mathcal{G}$ sends the only object 1 of $\mathbf{1}$ to A in $\mathbf{A}$.

2.6.2. Theorem (Goguen and Burstall [16]). *Let* $\mathbf{B}$ *and* $\mathbf{C}$ *be two cocomplete categories and let* $\mathcal{G}: \mathbf{B} \to \mathbf{C}$ *be a functor. Then the comma category* $(\mathbf{C}\downarrow\mathcal{G})$ *is cocomplete.*

As was noticed earlier, the proof given by Goguen and Burstall makes use of the standard technique of Theorem 2.3.1.

2.7. Algebroidal categories

This notion was introduced by Smyth (see [30, 31, 32]) in order to generalize to categories the notion of algebraic complete partial orders: "they are categories with a countable basis of finite objects" [32] (see also [7] and [6] for a more general definition).

2.7.1. Definitions. An object c in a category $\mathbf{C}$ is *finite* in $\mathbf{C}$ if for any ω-diagram $\Delta = (V_n, f_n)_{n\in\mathbb{N}}$ in $\mathbf{C}$ with ω-limit $\mu: \Delta \to V$ one has: for any arrow $v: c \to V$ there exists an integer N such that for any $n \geq N$, there is a unique arrow $u: c \to V_n$ such that $v = \mu_n \circ u$.

A category $\mathbf{C}$ is *algebroidal* if the three following conditions hold:

(i) $\mathbf{C}$ is initial and has at most countably finite objects,

(ii) every object in $\mathbf{C}$ is the ω-limit of an ω-diagram of finite objects,

(iii) every ω-diagram of finite objects has a colimit in $\mathbf{C}$.

The second condition may be expressed by saying that the full subcategory $\mathbf{C}_0$ of finite objects of $\mathbf{C}$ is ω-*dense* in $\mathbf{C}$.

The following result is proved in [30] and [32, Theorems 4 and 5].

2.7.2. Theorem. *Every algebroidal category has all ω-colimits (is ω-complete). If* $\mathbf{C}$ *is an algebroidal category and* $\mathbf{D}$ *is an ω-category, any functor $\mathscr{F}_0$ from* $\mathbf{C}_0$ *into* $\mathbf{D}$ *extends uniquely (up to natural isomorphism of functors) to an ω-functor from* $\mathbf{C}$ *to* $\mathbf{D}$.

2.8. Effectivity

Dealing with infinite objects raises computability issues. These have been addressed within a categorical framework by several authors such as Smyth ([30] which gives a very good introduction to the topic) or Kanda ([20]). We shall borrow some definitions here from [32] since they are fully adequate in our context. Basic definitions of the theory of computability may be found, e.g. in [34].

2.8.1. Definition. Let $\mathbf{C}$ be an algebroidal category and $\mathbf{C_0}$ be the subcategory of finite objects of $\mathbf{C}$. Assume that $(C_n)_{n\in\mathbb{N}}$ and $(f_n)_{n\in\mathbb{N}}$ are enumerations of the objects and arrows of $\mathbf{C}_0$. We say that $\mathbf{C}$ is *effectively given* relative to these enumerations, provided that the following predicates are recursive in the indices:

(i) $C_i = C_j, f_i = f_j$,

(ii) $\mathrm{dom}(f_k) = A_i$, $\mathrm{cod}(f_k) = A_i$,

(iii) f_k is an identity,

(iv) $f_i \circ f_j = f_k$.

An object is *effectively given* if it is given as the colimit of an effective ω-diagram of finite objects, that is as the colimit of a diagram $(A_{r(i)}, f_{s(i)})_{i\in\mathbb{N}}$ in $\mathbf{C_0}$, where the functions r and s are recursive.

A morphism is *effectively given* if it is given as the colimit of an effective ω-diagram of finite morphisms, that is as the colimit of a diagram $(A_{r(i)}, f_{s(i)})_{i\in\mathbb{N}}$ in the category of arrows of $\mathbf{C_0}$, where the functions r and s are recursive.

A *computable functor* $\mathscr{F}$ is an ω-continuous functor such that for every finite object A and every finite morphism f, $\mathscr{F}A$ and $\mathscr{F}f$ are effectively given.

The following result follows from these definitions.

2.8.2. Proposition. *The initial fixpoint of a computable functor is effectively given.*

3. Hypergraphs

In this paper, we consider oriented hyperedge-labelled hypergraphs having a finite sequence of distinguished vertices called sources. More formally, let $A = \bigcup_{n\in\mathbb{N}} A_n$ be a ranked alphabet with rank function $\tau : A \to \mathbb{N}$. Then:

3.1. Definitions

A (*concrete*) *sourced hypergraph of type n* (or an *n-hypergraph*) *over A*, is a sextuplet $G = \langle V_G, E_G, \mathrm{lab}_G, \mathrm{vert}_G, \mathrm{src}_G, n\rangle$ where:

- V_G and E_G are countable sets (respectively of vertices and hyperedges),
- $\mathrm{lab}_G : E_G \to A$ labels every hyperedge e with some letter in A such that $\tau(\mathrm{lab}_G(e)) = |\mathrm{vert}_G(e)|$,
- $\mathrm{vert}_G : E_G \to V_G^*$ assigns to each e, a word on V_G representing the ordered sequence of its vertices, $\mathrm{vert}_G(k, e)$ denoting the kth element of the sequence,
- $\mathrm{src}_G : [n] \to V_G$ defines the sequence $\mathrm{src}_G([n])$ of *sources* of the hypergraph G.

The set of n-hypergraphs over A is denoted by $\mathfrak{H}\mathfrak{G}_n^\infty(A)$, the set of all hypergraphs by $\mathfrak{H}\mathfrak{G}^\infty(A)$, the corresponding sets of finite hypergraphs by omitting the symbol ∞.

It is easily checked that an alternative definition could be the following: let $G = \langle V_G, L_G, \mathrm{lab}_G, \mathrm{src}_G, n\rangle$, where V_G, src_G and n are as before, L_G is a finite language over V_G and $\mathrm{lab}_G : L_G \to A$. Clearly, L_G is meant to encompass both E_G and vert_G. We shall use this definition in Section 3.8.

For all hypergraphs of type 0, we shall omit the two last components of the definition and simply talk of hypergraphs. Clearly, with any n-hypergraph G, we can associate a unique hypergraph G° (called the *underlying hypergraph*) by simply discarding these two last components. In this way, we can consider an n-hypergraph G as a triple: $G = \langle G^\circ, \mathrm{src}_G, n\rangle$.

A sourced hypergraph is *finite* whenever both sets V_G and E_G are finite. Such hypergraphs have been defined in [4]. A very similar notion has been introduced independently in [18].

A sourced hypergraph is *bounded* when V_G is finite and *has locally finite degree* when every vertex has finite degree, where the degree of a vertex v is the number of times v appears as a vertex of some hyperedge of the hypergraph.

Let X be a ranked alphabet whose elements $x_1, \dots, x_n, \dots$ will be called *variables.* The set of *hypergraphs with variables in X over the alphabet A* is the set $\mathfrak{H}\mathfrak{G}^\infty(A \cup X)$. Most definitions and results hold for hypergraphs with or without variables. The variables will explicitly appear only when necessary.

3.2. Examples

(i) Numerous examples of finite sourced hypergraphs may be found in [4]. Let us simply recall that with each integer n, we associate the discrete n-hypergraph $\underline{n}$ having n distinct isolated vertices, no hyperedge and a source mapping sending $i \in [n]$ to the ith vertex of $\underline{n}$. This hypergraph will be drawn as

$$\bullet_1 \qquad \bullet_2 \qquad \bullet_3 \dots \bullet_{n-1} \qquad \bullet_n.$$

Its underlying hypergraph is merely $[n]$.

Similarly, with a letter $a \in A$ with rank n, we associate the n-hypergraph $\underline{a}$ with n distinct vertices, one hyperedge labelled by a, and source mapping sending $i \in [n]$ to the ith vertex of $\underline{a}$ (Fig. 4).

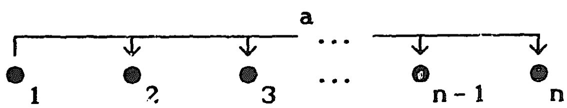


Fig. 4.

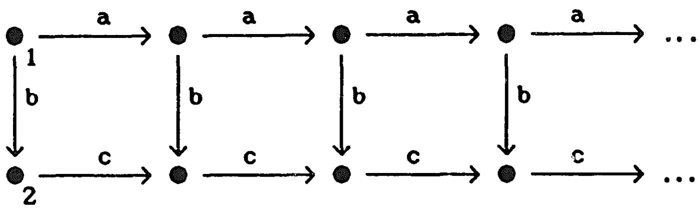


Fig. 5.

(ii) The 2-hypergraph G_1 shown in Fig. 5 is infinite, but has a locally finite degree. This 2-hypergraph might be formally described by

- $A = \{a, b, c\}$, $V_G = E_G = \mathbb{N}$, $n = 2$, $\mathrm{vert}_G(1) = 1.2$,
- $\mathrm{vert}_G(i) = \mathrm{vert}_G(1, i+1).\mathrm{vert}_G(1, i+2)$,
- $\mathrm{vert}_G(i+1) = \mathrm{vert}_G(1, i).\mathrm{vert}_G(1, i+3)$,
- $\mathrm{vert}_G(i+2) = \mathrm{vert}_G(2, i).\mathrm{vert}_G(2, i+3)$,
- $\mathrm{lab}_G(i) = b$, $\mathrm{lab}_G(i+1) = a$, $\mathrm{lab}_G(i+2) = c$,
- $\mathrm{src}_G(1) = 1$, $\mathrm{src}_G(2) = 2$.

(iii) The 2-hypergraph G_2 shown in Fig. 6 is bounded but not finite, hence does not have a locally finite degree (a formal definition is left to the reader).

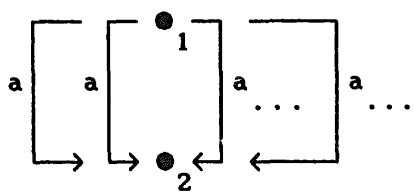


Fig. 6.

3.3. Hypergraph morphisms

3.3.1. Definition. Let G and G' be two concrete sourced hypergraphs with respective types n and n'. A *morphism of sourced hypergraphs* $g : G \to G'$ is a triple $g = (Vg, Eg, \varphi)$ of arrows in **Set** such that

$$Vg : V_G \to V_{G'} \quad Eg : E_G \to E_{G'} \quad \varphi : [n] \to [n']$$

satisfy the following conditions:

$$\mathrm{lab}_G = \mathrm{lab}_{G'} \circ Eg,$$

$$Vg^* \circ \mathrm{vert}_G = \mathrm{vert}_{G'} \circ Eg,$$

$$Vg \circ \mathrm{src}_G = \mathrm{src}_{G'} \circ \varphi$$

(where Vg^* denotes the canonical extension of Vg to the monoid of words on V_G) expressing the commutativity of the following diagrams in the category **Set**:

$$\begin{tikzcd} E_G \arrow[r, "Eg"] \arrow[d, "\mathrm{lab}_G"'] & E_{G'} \arrow[dl, "\mathrm{lab}_{G'}"] \\ A & \end{tikzcd} \qquad \begin{tikzcd} E_G \arrow[r, dashed, "Eg"] \arrow[d, "\mathrm{vert}_G"'] & E_{G'} \arrow[d, "\mathrm{vert}_{G'}"] \\ V_G^* \arrow[r, "Vg^*"'] & V_{G'} \end{tikzcd} \qquad \begin{tikzcd} {[n]} \arrow[r] \arrow[d, "\mathrm{src}_G"'] & {[n']} \arrow[d, "\mathrm{src}_{G'}"] \\ V_G \arrow[r, "Vg"'] & V_{G'} \end{tikzcd}$$

3.3.2. Lemma. *The set of morphisms of concrete n-hypergraphs is closed under composition (defined componentwise). For any sourced hypergraph, the identity morphism is the unique morphism whose components are all set identities.*

It follows from this lemma that, with the morphisms we have just defined, the set $\mathfrak{C}\mathfrak{G}^\infty(A)$ of concrete sourced hypergraphs is a category. Restricting to finite hypergraphs and corresponding morphisms we obtain the category $\mathfrak{C}\mathfrak{G}(A)$ of finite sourced hypergraphs (as introduced in [4]).

Restricting the morphisms to those whose third component φ is the identity on $[n]$, we define for each integer n the category of concrete n-hypergraphs that we shall denote by $\underline{\mathfrak{C}\mathfrak{G}}_n^\infty(A)$. We shall let $\underline{\mathfrak{C}\mathfrak{G}}_n(A)$ denote the full subcategory of finite n-hypergraphs (note: the greater generality of $\mathfrak{C}\mathfrak{G}^\infty(A)$ will be needed only in the second part).

Finally, we shall let the category $\underline{\mathfrak{C}\mathfrak{G}}^\infty(A)$ be the union of the categories $\underline{\mathfrak{C}\mathfrak{G}}_n^\infty(A)$ for $n \in \mathbb{N}$.

3.3.3. Lemma. *A morphism of sourced hypergraphs is a monomorphism (resp. an epimorphism, resp. an isomorphism) iff its components are such when considered as arrows in* **Sets**.

3.3.4. Abstract hypergraphs

As noticed in [4], many properties of hypergraphs do not depend on any specific enumeration of their hyperedges or vertices. This leads us to consider for each of the categories of concrete hypergraphs that we have heretofore defined, the associated skeletal [28] category of abstract hypergraphs which, if needed, will be denoted merely by omitting the prefix $\mathfrak{C}$.

3.4. Theorem. *The category $\underline{\mathfrak{C}\mathfrak{G}}_n^\infty(A)$ has the following properties:*

(i) *it is cocomplete hence is ω-complete and has pushouts,*

(ii) *the discrete n-hypergraph $\underline{n}$ is an initial object of the category and for any object G we let $\perp_G$ denote the unique arrow $\perp_G : \underline{n} \to G$,*

(iii) *if $\mathcal{F}$ is an ω-continuous endofunctor of $\underline{\mathfrak{C}\mathfrak{G}}_n^\infty(A)$, its fixpoints form a category which has an initial object.*

Proof. (iii) is a mere consequence of (i) and (ii) thanks to Theorem 2.5. Point (i) could be given a direct proof which is merely a lengthy routine checking. We prefer to give here a shorter proof using Theorem 2.6.2, hence inspired from [16].

Let $\mathcal{Q}$ be the endofunctor of **Set** which sends a set V to the set V^* of words over V and any arrow $h: V \to W$ to its extension h^* to words. Then the comma category $\mathbf{G} = (\mathbf{Set}\downarrow\mathcal{Q})$ is the category whose objects are triples $(E, V, \mathrm{vert}_G: E \to \mathcal{Q}V = V^*)$ where E and V are two sets, i.e. the category of directed unlabelled hypergraphs (the arrows of the comma category are actually hypergraph morphisms). It follows from Theorem 2.6.2 that $\mathbf{G}$ is cocomplete.

Now let A be a ranked alphabet and let $\mathcal{A}$ be the hypergraph $(A, \{\bullet\}, \mathrm{vert}_{\mathcal{A}}: A \to \{\bullet\}^*)$ in $\mathbf{G}$, where $\{\bullet\}$ denotes the one element set (vert_A is a kind of "representation" of the arity function τ). Then $\underline{\mathfrak{CG}}_0^{\infty}(A)$ is the comma category $(\mathbf{G}\downarrow\mathcal{A})$ (defined with $\mathbf{B} = \mathbf{A}$, $\mathcal{F} = \mathbf{1}_{\mathbf{A}}$, $\mathbf{C} = \mathbf{1}$ and $\mathcal{G}$ sending the only object 1 of $\mathbf{1}$ to $\mathcal{A}$). Indeed the objects of this category are triples $(G, 1, \mathrm{lab}_G: G \to \mathcal{A})$ where lab_G is a hypergraph morphism, i.e. a pair $(k: E \to A, h: V \to \{\bullet\})$ such that $\mathrm{vert}_{\mathcal{A}} \circ k = h^* \circ \mathrm{vert}_G$. Its component k is the actual labelling of the hypergraph. It follows from Theorem 2.6.2 that $\underline{\mathfrak{CG}}_0^{\infty}(A)$ is cocomplete.

Let $\underline{n} = (\emptyset, [n], \emptyset)$ be the discrete hypergraph with n vertices. Then $\underline{\mathfrak{CG}}_n^{\infty}(A) = (\underline{n}\downarrow\underline{\mathfrak{CG}}_0^{\infty}(A))$ is the category of objects of $\underline{\mathfrak{CG}}_0^{\infty}(A)$ under $\underline{n}$ (a comma category with $\mathbf{A} = \mathbf{C} = \underline{\mathfrak{CG}}_0^{\infty}(A)$, $\mathcal{G} = \mathrm{Id}$, $\mathbf{B} = \mathbf{1}$ and $\mathcal{F}$ sending 1 to $[n]$), i.e. of triples $(1, G, \mathrm{src}_G: \underline{n} \to G)$.

Now let $\mathcal{F}: \mathbf{J} \to \underline{\mathfrak{CG}}_n^{\infty}(A)$ be a diagram, namely $\mathcal{F}$ sends any arrow $i \to j$ into a commuting diagram

$$
\begin{tikzcd}
\underline{n} \arrow[r] \arrow[dr] & \mathcal{F}i \arrow[d] \\
 & \mathcal{F}j
\end{tikzcd}
$$

It defines a diagram $\mathcal{F}_0: \mathbf{J} \to \underline{\mathfrak{CG}}_0^{\infty}(A)$ with colimiting cocone $\mathcal{F}_i \xrightarrow{g_i} G$ and the commutation properties of the diagram show that $(G, g_i \circ \mathrm{src}_{g_i})$ is a colimiting cocone for the original diagram. This concludes the proof of (i).

Lastly, it follows from the definition of $\underline{\mathfrak{CG}}_n^{\infty}(A)$ as a category of objects under $\underline{n}$ that $\underline{n}$ is an initial object of the category. $\square$

3.5. *Operations on hypergraphs [4]*

In this section, we briefly recall from [4] the definition of three elementary operations on hypergraphs, whose composition will define some interesting functors.

3.5.1. *Sum of two hypergraphs*

Let H' and H'' be two (concrete) hypergraphs of respective types n' and n''. The sum $H' \oplus H''$ of these two hypergraphs is the $(n'+n'')$-hypergraph $H = \langle H^\circ, \mathrm{src}_H, n'+n'' \rangle$ where:

- H° is the *disjoint* union of the 0-hypergraphs H'° and H''°,
- $\mathrm{src}_H(i) = \mathrm{src}_{H'}(i)$ for $1 \leq i \leq n'$, $\mathrm{src}_H(i) = \mathrm{src}_{H''}(i - n')$ for $n' < i \leq n' + n''$.

In other words, the sum of two hypergraphs is a distinguished element of the isomorphism class of the coproduct of the two hypergraphs in the category of sourced hypergraphs.

3.5.2. Redefinition of sources

For any map $\alpha:[n]\to[p]$, the source redefinition mapping σ_α is defined as follows:

$$\text{for any concrete } n\text{-hypergraph } H=\langle H^\circ, \mathrm{src}_H, n\rangle,$$

$$\sigma_\alpha(H)=\langle H^\circ, \mathrm{src}_H\circ\alpha, n\rangle.$$

3.5.3. Source fusion

For any equivalence relation δ on $[n]$, the source fusing mapping Θ_δ is defined by: if $H=\langle H^\circ, \mathrm{src}_H, n\rangle$, $\Theta_\delta(H)=\langle H'^\circ, \mathrm{src}_{H'}, n\rangle$ where

- $V_{H'}$ is the quotient of V_H by the equivalence relation

$$v\simeq v' \Leftrightarrow v=v' \quad \text{or} \quad (v=\mathrm{src}_H(i) \text{ and } v'=\mathrm{src}_H(i') \text{ for some } (i,i')\in\delta,$$

- $E_{H'}=E_H$,
- $\mathrm{vert}_{H'}=f\circ\mathrm{vert}_H$ where f is the canonical surjection $v_H\to V_{H'}$
- $\mathrm{lab}_{H'}=\mathrm{lab}_H$ and $\mathrm{src}_{H'}=\mathrm{src}_H$.

3.5.4.

Various examples explaining and illustrating the use of these operations will be found in [4].

3.5.5. Definition. A functor is *ranked* if its object component maps n-hypergraphs into n-hypergraphs. A functor $\mathcal{F}$ is a *derived operator* if it is a composition of the three basic operation functors $\oplus$, σ_α, Θ_δ.

The following result is quite straightforward.

3.5.6. Proposition. $\oplus$ *defines an ω-continuous bifunctor from $\mathfrak{C}\mathfrak{G}^\infty_{n'}(A)\times\mathfrak{C}\mathfrak{G}^\infty_{n''}(A)$ to $\mathfrak{C}\mathfrak{G}^\infty_{n'+n''}(A)$, σ_α is an ω-continuous functor from $\mathfrak{C}\mathfrak{G}^\infty_{n}(A)$ to $\mathfrak{C}\mathfrak{G}^\infty_{p}(A)$ and Θ_δ is an ω-continuous endofunctor of $\mathfrak{C}\mathfrak{G}^\infty_{n}(A)$. If an endofunctor $\mathcal{F}$ of $\underline{\mathfrak{C}\mathfrak{G}}^\infty(A)$ is a derived operator, it is ω-continuous and has a fixpoint.*

3.5.7. Remark. This set of operations as defined originally in [4], has no intrinsic good properties besides its simplicity and may in other situations be replaced by various sets of *ad hoc* operations, with the same generating power but nicer properties. Nevertheless, they will perfectly fit our own purposes.

3.6. Substitution

3.6.1. Definition. Let G be an n-hypergraph with one occurrence of a variable x of type m, i.e. one hyperedge e_i of which is labelled by x. Let then Γ be the $(n+m)$-hypergraph defined by (with $\upharpoonright$ denoting restriction):

$$V_\Gamma = V_G, \qquad E_\Gamma = E_G - \{e_i\}, \qquad \mathrm{lab}_\Gamma = \mathrm{lab}_G \upharpoonright E_\Gamma, \qquad \mathrm{vert}_\Gamma = \mathrm{vert}_G \upharpoonright E_\Gamma,$$

$$\mathrm{src}_\Gamma : [n+m] \to V_\Gamma,$$

$$k \to \begin{cases} \mathrm{src}_G(k) & \text{for } 1 \leq k \leq n, \\ \mathrm{vert}_\Gamma(k-n, e_i) & \text{for } n+1 \leq k \leq n+m. \end{cases}$$

The hypergraph Γ is said to be the *context* of the occurrence of x in G and we write $G = \Gamma[x]$. The vertices of the hyperedge e_i labelled by x will be called the *occurrence vertices* of x. They are the vertices common to the context and to the occurrence of the variables. The following result is but a rephrasing of the definition.

3.6.2. Lemma. *If* $G = \Gamma[x]$, *then* $G = \sigma_\alpha \Theta_\delta(\Gamma \oplus \underline{x})$ where $\delta \in \mathrm{Equiv}[n+m+m]$ is generated by $\{(n+i, n+m+i) / 1 \leq i \leq m\}$ and $\alpha : [n] \to [n+m+m]$ is the canonical injection.

3.6.3. Substitution [4] (or hyperedge replacement [18])

Let $G = \Gamma[x]$ be an n-hypergraph with one occurrence of a variable x of type m and let H be an m-hypergraph. Then, *the result of the substitution* of H for x in G is the n-hypergraph $G' = \Gamma[H/x] = \sigma_\alpha \Theta_\delta(\Gamma \oplus H)$ (with α and δ as defined in the previous lemma).

3.6.4. Proposition. *Any finite or infinite n-hypergraph* $G = \Gamma[x]$ *with one occurrence of a variable x of type m gives rise to a derived operator* $\mathscr{F}_\Gamma$ *from* $\underline{\mathfrak{CG}}_m^\infty(A \cup X)$ *to* $\underline{\mathfrak{CG}}_n^\infty(A \cup X)$ *whose object component maps* $\Gamma[x]$ *into* $\Gamma[H/x]$. *In other words, substitution is an ω-continuous functor.*

Proof. At set level the result of the substitution is defined by $V_{G'} = (V_\Gamma \cup V_H)/_\equiv$ where $\equiv$ is the equivalence relation generated by the pairs $(\mathrm{src}_\Gamma(k+n), \mathrm{src}_H(k))$ for all k, $1 \leq k \leq m$,

$$E_{G'} = E_\Gamma \cup E_{H'} \qquad \mathrm{lab}_{G'} = \mathrm{lab}_\Gamma \cup \mathrm{lab}_{H'}$$

$$\mathrm{vert}_{G'} = \mathrm{vert}_\Gamma \cup \mathrm{vert}_{H'} \qquad \mathrm{src}_{G'} = \mathrm{src}_G.$$

The arrow component $\mathscr{F}_\Gamma$ is defined for any arrow $g : H \to H'$ to be the identity on Γ and g on H. $\square$

Intuitively, the hyperedge of G corresponding to the variable x has been removed and replaced by the hypergraph H, the sources of H being glued to the corresponding sources of the context of x. The simultaneous substitution of H for all occurrences of the variable x in G is defined in the very same way, only the formal definition is slightly more awkward.

Indeed, assuming that G has p occurrences of the m-variable x, we can write $G = \Gamma[x^p] = \sigma_\alpha \Theta_\delta(\Gamma \oplus \underset{=}{x} \oplus \cdots \oplus \underset{=}{x})$ for some appropriate α and δ, $\underset{=}{x}$ appearing p times in the expression. The previous construction clearly defines a derived operator F_Γ from $(\underline{\mathfrak{C}\mathfrak{G}}_m^\infty(A))^p$ into $\underline{\mathfrak{C}\mathfrak{G}}_n^\infty(A)$. The substitution functor is now defined to be the composition $\mathscr{F}_\Gamma$ of F_Γ and of the diagonal functor $\Delta^p : \underline{\mathfrak{C}\mathfrak{G}}_m^\infty(A) \to (\underline{\mathfrak{C}\mathfrak{G}}_m^\infty(A))^p$ which sends any object H to the p-uple $(H, \ldots, H)$ and any arrow to the corresponding p-uple.

Similarly, the definition may be extended to simultaneous substitution for a finite number of variables. Note that this substitution may be (and has been) described in a purely categorical framework as a pushout (cf. [14] and [4] where the algebraic and categorical approach are shown to coincide).

3.7. Subhypergraphs

3.7.1. Definition. Let G and G' be two n-hypergraphs. We say that G is a *sub-n-hypergraph* of G' if:

- $V_G \subseteq V_{G'}$
- $E_G \subseteq E_{G'}$
- $\mathrm{lab}_G = \mathrm{lab}_{G'} \upharpoonright E_G$,
- $\mathrm{vert}_G = \mathrm{vert}_{G'} \upharpoonright E_G$,
- $\mathrm{src}_G = \mathrm{src}_{G'}$.

Note that this notion of sub-n-hypergraph respects the sources of the n-hypergraphs. However, when there are no sources (i.e. $n = 0$), this notion is exactly the classical notion of a subhypergraph.

The *frontier* $\mathrm{fr}(G', G)$ of G in G' is the set of vertices of G belonging to a hyperedge of G' which is not in G.

If $G = \Gamma[x]$ is a hypergraph with a variable x and e is a hyperedge labelled with an occurrence of the variable x, the occurrence vertices of (this occurrence) of x are precisely the vertices in the frontier in G of the hypergraph reduced to the hyperedge e.

3.7.2. Proposition. (i) *Let H be a sub-n-hypergraph of H'. Then there exists an m-hypergraph G such that $H' = H[G/x]$, where x is an m-variable labelling an extra hyperedge glued to the frontier of H in H',*

(ii) *Let H be an n-hypergraph, H' an n'-hypergraph and let us assume that H° is a subhypergraph of H'° (i.e. with no sources involved) but that H is* not *a sub-n-hypergraph of H'. Then there exists an integer m and an $(n' + m)$-hypergraph G such that $H' = G(H/x]$. In this case, we say that G is the* context *of H in H'.*

Proof. (i) We let the underlying hypergraph G° of G be defined as follows:
- $V_G = V_{H'} - V_H \cup \mathrm{fr}(H', H)$,
- $E_G = E_{H'} - E_H$,
- $\mathrm{lab}_G = \mathrm{lab}_{H'} \upharpoonright G$,
- $\mathrm{vert}_G = \mathrm{vert}_{H'} \upharpoonright G$.

Let m be the cardinality of the frontier of H in H' and let $\mathrm{src}_G([m])$ be precisely this frontier. Let x be a variable x of type m. We can consider H as an $(n+m)$-hypergraph by adding $\mathrm{src}_G([m])$ to its own sources and set $H' = H[G/x]$.

(ii) Let G° be same hypergraph as in (i). Then, we let G have as sources $\mathrm{src}_{H'}([n']) \cup \mathrm{fr}(H', H)$. Now if we let x be a variable of type $m = \#\mathrm{fr}(H', H)$ we can write $H' = G[H/x]$. □

3.8. Approximations of an infinite hypergraph

3.8.1. A sequence of finite hypergraphs approximating an infinite hypergraph

Let $G = \langle E, V, \mathrm{vert}_G, \mathrm{lab}_G, \mathrm{src}_G \rangle$ be an n-hypergraph and $I \subseteq V$ be the set of its isolated vertices which are not sources.

Let us assume that enumerations $\varepsilon : \mathbb{N} \to E$ of the set of hyperedges and $\nu : \mathbb{N} \to I$ of the set of isolated vertices are given. Let $\mu : \mathbb{N} \to \{0, 1\}$ be the characteristic mapping of the domain of ν.

We can now enumerate the vertices of G in the following way:
- first we take the sources of G in the same order, obtaining p_0 distinct elements for some $1 \leqslant p_0 \leqslant n$,
- then we alternatively enumerate the vertices of the hyperedges of G in the order given by ε and the isolated vertices in the order given by ν.

More precisely, if $\rho(e_i)$ is the number of vertices of e_i which have not yet been enumerated, those vertices will be numbered by some integers k with

$$p_{i-1} = p_0 + \sum_{j=1}^{j=i-1} \rho(e_j) \leqslant k \leqslant p_0 + \sum_{j=1}^{j=i} \rho(e_j),$$

and we shall set $p_i = p_{i-1} + \rho(e_i) + \mu(i)$ in order to take into account the ith isolated vertex.

Let us set $G_0 = \underline{n}$ and let $G_i = \langle E_i, V_i, \mathrm{vert}_{G_i}, \mathrm{lab}_{G_i}, \mathrm{src}_{G_i} \rangle$ be the n-hypergraph defined in the following way for $i \geqslant 1$:
- $E_i = \{e_j = \varepsilon(j) \mid 1 \leqslant j \leqslant i\} \subseteq E$,
- $V_i = \{v_j \mid 1 \leqslant j \leqslant p_i,\ v_j \in \mathrm{vert}(e_k,$ or $v_j = \nu(k)$ for some $k \leqslant i\} \subseteq V$,
- $\mathrm{vert}_{G_i}, \mathrm{lab}_{G_i}, \mathrm{src}_{G_i}$ are the restriction of vert_G, lab_G, src_G to E_i and V_i.

The proof of the following proposition is now routine checking.

3.8.2. Proposition. *The canonical monomorphisms $\iota_i : G_i \to G_{i+1}$ for $i \geqslant 0$ define an ω-diagram (that we shall call an* approximating diagram *for G) whose ω-limit is the original hypergraph G. Of course, the result still holds if G is finite.*

The approximation problem may also be considered from the point of view of morphisms. Namely, let us assume that we are given two hypergraphs G and G' with two approximating diagrams $(G_n, \iota_n)_{n\in\mathbb{N}}$ and $(G'_n, \iota'_n)_{n\in\mathbb{N}}$ colimiting cocones $(G, \gamma_n)_{n\in\mathbb{N}}$ and $(G', \gamma'_n)_{n\in\mathbb{N}}$ and a morphism $g: G \to G'$.

For any integer n, the restriction of g to G_n defines a subhypergraph of G'. Since the category of hypergraphs is algebroidal, there is an integer $\zeta(n)$ and an arrow $g_n: G_n \to G_{\zeta(n)}$ such that, if m is any other integer $m \geq n$, one has $\iota'_{\zeta(n),\zeta(m)} \circ g_n = g_m \circ \iota_{n,m}$, where for $p \leq q$, $\iota_{p,q}$ denotes the composition of all the ι' from p to q.

This construction defines an ω-diagram $Z = (g_n, (\iota_n, \iota_{\zeta(n)}))$ in the category of arrows. It is now readily checked that:

3.8.3. Proposition. *The cocone* $(g, (\gamma_n, \gamma'_{\zeta(n)}))$ *over* Z *is a colimiting cocone in the category of arrows.*

Or, in other words, that the morphism g is approximated by the sequence $(g_n)_{n\in\mathbb{N}}$.

We can now investigate some structural properties of the category of n-hypergraphs.

3.8.4 Lemma. *Any finite n-hypergraph is a finite object in the category* $\underline{\mathfrak{CG}}_n^\infty(A)$.

Proof. Let $H = \langle E_H, V_H, \mathrm{vert}_H, \mathrm{lab}_H, \mathrm{src}_H\rangle$ be a finite n-hypergraph and $(G_i, g_i)_{i\in\mathbb{N}}$ be an ω-diagram in $\underline{\mathfrak{CG}}_n^\infty(A)$ with ω-limit $C = \langle E_G, V_G, \mathrm{vert}_G, \mathrm{lab}_G, \mathrm{src}_G\rangle$ and ω-limiting cone (G, γ_i). Let $h: H \to G$ be an n-hypergraph morphism, i.e. a pair (Eh, Vh) of set mappings satisfying the required commutation relations. Since both E_H and V_H are finite, there exist some integer N such that $E_H \subseteq E_{G_N}$ and $V_H \subseteq V_{G_N}$. Moreover there are unique inclusion mappings $I: E_H \to E_{G_N}$ and $J: V_H \to V_{G_N}$ in **Set** defining an n-hypergraph monomorphism $v_N: H \to G_N$ such that $h = \gamma_N \circ v_N$. For any $n \geq N$, $v = g_n \circ g_{n-1} \circ \cdots \circ g_N \circ v_N$ is the unique arrow such that $h = \gamma \circ v_n$. □

Most of the interesting properties of infinite hypergraphs arise from the following.

3.8.5. Theorem. *The category of hypergraphs is algebroidal.*

Proof. Parts (ii) and (iii) of Definition 2.7.1 have been proved respectively in Proposition 3.8.2 and Theorem 3.4. Part (i) follows from the previous lemma. □

Let us now assume either that a recursive enumeration $\chi: \mathbb{N} \to A$ of the alphabet is given or that for each integer n, the set A_n of letters of arity n is finite (in which case such an enumeration is easily built). For such a χ, we let $\rho: A \to \mathbb{N}$ be such that $\rho(a) = i \Leftrightarrow \chi(i) = a$. Then:

3.8.6. Proposition. *The objects and arrows of the subcategory of finite hypergraphs can be enumerated in such a way that the category* $\underline{\mathfrak{CG}}_n^\infty(A)$ *of hypergraphs is effectively* *given. Then, a hypergraph G is effectively given if and only if its sets of hyperedges and isolated vertices can be recursively enumerated (i.e. if the mapping ε and ν of Section 3.8.1 are recursive).*

Proof. Let us first define an effective enumeration of the objects of $\mathfrak{CG}_n(A)$. Let $G=\langle V_G, L_G, \mathrm{lab}_G, \mathrm{src}_G, n\rangle$ be a finite n-hypergraph, with $V_G=\{v_1,\ldots,v_p\}$ and $\mathrm{src}_G([n])=\{v_{s_1},\ldots,v_{s_n}\}$. Let $p=\#V_G$, $q=\#L_G$, $|w|$ be the length of a word w, let $L_G=\{l_1,\ldots,l_q\}$ be ordered lexicographically with respect to the enumeration χ of A, let $l_1=v_{i_{l_1,1}}\ldots v_{i_{l_1,|l_1|}}$ and $\alpha_i=\mathrm{lab}_G(l_i)$.

We associate with G the following *uniquely defined* sequence of integers: $(n, p, q, |l_1|, \rho(\alpha_1), i_{l_1,1},\ldots, i_{l_1,|l_1|},\ldots, |l_q|, \rho(\alpha_q), i_{l_q,1},\ldots, i_{l_q,|l_q|}, s_1,\ldots,s_n)$ which clearly provides us in the usual way with a *Gödel numbering* of the set of finite hypergraphs.

The rest of the proposition follows routinely. □

4. Hypergraph expressions

In this section, we briefly recall the algebraic framework which has been developed in [4] and show how it can be extended to infinite hypergraphs. This will provide us with a second way to approach problems on hypergraphs whose relationship with the earlier one will be described in Section 4.4. Both methods will be used simultaneously in the sequel.

4.1. Preliminaries (cf. [9])

4.1.1.

Let $\mathcal{S}$ be a set called the set of *sorts*. An *$\mathcal{S}$-signature F* is a set of symbols given with two mappings $\alpha: F\to\mathcal{S}^*$ (the *arity* mapping) and $\sigma: F\to\mathcal{S}$ (the *sort* mapping). The *profile* of $f\in F$ is the word $\alpha(f)\to\sigma(f)$, its rank is the integer $|\alpha(f)|$. A symbol is constant if its rank is 0. Whenever necessary, a subset $U=\{x_1,\ldots,x_n,\ldots\}$ of F will be singled out and its elements called variables.

4.1.2. Tree domains

Let n be an integer. A tree domain is a subset $\mathcal{D}$ of $[n]^*$ such that:

- $\mathcal{D}$ is prefix closed (if $\alpha,\beta\in[n]^*$, $\alpha\beta\in\mathcal{D}\Rightarrow\alpha\in\mathcal{D}$),
- if $\alpha\in[n]^*$, $1\leqslant i\leqslant j$ and $\alpha j\in\mathcal{D}$ then $\alpha i\in\mathcal{D}$.

Let n be the maximum arity of a symbol in F. An *expression* over the signature F is a partial mapping $\gamma:[n-1]^*\to F$ whose domain is a tree domain and such that for $\alpha\in\mathrm{Dom}(\gamma)$, if $\mathrm{rank}(\gamma(a))=k$ then $\alpha i\in\mathrm{Dom}(\gamma)\Leftrightarrow 1\leqslant i\leqslant j$.

An expression is *finite* if $\mathrm{Dom}(\gamma)$ is finite, infinite otherwise. Let $\mathbf{E}^\infty(F)$ (resp. $\mathbf{E}(F)$) denote the set of infinite (resp. finite) expressions over F.

A node $u\in\mathrm{Dom}(\gamma)$ is an *occurrence* of the symbol s in γ if $\gamma(u)=s$. We let $\mathrm{occ}(\gamma, s)$ denote the set of occurrences of the symbol s in γ.

The *subexpression* of γ issued from the node u is the expression ξ with domain $\{v \mid uv \in \mathrm{Dom}(\gamma)\}$ and such that $\xi(v) = \gamma(uv)$. We let $\mathbf{sub}(\gamma)$ denote the set of subexpressions of γ.

An expression is *regular* if it has only a finite number of distinct subexpressions.

The set of *terminal nodes* of an expression $\Phi : D \to F$ is the set of nodes labelled by a constant (its leaves):

$$\Phi_{\#} = \{u \in D \mid \mathrm{rank}(\Phi(u)) = 0\}.$$

An expression is *locally finite* if all its nodes are prefix of some terminal node, *non-proper* if none is such. If a node u is not a prefix of some terminal node, it is clear that the subexpression rooted in u is non-proper.

4.1.3. Many-sorted magmas (or heterogeneous algebras)

A many-sorted F-magma is a family $\mathbf{M} = (M_s, s \in \mathcal{S})$ of sets together with a family of mapping $\bar{f} : M_{s_1} \times \cdots \times M_{s_n} \to M_s$ for each f in F with profile $s_1 \ldots s_n \to s$.

It is well known (see e.g. [9] or [15]) that the set $\mathbf{E}(F)$ of expressions over the signature F has a canonical structure of F-magma and that this F-magma is initial in the category of F-magmas. This means that for any other F-magma $\mathbf{M}$, there is a unique arrow (morphism of F-magmas) $\mathbf{eval}_{\mathbf{M}} : \mathbf{E}(F) \to \mathbf{M}$ (given by a simple recursive definition).

4.1.4. Definition. A *system of regular equations* is a finite system of the form $\Sigma = \langle x_1 = \Sigma_1, \ldots, x_n = \Sigma_n \rangle$, where $U = \{x_1, \ldots, x_n\}$ is a finite set of unknowns, and for each integer i, $1 \leqslant i \leqslant n$, Σ_i is a finite expression with variables in U, and x_i and Σ_i are of the same sort. If the Σ_i's are not required to be finite, the system is called generalized. If they are regular expressions, then the system is said to be extended.

A component to the solution to such a system is called a *regular expression.*

A comprehensive treatment of the methods used to solve such a system in the homogeneous case may be found in [9], which is easily transposed to the many-sorted setting. It is fairly well known that two approaches are available, a metric approach and an order theoretic approach, but we shall only be interested here in the latter. Let us simply recall that one must then add to the set of symbols a bottom element for each of the sorts. The main result is that an extended system of regular equations has a least solution (with respect to some order relation) whose components are regular.

We shall not go any further into this theory. Necessary definitions and results will be presented in our own setting in Definition 4.4.1.

4.2. Hypergraph expressions

4.2.1. Definition. We let $\mathbb{N}$ be considered as a set of sorts. Then, with any ranked alphabet A with rank function $\tau : A \to \mathbb{N}$ we associate an $\mathbb{N}$-sorted signature $\mathbb{H}_A$ which consists of the following symbols:

(1) *Constant symbols*:

$\boldsymbol{a}$ with profile $(\rightarrow \tau(a))$ for each letter a in A,

$\boldsymbol{n}$ with profile $(\rightarrow n)$ for each integer $n \in \mathbb{N}$.

(2) *Operator symbols*:

$\oplus_{n,m}$ with profile $(n, m \rightarrow n+m)$ for each pair $(n, m) \in \mathbb{N}^2$,
$\sigma_{\alpha,n,p}$ with profile $(n \rightarrow p)$ for each pair $(n, p) \in \mathbb{N}^2$ and each mapping $\alpha : [n] \rightarrow [p]$,
$\Theta_{\delta,n}$ with profile $(n \rightarrow n)$ for each $n \in \mathbb{N}$ and each equivalence relation δ on $[n]$.

We let $\mathbf{E}^\infty(A)$ (resp. $\mathbf{E}(A)$) be simpler notations for the many-sorted magma $\mathbf{E}^\infty(\mathbb{H}_A)$ (resp. $\mathbf{E}(\mathbb{H}_A)$) of hypergraph (resp. finite hypergraph) expressions over A (with tree domains over $\{0, 1\}^*$).

If X is a ranked alphabet whose elements of rank n will again be called *n-variables*, we let $\mathbf{E}^\infty(A, X) = \mathbf{E}^\infty(A \cup X)$ be the many-sorted magma of hypergraph expressions over A with variables in X.

It follows from Section 4.1.3 that every finite hypergraph expression may be evaluated to a finite hypergraph. The unique evaluation mapping $\mathbf{eval}_{\mathfrak{G}} : \mathbf{E}(A) \rightarrow \mathfrak{C}\mathfrak{G}(A)$ which is recursively defined by

$$\mathbf{eval}_{\mathfrak{G}}(\boldsymbol{a}) = \underline{a},$$

$$\mathbf{eval}_{\mathfrak{G}}(\boldsymbol{n}) = \underline{n},$$

$$\mathbf{eval}_{\mathfrak{G}}(\oplus_{m,n}) = \oplus,$$

$$\mathbf{eval}_{\mathfrak{G}}(\Theta_{\delta,n}) = \Theta_\delta,$$

$$\mathbf{eval}_{\mathfrak{G}}(\sigma_{\alpha,n,p}) = \sigma_\alpha,$$

$$\mathbf{eval}_{\mathfrak{G}}(s(\Phi_1, \ldots, \Phi_k)) = \mathbf{eval}_{\mathfrak{G}}(s)(\mathbf{eval}_{\mathfrak{G}}(\Phi_1), \ldots, \mathbf{eval}_{\mathfrak{G}}(\Phi_k)),$$

associates with any finite hypergraph expression a *unique* finite hypergraph (*the hypergraph denoted by the expression*). Of course, many distinct hypergraph expressions may be evaluated to the same hypergraph.

4.2.2. Examples

Various examples of finite hypergraph expressions together with their interpretations have been given in [4] to which the reader is referred. Here, we shall only give examples of infinite hypergraph expressions which *intuitively* denote the sample infinite hypergraphs of Section 3.2. We shall see later on that they actually denote these hypergraphs with a very precise meaning.

(i) The hypergraph G_2 may be described by the following expression:

$$\begin{array}{ll} \gamma;\{0,1\}^* & \to \mathbb{H}_A \\ (001)^* & \mapsto \sigma_{\alpha,2,4} \\ (001)^*0 & \mapsto \Theta_{\{(1,3),(2,4)\},4} \\ (001)^*00 & \mapsto \oplus_{2,2} \\ (001)^*000 & \mapsto \boldsymbol{a} \end{array}$$

which may be graphically represented by the infinite tree shown in Fig. 7 (with the same tree reproduced at the ...).

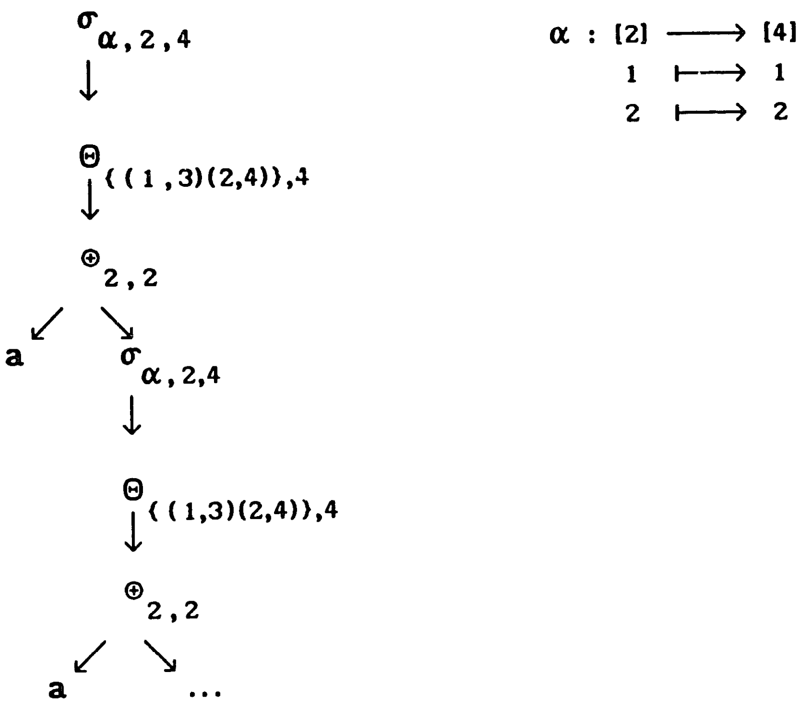


Fig. 7.

(ii) In a similar way, the hypergraph G_1 may be described by a hypergraph expression with the same tree domain which may be depicted by the tree shown in Fig. 8 (... is to denote the reproduction of the same tree at that place) (the expression is more complex and will be left to the reader).

4.2.3. First-order substitution

For any set of variables X, let us define a first-order substitution σ to be a ranked mapping from X into $\mathscr{P}(\mathrm{E}^\infty(A \cup X))$, i.e. a mapping such that $\sigma(x) \in \mathscr{P}(\mathrm{E}_n^\infty(A \cup X))$ whenever $\tau(x) = n$. Then the result of the application of σ to a finite hypergraph

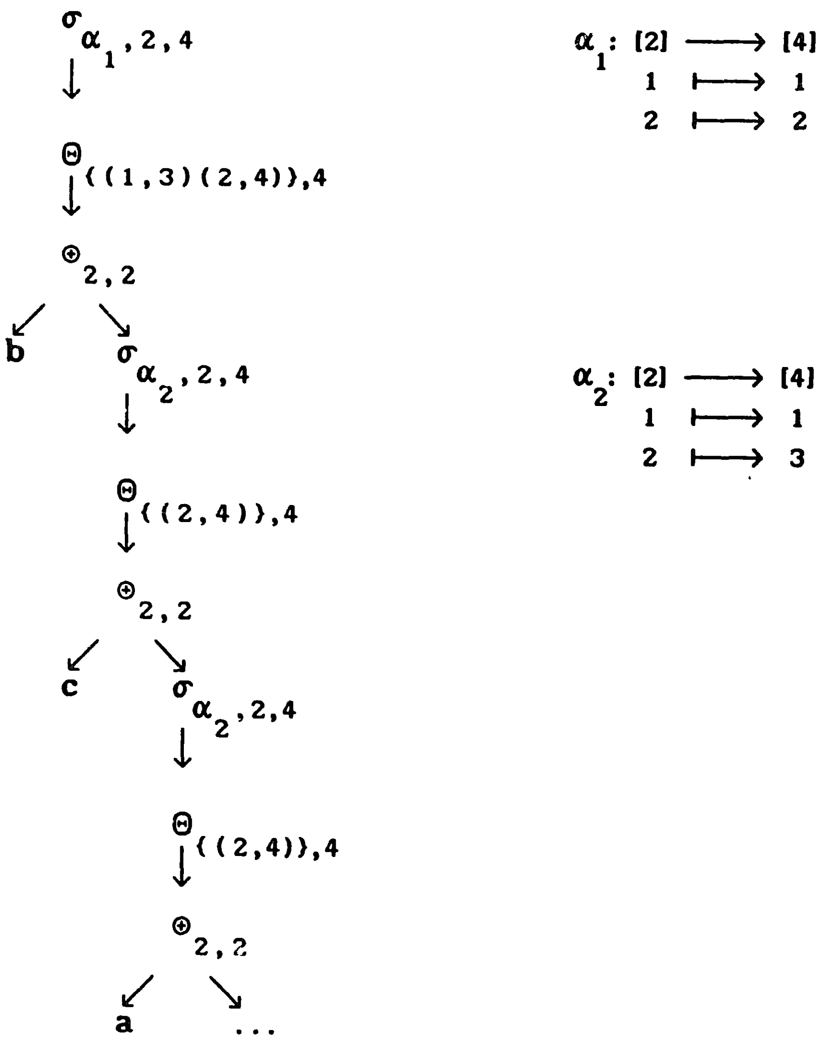


Fig. 8.

expression φ is the hypergraph expression $\sigma(\varphi)$ where every x-labelled leaf of φ has been replaced by some element in $\sigma(n)$. It must be noted that such a substitution is a non-deterministic mapping.

Let us now consider a very simple case. Let φ be a finite hypergraph expression with type n presenting an x-labelled leaf, for some x of rank p, and let φ' be a finite hypergraph expression of type p. Let $\varphi[\varphi'/x]$ denote the result of the substitution of φ' for that occurrence of x in φ.

The recursive evaluation of φ gives out an n-hypergraph $G = \mathbf{eval}_G(\varphi)$. Let $v_1, \ldots, v_p$ be the vertices of G corresponding to the x-labelled leaf and let $\underline{G} = \mathbf{ev}(\varphi)$ be the $n+p$-hypergraph with underlying hypergraph $G°$ and source mapping

$$\mathrm{src}_{\underline{G}} : [n+p] \to V_G$$

$$j \to \mathrm{src}_G(j) \quad \text{for } 1 \leq j \leq n,$$

$$j \to v_{j-n} \quad \text{for } n+1 \leq j \leq n+p.$$

Then a routine proof shows that:

4.2.4. Lemma. (i) $\mathbf{eval}_{\mathfrak{G}}(\varphi[\varphi'/x]) = \pi_\alpha(\Theta_\delta(\mathbf{ev}(\varphi) \oplus \mathbf{eval}_{\mathfrak{G}}(\varphi')))$ *for some mapping* $\alpha : [n] \to [n+2p]$ *and some equivalence relation* δ *on* $[n+2p]$.

(ii) $\mathbf{eval}_{\mathfrak{G}}(\varphi[\varphi'/x]) = \mathbf{ev}(\varphi)[\mathbf{eval}_{\mathfrak{G}}(\varphi')/x]$, *i.e.* $\mathbf{ev}(\varphi)$ *is the context of* $\mathbf{eval}_{\mathfrak{G}}(\varphi')$ *in* $\mathbf{eval}_{\mathfrak{G}}(\varphi[\varphi'/p])$.

4.3. Direct evaluation of an infinite hypergraph expression

In this section, we show how to interpret directly an infinite expression as an infinite hypergraph (i.e. without any reference to the way of interpreting its finite subexpressions by the canonical morphism $\mathbf{eval}_{\mathfrak{G}}$. The relationship between those two approaches will be studied in Section 4.4).

4.3.1. Proposition. *Any hypergraph expression* Φ *evaluates to an abstract hypergraph* G *which will be called the* canonical hypergraph denoted by Φ. *More formally there is a canonical mapping* $\mathbf{eval}_\infty$ *from* $\mathrm{E}^\infty(A)$ *to* $\underline{\mathfrak{C}\mathfrak{G}}^\infty(A)$. *If the expression is infinite, the evaluation is not effective.*

Proof. Let Φ be a (possibly infinite) hypergraph expression. To define the corresponding hypergraph G we first set

$$E_G = \{u \in \mathrm{Dom}(\Phi) \mid \Phi(u) \in A\},$$

$$\mathrm{lab}_G(u) = \Phi(u).$$

We now define a set V of "prevertices":

$$V = \{u;\ i \mid u \in \mathrm{Dom}(\Phi) \text{ and } 1 \leqslant i \leqslant \sigma(\Phi(u))\},$$

i.e. with each node u of the expression of sort n, we associate n vertices. If this node is a leaf labelled by some constant a, it defines a hyperedge with vertices: $u;\ 1, \ldots, u;\ n$. Let us set $\mathrm{vert}(u) = u;\ 1, \ldots, u;\ n$.

We must now explain how the elements will be glued together to form the required infinite hypergraph. Since the general construction is rather formal, we shall first describe the method for a very simple finite expression. Let φ be the expression on the alphabet $\{a, b\}$ with $\tau(a) = 2$ and $\tau(b) = 3$ which is depicted in Fig. 9.

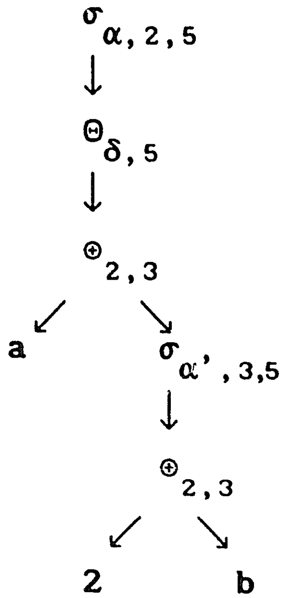
With φ we associate the following set of hyperedges and labelling:

$$\mathrm{E}_G = \{000, 001001\} \quad \text{and} \quad \mathrm{lab}_G = \{(000, a), (001001, b)\}.$$

With the root ε of the expression whose label has sort 2, we associate the elements $\varepsilon 1$ and $\varepsilon 2$, written as 1 and 2. Proceeding, each node of the tree yields the set

$$\begin{aligned} V = \{&1, 2, 0;\ 1, 0;\ 2, 0;\ 3, 0;\ 4, 0;\ 5, 00;\ 1, 00;\ 2, 00;\ 3, 00;\ 4, 00;\ 5,\\ &000;\ 1, 000;\ 2, 001;\ 1, 001;\ 2, 001;\ 3, 0010;\ 1, 0010;\ 2, 0010;\ 3,\\ &0010;\ 4, 0010;\ 5, 00100;\ 1, 00100;\ 2, 00101;\ 1, 00101;\ 2, 00101;\ 3\} \end{aligned}$$

consisting of 27 prevertices, some of which we must identify to glue the hyperedges and build the hypergraph G denoted by φ.



$\alpha : [2] \longrightarrow [5]$
$\alpha = \{ (1,3), (2,1)\}$

δ generated by $\{(1,5),(2,4)\}$

$\alpha' : [3] \longrightarrow [5]$
$\alpha' = \{(1,2),(2,4),(3,3)\}$

Fig. 9.

As an example, let us choose two elements of V: 000; 2 and 0010; 4. Must we identify them or do they correspond to different vertices?

The first one comes from the node labelled by a, the second one from the deepest node labelled by $\oplus_{2,3}$. Going up the tree, the latter will be renumbered by $\sigma_{\alpha',3,5}$ as the second vertex coming from this node, hence must be identified with 001; 2. Then, it will be renumbered by $\oplus_{2,3}$ as the 4th source, while the element 000; 2 will be renumbered as the 2nd one. Let us keep climbing up the tree: $\Theta_{\delta,5}$ fuses the 2nd and 4th sources to its argument. Hence, the two elements we have considered will have to be identified during the evaluation of the expression.

Using these arguments, the reader will easily check that the above graph evaluates to as shown in Fig. 10.

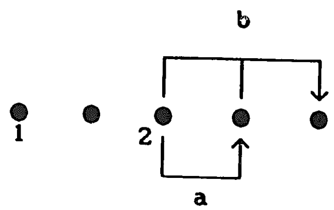


Fig. 10.

We may now turn to the formal proof of the proposition.

Let $\# : \mathrm{Dom}(\Phi) \times V \to \mathbb{N}$ be the partial mapping (called the *renumbering mapping*) defined in the following way:

- $\#(u, u; i) = i$,
- let u' be any prefix of u where $\#(u', u; i)$ is defined. If $u' = tk$ with $k = 0$ or 1, we shall set:

$$\text{if } \Phi(t) = \oplus_{p,q} \begin{cases} k = 0, & \#(t, u; i) = \#(u', u; i), \\ k = 1, & \#(t, u; i) = p + \#(u', u; i), \end{cases}$$

$$\text{if } \Phi(t) = \sigma_{\alpha,p,q}, \qquad \text{if } \alpha(i') = i \text{ then } \#(t, u; i) = i' \text{ else undefined},$$

$$\text{if } \Phi(t) = \Theta_{\delta,n}, \qquad \#(t, u; i) = \#(u', u; i),$$

- undefined otherwise.

The mapping $\#$ may be computed at any node w prefix of u. We let ρ be the equivalence relation on V generated by the set $\{(u; i, v; j) \mid \exists w \in \mathrm{Dom}(\Phi)$, prefix of u and v such that $\Phi(w) = \Theta_{\delta,p,q}$ and $(\#(w, u; i), \#(w, vj)) \in \delta\}$.

We then set $V_G = V/\rho$ with canonical surjection μ and $\mathrm{vert}_G = \mu \circ \mathrm{vert}$.

Let $R = \{w \in \mathrm{Dom}(\Phi) \mid \Phi(w) = \sigma_{\alpha,n,p}\}$ and let w_0 be such that $|w_0| = \min\{|w| \mid w \in R\}$. Then n is the sort of Φ, hence of G and $\mu(u; i)$ is a source of G if and only if $\#(w, u; i) = \alpha(j)$ for some $j \in [n]$.

We let the equivalence class of $G = \langle E_G, V_G, \mathrm{vert}_G, \mathrm{lab}_G, \alpha, n\rangle$ be the abstract hypergraph denoted by the expression Φ. $\square$

The following lemma extends Lemma 4.2.4 to infinite expressions.

4.3.2. Lemma. *Let Φ be a hypergraph expression of type n with one x-labelled leaf and Φ' be a hypergraph expression of type $\tau(x)$. Then*

$$\mathbf{eval}_x(\Phi[\Phi'/\ ]) = \mathbf{eval}_x(\Phi)[\mathbf{eval}_x(\Phi')/x]$$

or

$$\mathbf{eval}_x(\Phi[\Phi'/x]) = \sigma_\alpha(\Theta_\delta(\mathbf{eval}_x(\Phi) \oplus \mathbf{eval}_x(\Phi'))$$

for some mapping $\alpha : [n] \to [n + 2\tau(x)]$ and some equivalence relation δ on $[n + 2\tau(x)]$.

Proof. We use the tools defined in Proposition 4.3.1. Let u be the node of Φ where the substitution takes place, let ρ, ρ' and ρ'' be the equivalence relations associated with Φ, Φ' and $\Phi[\Phi'/x]$. Let $v; i$ and $w; j$ be two prevertices generated by Φ' and $uv; i, uw; j$ be the corresponding prevertices of $\Phi[\Phi'/x]$. Now, the pair $(uv; i, uw; j)$ belongs to ρ'' iff there exists some node t of $\Phi[\Phi'/x]$ prefix of uv and uw, labelled by some $\Theta_{\delta,p,q}$ and such that $(\#(t, uv; i), \#(t, uw; j))$ belongs to δ.

It follows from this definition of ρ'' that for any node z on a path from t to u in the tree, the pair $((z; \#(z, uv; i), (z; \#(z, uw;, j))$ belongs to ρ''.

Let us first assume that u is a prefix of t, i.e. that $t = ut'$. Then $v = t'v'$ and $w = t'w'$ and clearly, to say that $(uv; i, uw; j)$ belongs to ρ'' amounts to saying that $(v'; i, w'; j)$ belongs to ρ'.

On the other hand, if $t \leq u$, i.e. if $u = tu'$, if $(uv;\, i,\, uw;\, j)$ belongs to ρ'' then, in particular $((u;\, \#(u, uv;\, i),\, (u;\, \#(u, uw;\, j))$ belongs to ρ.

This means that two hyperedges of the expression $\Phi[\Phi'/x]$ coming from Φ', generate a common vertex if either they do so in Φ' itself, or they generate source vertices of Φ' which will be glued by the substitution. $\square$

4.3.3. Examples

As suggested earlier, it is easily checked from the previous results that the hypergraph expressions of Section 4.2.2 *actually* denote the hypergraphs they were claimed to denote. Let us rather look at an example of a *non-proper* hypergraph expression such as the following:

$$((000)^*(001)^*)^* \mapsto \sigma_{\alpha,2,4},$$

$$((000)^*(001)^*)^*0 \mapsto \Theta_{\delta,4},$$

$$((000)^*(001)^*)^*00 \mapsto \oplus_{2,2},$$

where the graph of $\alpha : [2] \dashrightarrow [4]$ is $\{(1, 1), (2, 3)\}$ and δ is generated by $\{(1, 4), (2, 3)\}$, which might be depicted by the "complete binary tree" drawn in Fig. 11.

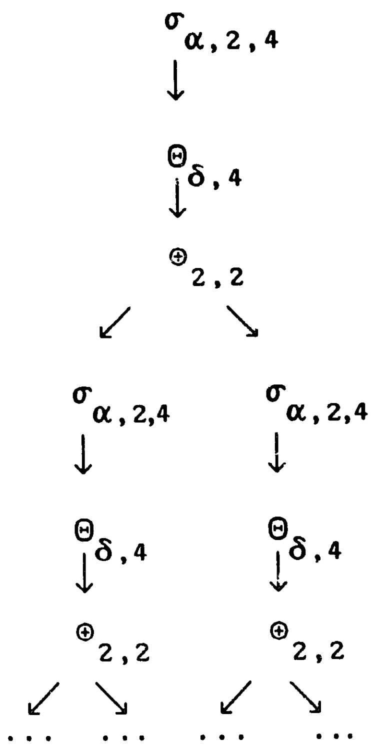


Fig. 11.

It is clear from Proposition 4.3.1 that such an expression evaluates to a discrete hypergraph of type 2 since none of its leaves can generate a hyperedge. Moreover, since δ has only two equivalence classes, the hypergraph has only two distinct vertices. Hence from the definition of α, this hypergraph is isomorphic to the 2-hypergraph $\underline{2}$.

This example shows that any hypergraph expression of this kind with type n evaluates to the discrete n-hypergraph $\underline{n}$, hence is equivalent (in some sense to be made precise later on) to the constant $\boldsymbol{n}$.

4.3.4. Lemma. *For any hypergraph expression Φ there exists a locally finite hypergraph expression Φ' such that* $\mathbf{eval}_\infty(\Phi)$ *and* $\mathbf{eval}_\infty(\Phi')$ *are isomorphic.*

Proof. Indeed, it follows from Lemma 4.3.2 and Section 4.3.3, that any non-proper subexpression of Φ with type n may be replaced by an n-labelled leaf. Therefore, in what follows we shall be mainly concerned with locally finite expressions, that is with expressions which do not contain any components of the aforementioned kind. □

We shall call Φ' the *reduced expression of* Φ. Of course, reducing an infinite hypergraph expression is *not an effective process*, since checking that an expression is not locally finite is not effective.

4.4. Order-theoretic properties of the set of infinite hypergraph expressions

In this section we briefly recall some fundamental properties of infinite expressions. All definitions and results are adapted from [9] where more details may be found.

4.4.1. Definitions. Let us consider the following binary relation on $\mathbf{E}^\infty(A)$: $\Phi \leqslant \Phi' \Leftrightarrow \mathrm{Dom}(\Phi) \subseteq (\Phi')$ and for $u \in \mathrm{Dom}(\Phi)$, $\Phi(u) = \Phi'(u)$ except perhaps if $\Phi(u) = \boldsymbol{n}$ for some $n \in \mathbb{N}$.

It is easily seen that $\leqslant$ is a partial order on $\mathbf{E}^\infty(A)$ and that for each $n \in \mathbb{N}$, $\boldsymbol{n}$ is the least element of $\mathbf{E}_n^\infty(A)$, i.e. that $\boldsymbol{n}$ plays the role of the symbol Ω in the case of ordinary infinite trees.

If $\varphi \in \mathbf{E}(A)$ and $\varphi \leqslant \Phi$ we say that φ is a *finite approximant of* Φ. In the sequel finite expressions will usually be denoted by a lower case greek letter, while upper case greek letters will be used to denote infinite expressions.

4.4.2. Theorem. (i) *$\Phi \leqslant \Phi' \Rightarrow \Phi$ and Φ' both have the same sort,*

(ii) *$\Phi \leqslant \Phi' \Leftrightarrow \Phi' \in \sigma(\Phi)$ for some first-order substitution σ,*

(iii) *for each $n \in \mathbb{N}$, $\mathbf{E}_n^\infty(A)$ is an algebraic ω-complete many-sorted $\mathbb{H}_A$-magma (i.e. every directed set has a least upper bound and any infinite expression is the least upper bound of an increasing sequence of finite expressions),*

(iv) *every ω-continuous mapping of an ω-complete many-sorted magma into itself has a least fixpoint (Tarski's theorem).*

Proof. Result (i) is trivial. All other points are proved in [9]. □

4.4.3. Systems of regular equations on expressions

It is clear from the previous theorem that general definitions as presented in Definition 4.1.4 are easily transposed to the case of hypergraph expressions, yielding a notion of systems of regular equations and a way to solve them.

4.4.4. The category of infinite hypergraph expressions

As noted in Section 2.1, any (pre)ordered set may be viewed as a category. In particular, the set $\mathbf{E}_n^\infty(A)$ of infinite hypergraph expressions may be interpreted as a category, whose objects are precisely the hypergraph expressions, i.e. the partial mappings $\Phi:\{0,1\}^* \to \mathbb{H}_A$, and where there is an arrow from Φ to Φ' whenever $\Phi \leq \Phi'$. From the definition of the partial order, it follows that this arrow is induced by the inclusion mapping $i:\mathrm{Dom}(\Phi) \to \mathrm{Dom}(\Phi')$.

More precisely, $\Phi \leq \Phi' \Leftrightarrow$ there exists a one-to-one mapping $i:\mathrm{Dom}(\Phi) \to \mathrm{Dom}(\Phi')$ such that $\Phi' = i \circ \Phi$.

In this context, Theorem 4.4.2(iii) may be understood as stating that $\mathbf{E}_n^\infty(A)$ is an algebroidal category (hence an initial ω-complete category).

We are now in a position to study the relationship between $\mathbf{eval}_\infty$ and $\mathbf{eval}_{\mathcal{G}}$. The following lemma is routine checking.

4.4.5. Lemma. *The mapping* $\mathbf{eval}_{\mathcal{G}}$ *defines a ranked functor from* $\mathbf{E}(A)$ *to* $\underline{\mathfrak{CG}}(A)$. *The mapping* $\mathbf{eval}_\infty$ *defines an ω-continuous ranked functor from* $\mathbf{E}^\infty(A)$ *to* $\underline{\mathfrak{CG}}^\infty(A)$.

4.4.6. Proposition. *If Φ is finite then* $\mathbf{eval}_\infty(\Phi) = \mathbf{eval}_{\mathcal{G}}(\Phi)$.

Proof. The proof is by induction on the structure of the expression Φ. Since the proofs in all three cases are quite similar, we shall merely deal with one case: $\Phi_1 = \sigma_{\alpha_1,n_1,n_2}(\Phi_2)$.

From the definition of $\mathbf{eval}_{\mathcal{G}}$ we know that

$$\mathbf{eval}_{\mathcal{G}}(\Phi_1) = \mathbf{eval}_{\mathcal{G}}(\sigma_{\alpha_1,n_1,n_2})(\mathbf{eval}_{\mathcal{G}}(\Phi_2))$$

and that $\mathbf{eval}_{\mathcal{G}}(\sigma_{\alpha_1,n_1,n_2})$ evaluates to σ_{α_1}.

The induction hypothesis is that $\mathbf{eval}_{\mathcal{G}}$ and $\mathbf{eval}_\infty$ coincide on the subterm Φ_2. Let G_i be the element of the class of $\mathbf{eval}_\infty(\Phi_i)$ defined in the previous proposition (for $i = 1, 2$). An isomorphism of n-hypergraphs $G_1 \simeq \sigma_{\alpha_1}(G_2)$ is built in the following way:

- the hyperedge component is merely the bijection sending u to $0u$ whenever u is a hyperedge-generating leaf of the expression,

- from the construction of the previous proposition, one has $V_1 = \{0; 1, \ldots, 0; n_1\} \cup 0V_2$ where $0V_2 = \{0u; i \mid u; i \in V\}_2$ is isomorphic to V_2.
  Since the equivalence relation δ_2 is generated by the set
  $$\{(0u; i, 0v; j) \mid (u; i, v; j) \in \delta_i\} \cup \{(0; i, 00; j) \mid \alpha_1(i) = j\},$$

it follows that there is a bijection between V_{G_1} and V_{G_2} which is the required vertex component. □

4.4.7. Theorem. $\mathbf{eval}_\infty$ *is the unique ω-continuous extension of* $\mathbf{eval}_{\mathfrak{G}}$.

Proof. The theorem follows from Proposition 4.4.4 and Theorem 2.7.2 since $\mathbf{E}_n^\infty(A)$ is algebroidal and $\underline{\mathfrak{CG}}_n^\infty(A)$ is ω-complete. □

4.5. Approximation in terms of hypergraph expressions

Let us keep the same notations as in Section 3.8. We shall now build an increasing sequence of finite hypergraph expressions φ_i whose terms denote the G_i of Section 3.8 and whose ω-limit will denote G.

4.5.1. Definitions

Let the $\mathscr{D}$ be the tree domain decomposed as
$$\mathscr{D} = (001)^* + (001)^*0 + (001)^*00 + (001)^*000 = (001)^*(0)^{3*}.$$

A finite hypergraph expression with type n is in *standard form with height* $N \geq 1$ if:

- its domain is a prefix closed subset of the tree domain $\mathscr{D}$,
- for each integer k with $0 \leq k \leq N-1$ there exists some $\Psi_k = (\alpha_k, \delta_k, n_k, p_k, \boldsymbol{a}_k)$ such that
  $$(001)^k \mapsto \sigma_{\alpha_k, q_{k-1}, q_k + p_k},$$
  $$(001)^k0 \mapsto \Theta_{\delta_k, q_k + p_k},$$
  $$(001)^k00 \mapsto \oplus_{p_k, q_k},$$
  $$(001)^k000 \mapsto \boldsymbol{a}_k \quad (\text{with } \tau(a_k) = p_k,\ a_k \in A \cup \{1\}),$$
- $(001)^N \mapsto q_N$ with p_k, q_k integers, $\alpha_k : [q_k] \to [q_k + p_k]$, $\delta_k \in \mathrm{Equiv}[q_k + p_k]$, and $q_N \in \mathbb{N}$.
  Of course, an expression in standard form is locally finite.

4.5.2. Proposition. *With each n-hypergraph G, we can associate an increasing sequence $(\varphi_i)_{i \in \mathbb{N}}$ of hypergraph expressions of type n, in standard form of height i whose ω-limit Φ denotes G. If the hypergraph is finite, the sequence is constant for n greater than some sufficiently large N_0; if the hypergraph is effectively given, the sequence itself is effectively given.*

Proof. The construction of the sequence proceeds as follows using the notations of Section 4.5.1:

- let us first set $\varphi_0 = \boldsymbol{n}$ which denotes $G_0 = \mathord{>}n$,
- let a_1 be the label of the hyperedge e_1 of G and let the hypergraph G'_1 with type $q_1 = n + p_1$ be the context of G_1 in G. We let $\Psi_1 = (\alpha_1, \alpha_1, \delta_1, q_1, p_1, \boldsymbol{a}_1)$ be such that $G = \sigma_{\alpha_1}\Theta_{\delta_1}(\underline{a}_1 \oplus (\sigma_{\alpha_1}, (\underline{1} \oplus G'_1))$, and we set

$$\varphi_1 = \begin{tikzcd}[column sep=small]
 & \sigma_{\alpha_1, n, p_1+q_1-1} \arrow[d] & & \\
 & \Theta_{\delta_1, q_1+p_1-1} \arrow[d] & & \\
 & \oplus_{p_1-1, q_1} \arrow[dl] \arrow[dr] & & \\
a_1 & & \sigma_{\alpha'_1, q_1, q_1+1} \arrow[d] & \\
 & & \oplus_{1, q_1} \arrow[dl] \arrow[dr] & \\
 & 1 & & q_1
\end{tikzcd}$$

which we abbreviate as

$$\begin{tikzcd}[column sep=small]
 & \oplus_{\Psi_1} \arrow[dl, dash] \arrow[dr, dash] & \\
a_1 & & q_1
\end{tikzcd}$$

We now define φ_1 inductively to be

$$\varphi_i = \varphi_{i-1}\left[\begin{tikzcd}[column sep=small]
 & \oplus_{\Psi_i} \arrow[dl, dash] \arrow[dr, dash] & \\
a_i & & q_i
\end{tikzcd} \Big/ q_{i-1}\right]$$

i.e. the result of the substitution of

$$\begin{tikzcd}[column sep=small]
 & \oplus_{\Psi_i} \arrow[dl, dash] \arrow[dr, dash] & \\
a_i & & q_i
\end{tikzcd} \quad \text{for } q_{i-1} \text{ in } \varphi_1,$$

where $\Psi_i = (\alpha_i, \alpha'_i, \delta_i, q_i, p_i, a_i)$, $a = \mathrm{lab}(e_i)$, $p_i = \tau(a)_i$, $q_i = q_i + p_i$, for some mapping α_i and some equivalence relation δ_i and

$$\begin{tikzcd} & \oplus_{\Psi_i} \arrow[dl, dash] \arrow[dr, dash] & \\ a_i & & q_i \end{tikzcd}$$

is meant to abbreviate an expression similar to φ_1.

The proposition follows easily from this inductive definition, from Lemma 4.2.4 and from Theorem 4.4.5. $\square$

4.5.3. Proposition. *Let H be an n-hypergraph and H' be a finite sub-n-hypergraph of H. Let φ' be a hypergraph expression (in standard form) denoting H'. Then there exists a hypergraph expression Φ (in standard form) denoting H such that $\varphi' \leqslant \Phi$.*

Proof. It is a consequence of the previous proposition, Proposition 3.7.2 and Lemma 4.3.2. $\square$

It must be noticed that the converse is not true: if H' is a (not necessarily finite) sub-n-hypergraph of H and Φ an expression denoting H, there may be no sub-expression of Φ denoting H'. Indeed, we shall see in the next section that this is true only up to some rewriting of the expression.

4.5.4. Proposition. *The category $\mathbf{E}_n^\infty(A)$ is effectively given, and the functor* $\mathbf{eval}_\infty$ *is computable.*

Proof. Assuming that a recursive enumeration of A is given, an enumeration of the set of symbols on which expressions are built is easily defined. Finite hypergraph expressions can then be enumerated through the standard Gödel numbering. The proof of the rest of the proposition is lengthy but straightforward. $\square$

5. Equivalence of hypergraph expressions

5.1. Definition. Let $\alpha : [n] \to [p]$, $\alpha' : [n'] \to [p']$ denote two mappings and δ, δ' two equivalence relations on $[n]$ and $[n']$, respectively. We let:

- Δ be the trivial equivalence relation on $[n]$ for any $n \in \mathbb{N}$,
- $$\begin{aligned} \alpha + \alpha' : [n+n'] &\to [p+p'] \\ i &\mapsto \alpha(i), \quad \text{for } 1 \leqslant i \leqslant n \\ i &\mapsto p + \alpha'(i), \quad \text{for } n < i \leqslant n+n', \end{aligned}$$
- $\delta + \delta'$ be the equivalence relation on $[n+n']$ generated by

$$\delta \cup \{(n+i, n+j) \mid (i,j) \in \delta'\},$$

- $\alpha(\delta)$ denote the equivalence relation on $[p]$ generated by the pairs $(\alpha(i), \alpha(j))$ for $(i, j) \in \delta$.

5.2. Definition. Let $\mathcal{R}$ denote the following set of 12 equation schemes (where all types have been omitted for the sake of readability, and where $u, v, w, \ldots$ denotes hypergraph expressions of the appropriate types):

(R1) $u \oplus (v \oplus w) = (u \oplus v) \oplus w$,

(R2) $\sigma_\beta \sigma_\alpha(u) = \sigma_{\alpha \circ \beta}(u)$,

(R3) $\sigma_{\mathrm{Id}}(u) = u$,

(R4) $\Theta_\delta(\Theta_{\delta'}(u)) = \Theta_{\delta \cup \delta'}(u)$,

(R5) $\Theta_\Delta(u) = u$,

(R6) $\sigma_\alpha(u) \oplus \sigma_{\alpha'}(v) = \sigma_{\alpha + \alpha'}(v \oplus u)$,

(R7) $\Theta_\delta(u) \oplus \Theta_{\delta'}(v) = \Theta_{\delta + \delta'}(u \oplus v)$,

(R8) $\Theta_\delta(u \oplus \underline{1}) = \sigma_\alpha(\Theta_{\delta'}(u))$ where δ is the equivalence relation on $[n+1]$ such that $(i, n+1) \in \delta$ for some $i \leqslant n$, δ' is the restriction δ to $[n]$ and α from $[n+1]$ to $[n]$ sending j to j for $j \leqslant n$ and $n+1$ to i,

(R9) $\Theta_\delta(\sigma_\alpha(u)) = \sigma_\alpha(\Theta_{\alpha(\delta)}(u))$,

(R10) $\sigma_\alpha(\Theta_\delta(u)) = \sigma_\beta(\Theta_\delta(u))$ where $\alpha, \beta : [n] \to [p]$ and $(\alpha(i), \beta(i)) \in \delta$, $\forall i \in [p]$,

(R11) $u \oplus \underline{0} = u$,

(R12) $\underline{n} = \underline{1} \oplus \cdots \oplus \underline{1}$ (n times).

5.3. Proposition. *The set $\mathcal{R}$ of equation schemes is valid in $\mathfrak{G}^\infty(A)$.*

Proof. This result was stated for finite hypergraphs in [4]. It is no less obvious for infinite ones. As was noted in [4], we do not know whether this set of rules is minimal. □

5.4. Definition. Let Φ and Φ' be two hypergraph expressions. We say that they are *equivalent* and we write $\Phi \sim \Phi'$ if they denote the same hypergraph, i.e., if $\mathbf{eval}_\infty(\Phi)$ and $\mathbf{eval}_\infty(\Phi')$ are isomorphic.

5.5. Theorem (Bauderon and Courcelle [4]). *Two finite hypergraph expressions Φ and Φ' are equivalent if and only if they are congruent with respect to the rewriting system $\overset{*}{\underset{\mathcal{R}}{\leftrightarrow}}$ generated by the set $\mathcal{R}$ of equations. In other words, $\mathfrak{G}(A)$ and $\mathbf{E}(A)/\overset{*}{\underset{\mathcal{R}}{\leftrightarrow}}$ are isomorphic as many-sorted $\mathbb{H}_A$-magmas.*

In order to extend this result to the case of infinite expressions, we shall have to use approximating sequences of finite expressions. Hence, a slightly different notion of congruence will be needed.

5.6. Definition. Two hypergraph expressions Φ and Φ' are congruent with respect to $\mathcal{R}$ and we write $\Phi \equiv_{\mathcal{R}} \Phi'$ if and only if

$$(\forall \varphi \in \mathbf{E}(A) \mid \varphi \leqslant \Phi)(\exists \varphi' \in \mathbf{E}(A) \mid (\varphi' \leqslant \Phi', \varphi(\leqslant \cup \overset{*}{\underset{\mathcal{R}}{\leftrightarrow}})\varphi'))$$

and conversely.

5.7. Theorem. *Two hypergraph expressions are equivalent if and only if their reduced expressions are congruent.*

Proof. According to Lemma 4.3.4, two hypergraph expressions are equivalent if and only if their reduced expressions are. Therefore, we only need to prove that locally finite expressions are equivalent if and only if they are congruent. The proof will follow from two lemmas.

5.7.1. Lemma. *For each locally finite expression Φ, there exists an expression Φ^{s} in standard form such that $\Phi \sim \Phi^{s}$ and such that $\Phi \equiv_{\mathcal{R}} \Phi^{s}$.*

Proof. Let us first consider finite expressions. The proof is by induction on the structure of the expressions and we must distinguish three cases:

(i) $\varphi = \sigma_{\alpha}(\varphi_1)$ where $\varphi_1 = \sigma_{\beta}(\Theta_{\delta}(a \oplus \varphi_2))$ is in standard form. Then, by rule (R2), $\varphi = \sigma_{\alpha \circ \beta}(\Theta_{\delta}(a \oplus \varphi_2))$ which is in standard form.

(ii) $\varphi = \Theta_{\delta}(\varphi_1)$ with the same notation for φ_1. Then, by rules (R9) and (R4), φ may be rewritten into $\sigma_{\alpha}(\Theta_{\beta(\delta) \cup \delta}(\varphi_2))$ which is in standard form.

(iii) Let us now suppose that $\varphi = \varphi_1 \oplus \varphi_2$, where $\varphi_i = \sigma_{\beta_i}(\Theta_{\delta_i}(\boldsymbol{a}_i \oplus \varphi_i'))$ for $i = 1, 2$.

$$\begin{aligned}
\varphi &= \sigma_{\beta_1}(\Theta_{\delta_1}(\boldsymbol{a}_1 \oplus \varphi_1')) \oplus \sigma_{\beta_2}(\Theta_{\delta_2}(\boldsymbol{a}_2 \oplus \varphi_2')) & \\
&\leftrightarrow \sigma_{\beta}(\Theta_{\delta_1}(\boldsymbol{a}_1 \oplus \varphi_1') \oplus \Theta_{\delta_2}(\boldsymbol{a}_2 \oplus \varphi_2')) & \text{by (R6)}, \\
&\leftrightarrow \sigma_{\beta_1}(\Theta_{\delta_1 + \delta_2}(\boldsymbol{a}_1 \oplus \varphi_1') \oplus (\boldsymbol{a}_2 \oplus \varphi_2')) & \text{by (R7)}, \\
&\leftrightarrow \sigma_{\beta_1}(\Theta_{\delta_1 + \delta_2}(\boldsymbol{a}_1 \oplus \sigma_{\alpha} \Theta_{\Delta}(\boldsymbol{a}_2 \oplus (\varphi_1' \oplus \varphi_2')))) & \text{by (R6) and (R5)}.
\end{aligned}$$

In each case, it follows by induction on the height of the expression that the result is true for any finite expression.

Now let Φ be an infinite hypergraph expression and let $(\varphi_i)_{i \in \mathbb{N}}$ be an increasing sequence of finite approximants of Φ. If $i \leqslant j$, then $\varphi_i \leqslant \varphi_j$ and there is a monomorphism $\mathbf{eval}_{\infty}(\varphi_i) \to \mathbf{eval}_{\infty}(\varphi_j)$. Let φ_i^{s} be an expression in standard form obtained by rewriting the finite expression φ_i. From Proposition 4.5.3 there is an expression in standard form φ_i^{s} equivalent to φ_i and such that $\varphi_i^{s} \leqslant \varphi_j^{s}$. Now, since both φ_j and φ_j^{s} are finite and denote the same hypergraph they are congruent (by Theorem 5.5).

Hence, from the increasing sequence $(\varphi_i)_{i \in \mathbb{N}}$ we can construct an increasing sequence $(\varphi_i^{s})_{i \in \mathbb{N}}$ of expressions in standard form, such that $\mathbf{eval}_{\infty}(\varphi_i) = \mathbf{eval}_{\infty}(\varphi_i^{s})$ for $i \in \mathbb{N}$. Let Φ^{s} be its ω-limit. It follows from the ω-continuity of the functor $\mathbf{eval}_{\infty}$ that Φ^{s} is equivalent to Φ. $\square$

5.7.2. Lemma. *Two expressions in standard form are equivalent if and only if they are congruent.*

Proof. ($\Rightarrow$): Let Φ and Φ' be two equivalent expressions in standard form and let φ be a finite approximant of Φ. Then there exists some $\varphi_1 \leqslant \Phi'$ such that $\mathbf{eval}_\infty(\varphi)$ is a subhypergraph of $\mathbf{eval}_\infty(\varphi_1)$.

Hence, φ is the "beginning" of a standard form approximation φ' of $\mathbf{eval}_\infty(\varphi_1)$. Now, since φ' and φ_1 are equivalent and both finite, it follows from Theorem 5.5 that φ' may be rewritten into φ_1 in a finite number of steps, hence that $\varphi'(\leqslant \cup \underset{\mathcal{R}}{\overset{*}{\Leftrightarrow}})\varphi_1$.

Since this is true in the other direction as well, the first implication is proved.

($\Leftarrow$): Let us now assume that $\Phi \equiv_{\mathcal{R}} \Phi'$ and let us set $G = \mathbf{eval}_\infty(\Phi)$, $G' = \mathbf{eval}_\infty(\Phi')$. Let φ be an approximant of Φ in standard form. From the definition, there exists some $\varphi' \leqslant \Phi'$ such that $\varphi\ (\leqslant \cup \underset{\mathcal{R}}{\overset{*}{\Leftrightarrow}})\varphi'$, i.e. some φ'' such that $\varphi \underset{\mathcal{R}}{\overset{*}{\Leftrightarrow}} \varphi''$ and $\varphi'' \leqslant \varphi'$. This defines a monomorphism

$$\mathbf{eval}_\infty(\varphi) \to \mathbf{eval}_\infty(\varphi') = \mathbf{eval}_\infty(\varphi),$$

hence a monomorphism $\mathbf{eval}_\infty(\varphi) \to \mathbf{eval}_\infty(\Phi') = G'$.

Let now $(\varphi_i)_{i \in \mathbb{N}}$ be an ω-diagram of approximants in standard form with ω-limit Φ. Since the functor $\mathbf{eval}_\infty$ is ω-continuous, it transforms this ω-diagram into an ω-diagram $(\mathbf{eval}_\infty(\varphi_i)_{i \in \mathbb{N}})$ whose ω-limit is $G = \mathbf{eval}_\infty(\Phi)$.

For every i, $i \in \mathbb{N}$, we may construct a monomorphism from $\mathbf{eval}_\infty(\varphi_i)$ into G', i.e. by the ω-limit property of G, a monomorphism of G into G'. Now, if this monomorphism were not onto, there would be some item (hyperedge or vertex) of G' which would not appear in G, and the hypergraph expression could not be congruent to G. $\square$

5.7.3. Remarks. (i) For genuinely infinite expressions, the proof of the theorem is obviously not effective. This is mainly because of the general quantification $\forall \varphi \in \mathrm{E}(A)$ in Definition 5.6 which requires the property to be checked for all φ. Intuitively, a slight change could be brought, replacing the $\forall \varphi \in \mathrm{E}(A)$ by a more restricted quantification, which would stop checking for some large enough expression. This would amount to saying that two expressions are indistinguable if they cannot be distinguished in "some reasonable manner". We shall not develop this idea any further.

(ii) It is shown in [8, Proposition 5.4] that a similar result does not hold for infinite words when they are described by means of "arrangements". More precisely, with such an infinite word a syntactic tree is associated and a congruence is defined on the corresponding magma to identify two trees denoting the same arrangement. It is then shown that the congruence of infinite trees cannot be described in terms of the congruence of its finite approximants, i.e. that the property of "algebraicity" or if we dare say of "algebroidality" is missing.

This fact stems from Courcelle's definition of an isomorphism between arrangements which is much finer than our notion of sourced hypergraph isomorphism. Indeed, when interpreted in our setting, the two arrangements $a^{\omega}a^{\omega}a^{-\omega}$ and $a^{\omega}a^{-\omega}a^{-\omega}$ (given in [8] as a counter-example) would appear to be isomorphic.

In [13], Dauchet and Timmerman have shown a result for infinite words which is similar to ours, thanks to a notion of "yield of an infinite tree", which is coarser than Courcelle's notion of an arrangement. Related works are [19] and [33].

6. Conclusion

In this first part we have described at length a fairly wide range of tools which will help us to study infinite hypergraphs:

- on their own,
- as limits of sequences of finite graphs,
- as denotations of some algebraic expressions.

All these tools will be used in part two in order to solve systems of recursive equations on hypergraphs.

References

[1] J. Adamek and V. Koubek, Least fixed points of a functor, *J. Comput. System Sci.* **19** (1979) 163-178.

[2] M. Bauderon, On some properties of infinite graphs, in: *Proc. WG'88*, Amsterdam, Lecture Notes in Computer Science **344** (Springer, Berlin, 1989) 54-73.

[3] M. Bauderon, Infinite hypergraphs, II. Systems of recursive equations, *Theor. Comput. Sci.*, to appear.

[4] M. Bauderon and B. Courcelle, Graph expressions and graph rewritings, *Math. Systems Theory* **20** (1987) 83-127.

[5] D. Caucal, Pattern graphs, Research report 441, IRISA, 1988.

[6] T. Coquand, Categories of embedding, *Theoret. Comput. Sci.* **68** (1989) 221-237.

[7] T. Coquand and T. Ehrart, An equational presentation of higher-order logic, in: *Category Theory and Computer Science*, Edinburgh, Lecture Notes in Computer Science **283** (Springer, Berlin, 1987) 40-56.

[8] B. Courcelle, Frontiers of infinite trees, *RAIRO Inform. Théor.* **12** (4) (1978) 319-337.

[9] B. Courcelle, Fundamental properties of trees, *Theoret. Comput. Sci.* **25** (1983) 95-169.

[10] B. Courcelle, The monadic 2nd order logic of graphs IV: definability properties of equational graphs, *Ann. Pure Appl. Logic* (1990) 193-255.

[11] B. Courcelle, The monadic 2nd order logic of graphs III: tree-decomposition, forbidden minors and complexity issues, to appear in *RAIRO Inform. Théor.*

[12] B. Courcelle, Graph rewriting: an algebraic and logic approach, in: J. van Leeuwen, ed., *Handbook of Theoretical Computer Science, Vol. B* (Elsevier, Amsterdam, 1990) 193-242.

[13] M. Dauchet and E. Timmerman, Continuous monoids and yields of infinite trees, *RAIRO Inform. Théor.* **20** (3) (1986) 251-274.

[14] H. Ehrig, *Introduction to the Algebraic Theory of Graphs*, Lecture Notes in Computer Science **73** (Springer, Berlin, 1977) 1-69.

[15] H. Ehrig and B. Mahr, Fundamentals of algebraic specifications, in: *EATCS Monographs in Theoretical Computer Science* (Springer, Berlin, 1985).

[16] J.A. Goguen and R.M. Burstall, Some fundamental algebraic tools for the semantics of computation, Theoret. Comput. Sci. **31** (1984) 175-209, 263-295.

[17] A. Habel, Hyperedge replacement grammars and language, Thesis, Bremen 1989.

[18] A. Habel and H-J. Kreowsky, *Some Structural Aspects of Hypergraph Languages Generated by Hyperedge Replacement*, Lecture Notes in Computer Science **247** (Springer, Berlin, 1987) 207-219; May we introduce to you: hyperedge replacement, Lecture Notes in Computer Science **291** (Springer, Berlin, 1987) 15-26.

[19] S. Heilbrunner, An algorithm for the solution of fixed point equations for infinite words *RAIRO Inform. Théor.* **14** (1980) 131-141.

[20] A. Kanda, Fully effective solutions of recursive domain equations, in: *MFCS 1979*, Lecture Notes in Computer Science **74** (Springer, Berlin, 1979) 326-336.

[21] J. Lambek, A fixpoint theorem for complete category, *Math. Z.* **103** (1968) 151-161.

[22] J. Lambek, *Fixpoint revisited, Logik at Botik*, Lecture Notes in Computer Science **363** (Springer, Berlin, 1989) 200-207.

[23] D.J. Lehmann, Categories for fixpoint semantics, in: *17th Symp. on Foundations of Computer Science*, Houston (1976) 122-126.

[24] D.J. Lehmann and M.B. Smyth, Algebraic specification of data types, *Math. Systems Theory* **14** (1981) 97-139.

[25] A. Lindenmayer, Models for multicellular development: characterization, inference and complexity of L-systems, in: *Trends, Techniques and Problems in Theoretical Computer Science*, Lecture Notes in Computer Science **281** (Springer, Berlin, 1987) 138-168.

[26] E. Manes, *Algebraic Theories* (Springer, Berlin, 1976).

[27] E. Manes and M. Arbib, *Algebraic Approach to Program Semantics*, The AKM Series in Theoretical Computer Science (Springer, Berlin, 1986).

[28] S. McLane, *Category for the Working Mathematician* (Springer, Berlin, 1971).

[29] D. Muller and P. Schupp, The theory of ends, pushdown automata and second-order logic, *Theoret. Comput. Sci.* **37** (1985) 51-75.

[30] M.B. Smyth, Powerdomains, *J. Comput. Systems Sci.* **16** (1978) 23-36.

[31] M.B. Smyth, Computability in categories, in: *Proc. 7th ICALP*, Lecture Notes in Computer Science **85** (Springer, Berlin, 1980) 609-620.

[32] M.B. Smyth and G. Plotkin, The category theoretic solution to recursive domain equations, *SIAM J. Comput.* **11** (4) (1982).

[33] W. Thomas, On frontier of regular trees, *RAIRO Inform. Théor.* **20** (4) (1986) 371-381.

[34] K. Weihrauch, Computability, in: *EATCS Monographs in Theoretical Computer Science* (Springer, Berlin, 1986).