Faisceaux et Sémantique des Programmes

by

Areski Nait-Abdallah

Technical Report CS-82-08

Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada
N2L 3G1

Cebbaγ w'ur nekkat uzzal
Ycmet waggus is
Am'win irefden uffal
Yerra t d leslah is
Neγ afsih deg lmital
Ur nessefruy seg ixf is.

Yusef U'Qasi

" So you've got to the end of our race course ? " said the Tortoise. " Even though it does consist of an infinite series of distances ? I thought some wiseacre or other proved that the thing could'nt be done ? "

" It can be done " said Achilles." It has been done ! Solvitur ambulando. You see the distances were constantly diminishing, and so ... "

Lewis Carroll

Remerciements

Je remercie Messieurs Louis Nolin, Maurice Nivat, Dana Scott, Jean-Jacques Lévy, Bernard Robinet, Gilles Ruggiu et Youssef Mentalechta de m'avoir fait l'honneur de constituer mon jury de thèse.
Ma dette est grande à l'égard de Karl H. Hofmann, Dana Scott et Louis Nolin, dont les commentaires fructueux ont permis d'améliorer grandement le contenu de ce travail.
Merci à mes amis Marcel Duboué et Raymond (Titouh) Durand pour leur aide appréciée.

Abstract : This work is devoted to the study of the notion of convergence in programming languages and λ-calculus semantics models. We formalize this notion, and several of its uses in connection with Nolin's, Scott's, Plotkin's ... constructions are examined. It is shown that bundle theory provides a general, and conceptually simple frame, for a local study of these models. The global approach is also described in this frame.

This work is an attempt to contribute to a unified theory of programming languages semantics models (*).

Résumé : Ce mémoire est consacré à l'étude de la notion de convergence dans les modèles de sémantique des langages de programmation et du λ-calcul. On formalise cette notion, et plusieurs de ses utilisations en liaison avec les constructions de Nolin, Scott, Plotkin, ... sont examinées. On montre que la théorie des faisceaux fournit un cadre général et conceptuellement simple pour une étude locale de ces modèles. L'approche globale est aussi décrite dans ce cadre.

Ce travail est une contribution à une théorie unifiée des modèles de sémantique des langages de programmation.

(*) An extended abstract is given further on.

TABLE DES MATIERES

SUMMARY

We consider the notions of approximation which have been defined by D. Scott 1969 (continuous lattices), L. Nolin 1974 (algorithm collections), K.H. Hofmann 1978 (approximating orders), A. Shamir and W. Wadge 1977 (type structure over a domain), ADJ 1977 (inductive posets), A. Arnold and M. Nivat 1978 (metric magma) etc..., and we outline a general frame for classifying them and studying their properties. Our intention here is to contribute to a unified programming language semantics theory.

I. BUNDLES :

The first notion of approximation which is considered is the one which allows us to obtain points as limits of other points.

I.1. NOTION OF A BUNDLE :

A von Neumann universe is a set $\mathcal{U}$ ("constructed from the empty set") such that :

(i) $x \in a \in \mathcal{U} \Rightarrow x \in \mathcal{U}$

(ii) $a \in \mathcal{U}$ and $b \in \mathcal{U} \Rightarrow \{a,b\}$, $\langle a,b \rangle$, $a \times b \in \mathcal{U}$

(iii) $x \in \mathcal{U} \Rightarrow \mathcal{P}(x) \in \mathcal{U}$ and $\bigcup x \in \mathcal{U}$

(iv) $\omega \in \mathcal{U}$ (where $\omega = \{0,1,2,3,\ldots\}$ is the set of finite ordinals)

(v) If $f : a \to b$ is a surjective function with $a \in \mathcal{U}$ and $b \subset \mathcal{U}$, then $b \in \mathcal{U}$. □

Let $\mathcal{U}$ be a fixed von Neumann universe. Let X be a set. The set $\mathcal{F}(X)$ of families of elements of X is

$$\mathcal{F}(X) = \{f : I \to X : I \in \mathcal{U}\}$$

A bundle over X is a couple $\langle \lambda, \beta \rangle$ with

⊙ $\lambda : \mathcal{F}(X) \to X$ partial surjective (sometimes denoted lim)

$$\lambda : f \to \lambda(f) = \lambda_{i \in I} . f(i) = \lim_{i \in I} f(i)$$

if $f : I \to X$

⊙ $\beta : X \to \mathcal{P}(\mathcal{F}(X))$ injective

λ satisfies the Fubini property on the one hand :

$\forall\ g : I \times J \to X \ , \ g \in \mathcal{F}(X)$

$\lambda_{i \in I}\ (\lambda_{i \in I} \cdot g(i,j))$ exists) $\Rightarrow (\lambda_{(i,j) \in I \times J} \cdot g(i,j))$ exists also and $\lambda_{i \in I} \cdot (\lambda_{j \in J} \cdot g(i,j)) = \lambda_{(i,j) \in I \times J} \cdot g(i,j)$ (resp. $\lambda_{j \in J} \cdot (\lambda_{i \in I} \cdot g(i,j)) = \lambda_{(i,j) \in I \times J} \cdot g(i,j)$)

and the point reconstruction property on the other hand :

$$\forall x \in X \quad \forall \sigma \in \beta(x) \qquad x = \lambda(\sigma)$$ □

The Scott topology on a poset, Nolin algorithm collections, type structures over a domain (Shamir and Wadge), the metric magma (Arnold and Nivat), the convergence notion which is used in Manna and Shamir 1977, the convergence notion in Kuratowski-Painlevé 's sens, the way λ-calculus is used in programming languages formal semantics, define bundle structures.

Bundles are closed under product and coproduct, and the notions of subbundle and quotient bundle can also be defined in the usual way.

I.2. BUNDLE CLASSES

Let X be a set which is supplied with a bundle structure $<\lambda,\beta>$. The set

$$N(X) = N_{\lambda,\beta}(X) = \{ u \in X : \exists x \in X, \ \exists \sigma \in \beta(x) , \ \sigma : I \to X , \quad \exists i \in I , \quad u = \sigma(i) \}$$

is called the Kernel of the bundle. An element $u \in X$ is rational iff

$$\forall \sigma \in \beta(u) , \quad \sigma : I \to X , \quad \exists i \in I \quad u = \sigma(i).$$

The bundle $<\lambda,\beta>$ is elementary iff its Kernel contains only rational elements, i.e.

$$\forall x \in X \quad \forall \sigma \in \beta(x) , \quad \sigma : I \to X \ , \quad \forall i \in I , \quad \sigma(i) \text{ is rational.}$$

The bundle $<\lambda,\beta>$ is ordered iff X is partially ordered and λ takes least upper bounds (resp. greatest lower bounds) of families of elements.

The bundle $<\lambda,\beta>$ is <u>ordered monic</u> (in the <u>restricted sense</u>) iff there is a monotone function st

$$\forall x \in X \qquad \forall \sigma \in \beta(x) \qquad \sigma = id_{s(x)} : s(x) \to s(x).$$

For any x, $s(x)$ is then called the <u>spectrum</u> of x ; and we denote $a < b$ iff $a \in s(b)$ (a <u>approximates</u> b).

<u>PROPOSITION 1.1.</u> : The ordered monic bundles (in the restricted sense), whose spectra are lowersets of the Kernel are exactly the Galois connections

$$X \underset{s}{\overset{\lambda^*}{\leftrightarrows}} T \subseteq \mathcal{P}(X)$$

such that $\lambda^* \circ s = id_X$.

<u>I.3. REGULAR FUNCTIONS</u> :

Let X be a set supplied with a bundle $<\lambda,\beta>$, Y supplied with a bundle $<\lambda',\beta'>$. A function $f : X \to Y$ is <u>regular at point</u> <u>$x \in X$</u> iff

$$\forall \sigma \in \beta(x) \qquad f(x) = \lambda'(f(\sigma))$$

where $f(\sigma)$ is the image family $f \circ \sigma : I \to Y$ of $\sigma : I \to X$

$$\begin{tikzcd} I \arrow[d] \arrow[dr, "f \circ \sigma"] & \\ X \arrow[r, "f"'] & Y \end{tikzcd} \quad \text{denoted by } f(\sigma)$$

Continuous functions in Scott's sense, algorithms in Nolin's sense , Shamir and Wadge's "tight" functions, Manna and Shamir 1977's continuous functions, ... are regular functions.

The identity and constant functions are always regular. The projections of a bundle product are regular

A function of many variables is regular iff it is regular for each of its variables (Prop. 1.4).

One denote $[X \to Y]$ the set of regular functions from X to Y , and one supplies it with the pointwise (extensional) order whenever Y is partially ordered.

If X and Y are bundles, a bundle on the space $[X \to Y]$ of regular functions from X to Y is faithful iff

$$\forall f \in [X \to Y] \qquad \forall \{f_i\}_{i \in I} \in \beta(f)$$
$$(\lim_i{}_{[X \to Y]} f_i) = f \Rightarrow \forall x \in X \qquad \lim_i{}_Y f_i(x) = f(x).$$

If the functions spaces are supplied with faithful bundles, then abstraction is a weakly regular function ($g : A \to B$ is weakly regular iff $\forall x \in A \quad \forall \sigma \in \beta(x) \quad \lim_B(f(\sigma))$ exists $\Rightarrow f(x) = \lim_B(f(\sigma))$). (Prop. 1.5). But if function spaces are supplied with faithful bundles, then the application (or evaluation) is always regular.

II - ORDERED BUNDLES :

Any elementary ordered bundle defines an elementary order (cf. Naït-Abdallah 1978) and reciprocally. Any algebraic epo has an elementary o.m.b. structure (2.8).

If s_1, s_2 are functions from X to $\mathcal{P}(X)$, which define o.m.b. on X, then the omb defined by s_1 is finer than the omb defined by s_2 iff $\forall x \in X \quad s_1(x) \subseteq s_2(x)$. If X is a poset, and if we order the set of omb's (in the restricted sense) by the relation "being finer than", than this set has a complete semi-lattice structure whose bottom is defined by

$$s : x \mapsto \downarrow x = \{ y \in X : y \sqsubseteq x \}$$

(Prop. 2.11).

A bundle $<\lambda, \beta>$ on a set X is interpolable at point x iff $\forall e \in X$ we have $\forall \sigma \in \beta(x)$
$(\exists i \in Dom(\sigma) \quad e = \sigma(i)) \Leftrightarrow \quad [\exists k \in Dom(\sigma), \quad u = \sigma(k), \quad \forall \tau \in \beta(u)$
$\exists j \in Dom(\tau) \quad e = \tau(j)]$.

In the ordered monic (restricted) case this reduces to

$$e \in s(x) \Leftrightarrow \exists u \in s(x) \quad e \in s(u).$$

Continuous Lattices, elementary (ordered) bundles are interpolable at each point. (Ordered) bundles are interpolable at each rational point.

An ordered bundle is <u>stable at point $x \in X$</u> iff $\forall y \in X$, $\forall a \in N(X)$ (Kernel of X) $a \sqsubseteq x \sqsubseteq y$ implies the equivalence of the following assertions

(i) $\exists\, \sigma \in \beta(x) \quad \exists\, i \in \mathrm{Dom}(\sigma) \quad a = \sigma(i)$

(ii) $\exists\, \tau \in \beta(y) \quad \exists\, j \in \mathrm{Dom}(\tau) \quad a = \tau(j)$

In the monic case this reduces to $\forall y \in X \quad x \sqsubseteq y \Rightarrow s(x) = ({\downarrow}x) \cap s(y)$. The bundle SW is stable ; the bundle N is stable iff its rational element cannot be compared between themselves.

It turns out that any omb which is a Galois connection is stable; any omb on a directed space X, which is not a Galois connection, has an unstability point (lemma 2.16). As far as omb are concerned, stability and interpolability are related in the following way:

<u>LEMMA 2.17.</u> : Let X a poset supplied with an omb , $x \in X$, and
$${\uparrow}x = \{ z \in X \ : x \in s(z) \} .$$
If the spectra of the $y \in {\uparrow}x$ are directed and if the omb is stable over ${\uparrow}x$, then the omb is interpolable at point x.

The notion of continuous lattice in Scott's sense fits nicely in this frame :

<u>PROPOSITION 2.18.</u> : Let X be a poset, $\mathcal{P}(X)$ its set of subsets ordered by inclusion, both being supplied with their Scott topology, $s : X \to \mathcal{P}(X)$ defining an omb on X.
Then

(i) If $s = s_i : x \mapsto \cap \{ I \in I_{cv} X : x \sqsubseteq \sqcup I \}$ and if the bundle is interpolable, then s is continuous at each point for the Scott topologies of X and $\mathcal{P}(X)$

(ii) If s is continuous at each point for the Scott topologies of X and $\mathcal{P}(X)$, then $\forall x \in X \quad s(x) \sqsubseteq \cap \{ I \in Id_{cv} X : x \sqsubseteq \sqcup I \}$ $\square$

III - REGULAR FUNCTIONS SPACES :

Any function which is regular over an omb is monotonic . If X is a (restricted) omb, Y a complete sup-semilattice, then any regular function $f : N(X) \to Y$ has a unique regular extension (3.5).

Regularity for the composition of two regular functions requires special conditions (3.7), but we can define the following "regular composition" :

$$O : N[X \to Y] \times N[Y \to Z] \times N(X) \to Z$$

$$O = \lambda f \in N[X \to Y].\ \lambda g \in N[Y \to Z].\ \lambda u \in N(X).g(f(u))$$

which generalizes the usual composition, and handles better the peculiar bundle structures one considers.Furthermore, if $[X \to Y]$, $[Y \to Z]$ are faithful and elementary omb's, if X is an elementary omb, if Z is a conditionnally complete poset, then the mapping O is regular over the space $[X \to Y] \times [Y \to Z] \times X$ supplied with the product bundle (Prop. 3.8). The associativity of O works in special cases (3.9).

III.1. BUNDLES ON REGULAR FUNCTIONS SPACES :

If X and Y are ordered bundles such that Y is normalized, the pointwise convergence bundle on $[X \to Y]$, when it exists, is defined by the relation :

$$f < g \iff (\forall x \in X \quad f(x) < g(x)) \iff (\forall x \in X \quad f(x) \in s(g(x))$$

The following problem appears then at once : which bundle structure (if any) is to be taken on the regular functions space ? A number of methods for solving this problem, in some cases which are relevant in computer science, are examined : lower threshold functions (Scott), upper threshold functions (Nolin), and codomain space continuity (Scott). Plotkin's and Smyth's powerdomain construction, considered from this point of view, takes its full prominence and relates smoothly to Arnold's and Nivat's denotational semantics.

If D and D' are ordered bundles, $e \in D$, $e' \in D'$, $m \in D'$, we define : the lower (or Scott) threshold function $[e,e']_m$ (with $m \leq e'$) by :

$$[e,e']_m : x \mapsto e' \text{ if } \quad \forall \sigma \in \beta(x)\ \exists i \in Dom(\sigma) \quad e = \sigma(i),\ m \text{ otherwise.}$$

and the upper (or Nolin) threshold function

$\langle e,e' \rangle_m$ (with $e' \sqsubseteq m$) by $\langle e,e' \rangle_m : x \mapsto e'$ if $x \sqsubseteq e$, m otherwise.

Every upper threshold function is regular.

Lower threshold functions method :

a) Monic case : If D and D' are monic, every regular function $f : D \to D'$, which has a lower bound, is the lub of some family of Scott threshold functions (3.12), and a lower threshold function. $[e,e']_m$ is regular iff D is interpolable at point e. Whence, if D is interpolable, a bundle structure on $[D \to D']_*$, whose approximating elements are the lower threshold functions, if

$[D \to D']_* = \{f \in [D \to D'] \mid f$ has a lower bound $\}$

By the same way, if we take

$[D \to D']_{**} = \{f \in [D \to D'] \mid \exists\ m(f) \in D'$ st $\forall x \in D\ \ m(f) < f(x)$,

we obtain a pointwise convergence bundle on $[D \to D']_{**}$ (Corollary 3.14).

This construction can be iterated in the following conditions :

PROPOSITION 3.16. : Let D, D' be omb's, suppose D is interpolable and let $[D \to D'^*]$ as above. Let us supply $[D \to D']_{**}$ with the pointwise convergence bundle, and suppose furthermore that D' verifies :

(i) it is stable

(ii) its spectra are supsemillatices.

Then the omb $[D \to D']_{**}$ is interpolable and verifies the above two conditions (i) and (ii).

In fact if D is elementary (which is a special case of interpolability), and in the same conditions as for corollary 3.14, some canonicity conditions can be imposed to the decomposition in Scott t.f. so as to obtain some kind of a "basis" (Prop. 3.17).

b) General case : The method is deduced from the monic case. If D and D' are ordered bundles the ltf $[e,e']_m$ is regular iff D is interpolable at point e (3.18). If the fibre $\mathfrak{z}(x)$ of each $x \in D$ has a smallest element in the extension sense (3.19), then every regular function $f : D \to D'$ which has a lower bound is the least upper bound of some family of Scott threshold functions.

The notion of a stable ordered bundle can be used in an analogous way : If D and D' are ordered bundles such that D' is stable every regular monotone function $f : D \to D'$ which has a lower bound is the lub of some family of ltf's.

Whence a bundle structure on this class of regular functions if we take D interpolable (Proposition 3.21).

We obtain thus a monic bundle on the space of functions, which leads us to the preceding case if we want to iterate.

Codomain space continuity method.:

a) Monic case : If D and D' are two interpolable omb, where D' is continuous in Scott's sense and contains $\perp$, then the space $[D \to D']$ has a continuous bundle structure and contains $\perp$.

b) General case : In the general case appears an other way for generalizing the interpolability notion we have on the omb : we call it global interpolability.

A bundle on a poset X is globally interpolable at point $e \in X$ iff for any $u \in X$ we have :

$$\forall \tau \in \beta(u) \quad \exists k \in Dom(\tau) \quad e = \tau(k) \Leftrightarrow$$

$$\exists x \in X(\forall \sigma \in \beta(x) \quad \exists i \in Dom(\sigma) \quad e = \sigma(i)) \wedge \forall \tau \in \beta(u) \quad \exists j \in Dom(\tau) \quad x = \tau(j)$$

We have then :

PROPOSITION 3.24. : Let D be an interpolable and globally interpolable ordered bundle, D' be a continuous ordered bundle in Scott's sense, which is interpolable and contains $\perp$. Then the space $[D \to D']$ of monotone regular functions, when supplied with the poitwise ordering has a continuous bundle structure and contains $\perp$.

Upper threshold functions method :

If D and D' are two ordered bundles, $[D \to D]_{max}$ is the space of regular functions which are monotone and have an upper bound, then any function of $[D \to D']_{max}$ is the greatest lower bound of some family of upper threshold functions (3.27).

If D is elementary a more "canonical" definition of this family can by given (3.28).

III.2. SEMANTICS FOR A LANGUAGE WITH TYPE DECLARATIONS, AND RETRACTES :

It has been pointed out (cf. [NOL] , [SHA]) that types can be considered as calculus objects. We use the ordered bundle notion to precise and develop this point of view, and construct a greatest fixpoint semantics for programs with type declarations :

$$\text{Fab} : G(x) \Leftarrow \tau[G]\,(x)$$

Let us also call τ the functional which is associated (by regular composition) with the term $\tau[G]$.

PROPOSITION 3.39. : Let X be an elementary omb, which is distributive and conditionnally complete, $f_o \in [X \to X]$. Now, if the sequence $\{\tau^n(f_o)\}_{n \in \mathbb{N}}$ is convergent, on $D \subseteq X$, for the bundle Sinv, which is associated to the Scott topology for the opposite order of $[X \to Y]$, then $\bigsqcap_n \tau^n(f_o)$ is solution, over D , of the functionnal equation $g = \tau(g)$. If $f_o = \lambda x.T$, $\bigsqcap_n \tau^n(\lambda x.T)$ is the greatest fixpoint of over D . □

We call convenient domain any elementary omb X which is

(i) conditionally complete

(ii) distributive, i.e.

$$\forall a \in X \quad \forall S \subset X \quad a \sqcap (\sqcup\{s : s \in S\}) = \sqcup\{a \sqcap s : s \in S\}$$

(iii) has a T .

The following (functional) transformation

$$N_{a,b} : \tau \mapsto \lambda f.\,\lambda x.\, \langle a,b \sqcap \tau(f)(x) \rangle_T\,(x)$$

is associated to the type declaration in the above recursive definition. It describes the filtering which is done while verifying types in an ALGOL or FORTRAN-like language.

If X is a convenient domain, if $[X \to X]$ is supplied with a faithful bundle, then $N_{a,b}$ transforms regular functionals into regular functionals (Prop. 3.40).

The function which is computed by the programm

$$\mathrm{Fab} \;:\; G(x) \Leftarrow \tau[G](x)$$

will be then the greatest fixpoint of the regular functional

$$\tau' = N_{a,b}(\tau).$$

To see the link with Scott's retract theory, let us call, if X is a bundle, a <u>retraction</u> over X any regular function $r : X \to X$ such that $r o r = r$; r defines then the <u>retract</u> $r(X)$.

Suppose that all function spaces are supplied with faithful bundles.

<u>LEMMA 3.42.</u> : Let X be a bundle, Y a convenient domain, $a \in X$, $b \in Y$. Then

(i) The transformation $N_{a,b}$ is a retraction over the space of regular functionals

(ii) There are no retractions $r : X \to X$, $t : Y \to Y$ such that $\forall f \in [X \to Y] \quad N_{a,b}(f) = t o f o r$. □

So there is a difference between Scott's retraction $a \; o\!\!\to b$ and the transformation $N_{a,b}$.

IV. <u>SAGITTAL BUNDLES AND LIMITS</u> :

A second form of the approximation notion is the one which allows to reconstruct spaces as limits of other spaces. Let $\mathcal{U}$ be a fixed universe. All the bundles we will consider will have their carrier in $\mathcal{U}$.

If X and Y are two bundles, a regular function $f : X \to Y$ is <u>strongly regular</u> iff $\forall x \in X \quad \forall \sigma \in \beta(x) \quad (f \; o \; \sigma) \in \beta(f(x))$. The category <u>Fais</u> which has bundles as objets and strongly regular functions as morphisms has equlizers, coequalizers, products and co-products. It is then complete and co-complete.

The same holds for the bundle categories <u>Foen</u>, <u>Ups</u>, <u>DCont</u>

DAlg, etc... This shows how to obtain certain objects as diagram limits (or colimits). We call sagittal bundle this generalization of the bundle notion, and from the computer science point of view shall restrict here our interest in category theory.

Bundle exponentiation can be done in several ways (Prop. 4.4., 4.5.) For suitably manipulating the notion of type (in the combinatory logic sense), we use a double bundle structure on a set, or 2-bundle (bifaisceau). There is a particular exponention on 2-bundles, which is obtained by using the upper threshold functions :

PROPOSITION 4.20. : Let BFt be the category whose objects have as a lower bundle a Foen-structure, and as an upper bundle a DAlg-structure, and which contain a $\top$, and whose arrows are the regular functions for the lower bundle. Then

$$N : \text{Foen} \times \text{BFt} \to \text{DAlg}$$
$$(X,Y) \longrightarrow [X \to Y]$$

which is defined over the objects by : $[X \to Y]$ is supplied by the algebraic bundle defined through Prop. 2.18. by the function

$$\forall f \in [X \to Y] \qquad s : f \mapsto s(f)$$

$s(f) = \{ \langle e,e' \rangle_{\top} : e' \sqsupseteq f(e),$ e rational, e' compact for the upper bundle of $Y\}$ defines a functor. □

By using the notion of a convenient 2-bundle (which are nothing else but convenient domains equipped with an upper bundle which is in DAlg), one constructs a domain with types, A_∞, which generalizes E_∞, Wadsworth's domain with atoms.

THEOREM 4.27. : Let D be a convenient 2-bundle. Then Wadsworth's approximating scheme supplies in Set a projective limit A_∞ which is such that

(i) $A_\infty = D + \varprojlim \Delta_k$, the limit of Δ_k being computed in DAlg (which provides A_∞ with a bundle structure).

(ii) If $[A_\infty \to A]_{\wp}$ is the space of partial regular functions from A_∞ to A_∞, then there exists a one-to-one correspondence

$(\underleftarrow{\lim}\ \Delta_k) \leftrightarrow [A_\infty \to A_\infty]\varphi$. In particular, A_∞ contains the set of all regular functions from A_∞ to A_∞ . □

Introduction

Dans un cours professé en 1798 à l'Ecole Polytechnique J. L. Lagrange [LAG] précise que l'un des objectifs de son livre "Théorie des fonctions analytiques", paru l'année précédente, était de "rattacher le calcul (différentiel) au reste de l'algèbre de manière à ne faire du tout qu'une seule méthode". Après de nombreux auteurs, nous pensons qu'un tel programme n'est pas déraisonnable dans le domaine de l'informatique; plus précisément qu'il conviendrait de rattacher entre elles, et au reste de l'algèbre, les différentes façons de définir le calcul informatique (i.e. la sémantique des langages de programmation), de manière à ne faire du tout qu'une seule et même méthode générale.

En plus de son apport mathématique, l'intérêt d'une telle mise en place se mesure finalement à l'importance de la sémantique des langages de programmation elle-même. Un parallèle avec le calcul différentiel peut en donner une idée. De même que la maîtrise du calcul différentiel a permis les progrès des sciences physiques où la notion de fonction joue un grand rôle, une maîtrise plus grande du calcul informatique permettra non seulement d'utiliser plus sûrement les ordinateurs, mais encore à plus long terme de faire progresser les sciences qui utilisent la programmation.

Il nous semble que le problème actuel de la sémantique (c'est en tout cas celui auquel nous nous intéressons ici) est l'existence de nombreux modèles pour rendre compte de la réalité informatique. Ceci peut sembler un avantage puisque l'on dispose de plusieurs outils pour manipuler un seul objet, le programme. Mais parfois ces outils ne s'appliquent pas en chaque circonstance. Certes, la confrontation de ces différentes approches paraît susceptible de faire jaillir des idées nouvelles; il reste que cette absence d'unité apparente est irritante du point de vue épistémologique, puisqu'au fond la réalité décrite est la même.

L'un des buts que l'on peut assigner à l'informatique théorique est de dégager des concepts de base, qui permettent de reconstruire tout le reste. (Un exemple analogue est celui de la température en physique : la température est une notion de base en physique non pas parce qu'elle ne se dérive pas au niveau théorique d'autres notions mais parce qu'elle correspond bien à un "objet" fondamental de la partie de l'univers que la physique se propose d'étudier).

Bien plus modestement, nous nous intéressons ici à un aspect particulier de ce problème général, qui est celui de la reconstruction des points dans les différents espaces sémantiques. Faute de pouvoir affirmer que les structures étudiées dans ce mémoire sont de tels "objets fondamentaux", nous voulons présenter notre démarche dans cette direction.

Notre thèse s'articule comme suit.

Nous commençons (Prolégomènes) par un long survol de trois théories importantes de la sémantique des langages de programmation : la sémantique dénotationnelle issue des travaux de D. Scott, la sémantique algébrique issue des travaux de M. Nivat, la théorie des algorithmes de L. Nolin. Nous nous attardons longuement sur cette dernière qui est la plus mal connue des trois, n'ayant fait l'objet que de publications rares et fragmentaires. A la fin de ce chapitre nous pouvons énoncer, plus précisément que nous ne le faisons ci-dessus, le problème qui nous intéresse.

Le chapitre 0 est consacré à un résumé sommaire du contenu des chapitres suivants.

Les autres chapitres, plus techniques, sont consacrés à l'exposé proprement dit. Nous y définissons une structure nouvelle, que nous appelons faisceau, et qui nous semble pouvoir jouer de façon satisfaisante un rôle de lien entre les trois théories mentionnées ci-dessus d'une part, et servir d'outil à une étude de leurs propriétés locales d'autre part.

Cette structure nous permet de simplifier, en les plaçant dans un cadre plus général, plusieurs constructions dûes à divers auteurs : modèle D_∞ de Scott avec D continu, powerdomains au sens de Smyth et Plotkin, etc.

Nous fournissons aussi avec notre modèle A_∞ une justification théorique à la théorie des algorithmes de L. Nolin, et nous énonçons plusieurs résultats nouveaux.

En esquissant ainsi une théorie locale des modèles de sémantique des langages de programmation, nous nous proposons de contribuer à une théorie unifiée de ces modèles.

Prolégomènes

Les programmes calculent des fonctions et l'on sait depuis les travaux de Schönfinkel [SCH] , Church [CHU] , et Curry et al. [CUR] que le λ-calcul est un moyen naturel d'exprimer (au moyen de la λ-notation) et calculer (au moyen des règles de réduction) des fonctions en un sens idéal. Le λ-calcul est composé d'un langage (le λ-langage), qui assure la fonction d'expression, et d'un ensemble de règles de réduction, qui assure la fonction de calcul. Le langage est engendré par la grammaire

$$\Lambda \to V \mid C \mid (\Lambda\,\Lambda) \mid (\lambda V . \Lambda)$$

où V, respectivement C, désigne des variables, respectivement des constantes. Les règles de réduction sont :

- la α-règle (changement de variable muette) :

  $\lambda x.M \rhd \lambda y.\ [y/x]\ M$ si y n'est pas libre dans M

- la β-règle (évaluation d'une fonction en un point) :

  $$(\lambda x.M)N \rhd (N/x)M$$

- et aussi, pour le $\beta\eta$-calcul, une η-règle

  $$(\lambda x.Mx) \rhd M \text{ si } x \text{ n'est pas libre dans } M$$

- avec parfois des δ-règles si C n'est pas vide (calcul sur les constantes).

De nombreux travaux [LAN] ,[BOH] ,... se sont attachés à montrer que les composants essentiels des langages de programmation usuels pouvaient s'écrire sous forme de λ-expressions. Par exemple en ce qui concerne le langage Algol 60 [ALG] :

- <u>L'appel de procédure</u> lorsque les paramètres effectifs sont transmis par nom : la β-réduction ne fait alors qu'exprimer ce phénomène.

- <u>le point-virgule</u> qui indique la succession de deux instructions :

  $$« ; » = CB = (\lambda x\, y\, z.\ x\, z\, y)\ (\lambda x\, y\, z.\ x(y\, z))$$

- <u>le conditionnel</u>

<u>si</u> ... <u>alors</u> ... <u>sinon</u> ... <u>fsi</u> = $B(B(B(BW)B))C'$

où $B = \lambda x\, y\, z.\ x(y\, z)$

$W = \lambda x\, y.\ x\, y\, y$

$C' = \lambda x\, y\, z.(z(K\, y)x)$ avec $K = \lambda x\, y.x$

et caetera.

Mais il est évident que le λ-calcul reste un langage qui exprime un calcul tout autant qu'un langage de programmation. La simplicité de sa définition peut le faire considérer comme un langage de programmation privilégié si on se place à un niveau conceptuel; mais le fait qu'on puisse tout dire avec des λ-expressions n'est pas toujours pratique : aucun arithméticien ne passera par les numéraux de Church pour étudier la théorie élémentaire des nombres.

Reste alors la question de savoir quels sont la nature et le comportement des objets que l'on (cherche à) exprime(r). C'est tout le problème de la modélisation. Nous discuterons ici cette question à travers l'examen de quelques structures déjà proposées par d'autres auteurs.

1. Sémantique dénotationnelle :

Le λ-calcul est un système formel; existe-t-il une structure ensembliste dont il exprime le calcul ? Le fait bien connu, découvert par Kleene et Rosser [KLR], que l'identification de tous les termes sans forme normale proposée par Church [CHU] conduit à une inconsistance, i.e. à identifier tous les termes, accroit l'intérêt de cette question. En effet si $\omega = \lambda x.x\,x$, $K = \lambda\, x\, y\,.\,x$, $P = \lambda\, x\, y\, z\,.\, x\, y(\omega\,\omega)$, $Q = \lambda\, x\, y\, z.\, x\, z(\omega\omega)$, P et Q sont sans forme normale, et identifier P et Q conduit à identifier $P\,K\,M\,N$ et $Q\,K\,M\,N$ donc M et N pour tous termes M et N.

La question purement mathématique, mais à retombées informatiques comme on l'a vu plus haut, de l'existence d'un modèle du λ-calcul, se pose donc naturellement.

Une réponse est fournie par D. Scott [1972] qui construit un modèle, "logique" selon son expression même [1973] . Dans sa construction l'auteur utilise des espaces injectifs et le modèle D_∞ est un tel espace. Par définition un espace injectif D est un T_0-espace topologique possédant la propriété de prolongement suivante :
Pour tous espaces topologiques X, Y tels que X soit un sous-espace de Y, pour toute fonction $f : X \to D$ continue, f possède un prolongement continu $\overline{f} : Y \to D$

$$\begin{tikzcd} X \arrow[r, hook] \arrow[dr, "f"'] & Y \arrow[d, dashed, "\overline{f}"] \\ & D \end{tikzcd}$$

Une autre façon, peut-être plus populaire, de voir les espaces injectifs est de les considérer comme des espaces partiellement ordonnés appelés <u>treillis continus</u>, où la notion d'ordre est censée représenter la notion "intuitive" d'augmentation d'information en cours d'un calcul dans un programme. La définition des treillis continus est un peu plus longue à énoncer :

Soit X un ensemble. Un <u>ordre partiel</u> sur X. est une relation binaire $\subseteq$ qui est

(i) réflexive i.e. $\forall x \in X \quad x \subseteq x$

(ii) antisymétrique i.e. $\forall x, y \in X \quad x \subseteq y$ et $y \subseteq x \Rightarrow x = y$

(iii) transitive i.e. $\forall x, y, z \in X \quad x \subseteq y \subseteq z \Rightarrow x \subseteq z$

L'ensemble X partiellement ordonné par $\subseteq$ est un <u>treillis complet</u> si et seulement si toute partie $S \subseteq X$ a une <u>borne supérieure</u> (i.e. un plus petit majorant) notée $\cup S$ et une <u>borne inférieure</u> (i.e. un plus grand minorant) notée $\cap S$. Le plus grand élément d'un treillis complet est généralement noté $\top$ (top) et le plus petit $\bot$ (bottom)

Une partie $S \subseteq X$ est dite <u>dirigée</u> ssi $\forall x\ y \in S \quad \exists z \in S$ tq $z \supseteq x$ et $z \supseteq y$. La topologie de Scott, qui joue un rôle important dans la construction, va être définie comme la topologie qui, intuitivement, fait converger les parties dirigées qui ont une borne supérieure dans l'espace ordonné considéré, X.

Plus précisément la <u>topologie (supérieure) de Scott</u> $\sigma(X)$ sur un espace partiellement ordonné X a pour ouverts les parties $\mathcal{U} \subset X$ telles que

(i) $\forall x \in \mathcal{U} \quad \forall y \in X \quad x \subseteq y \Rightarrow y \in \mathcal{U}$

(ii) $\forall S \subseteq X \quad S$ dirigée et $\cup S \in \mathcal{U}$ existe $\Rightarrow S \cap \mathcal{U} \neq \emptyset$.

On vérifie aisément que la famille $\sigma(X)$ définit bien une topologie sur tout espace partiellement ordonné X .

Si $x,y \in X$ où X est un treillis complet on dira que l'élément y est topologiquement bien en dessous de l'élément x et on note $y \ll x$ ssi :

$$x \in \text{Intérieur}\ (\{z \in X : z \supseteq y\})$$

Par définition on dira que le treillis complet X est un <u>treillis continu</u> ssi

$$\forall x \in X \qquad x = \cup\{y \in X : y \ll x\}$$

En fait Scott montre que l'espace de Sierpinski

$$\Sigma = \begin{array}{c} \top \\ | \\ \bot \end{array}$$

est à la fois un treillis continu et un espace injectif, et que tous les treillis continus (espaces injectifs) sont obtenus en prenant la clôture de $\{\Sigma\}$ par produit cartésien et rétraction (on dit que $f : X \to X$ est une <u>rétraction</u> ssi f est continue et la restriction de f à $f(X)$ est l'identité).

Des exemples concrets non triviaux de treillis continus sont [SCS] :

(i) tout segment $[a,b]$ de $\mathbb{R}$ est un treillis continu

(ii) si X est un espace quasi-compact (i.e. si tout recouvrement ouvert dénombrable de X contient un recouvrement fini) alors l'espace $O(X)$ des ouverts de X, ordonné par inclusion, est un treillis continu

(iii) si X est compact alors l'espace $SCI(X,\mathbb{R} \cup \{\pm\infty\})$ des fonctions semi-continues inférieurement de X dans la droite réelle achevée $\mathbb{R} \cup \{\pm \infty\}$ est un treillis continu.

Si l'on se donne une suite inductive/projective de treillis continus

$$D_0 \underset{i_0}{\overset{j_0}{\rightleftarrows}} D_1 \underset{i_1}{\overset{j_1}{\rightleftarrows}} D_2 \rightleftarrows \cdots\cdots\cdots\cdots \rightleftarrows D_n \underset{i_n}{\overset{j_n}{\rightleftarrows}} \cdots\cdots$$

où les (i_n, j_n) sont des projections au sens de Scott i.e. des couples d'applications continues

$$D_n \underset{i_n}{\overset{j_n}{\rightleftarrows}} D_{n+1}$$

tels que $j_n \circ i_n$ est l'identité et $\forall y \in D_{n+1} \quad i_n(j_n(y)) \subseteq y$
i.e. $i_n \circ j_n \subseteq id_{D_{n+1}}$ alors la limite projective

$$\underleftarrow{\lim}(D_n, j_n) = \{(x_m)_{m \in \mathbb{N}} \in \prod_{m=0}^{\infty} D_m \mid x_m = j_{m+1}(x_{m+1})\}$$

est encore un treillis continu à cause de la propriété de prolongement. Si on se place dans la catégorie des treillis complets munis des fonctions continues, alors la limite inductive est égale à la limite projective :

$$\underleftarrow{\lim}(D_n, j_n)_{n \in \mathbb{N}} = \underrightarrow{\lim}(D_n, i_n)_{n \in \mathbb{N}} = D_\infty$$

en un sens qu'on peut préciser de la manière suivante en utilisant un résultat de [SCS] , pp. 212 :

1/ Soit $INF\uparrow$ la catégorie dont les objets sont les treillis complets et les flèches les fonctions continues au sens de Scott préservant les bornes inférieures arbitraires (i.e. sont exactement les morphismes de semi-treillis continus au sens de Lawson ([SCS] , pp. 145))

2/ Soit UPS la catégorie dont les objets sont les treillis complets et les flèches les fonctions continues au sens de Scott

3/ Soit ω^{op} la catégorie $(Ob(\omega) = \mathbb{N})$

$$0 \leftarrow 1 \leftarrow 2 \leftarrow 3 \leftarrow 4 \ldots$$

4/ Soit $D : \omega^{op} \to INF\uparrow$ un système inverse dans $INF\uparrow$, $g_n : L \to D_n$ un cône [1] sur D. Alors il y a équivalence entre les deux assertions

[1] CF définition de cette notion Chap. IV . §2.

(i) $g_n : L \to D_n$ est <u>cône limite sur</u> D dans INF↑ (i.e. L est la limite projective des D_n dans INF↑)

(ii) $\hat{g}_n : D_n \to L$ est un <u>cône colimite sous</u> D dans UPS (i.e. L est la limite inductive des D_n dans UPS)

(où $\hat{g}_n$ désigne <u>l'adjointe supérieure</u> de g_n au sens des <u>connections de Galois</u> (cf [MCL] pp. 93-94, [SCS] p. 18 sqq)).

Ici on a $\forall n \quad g_n = j_n \quad \hat{g}_n = i_n$.

Si l'on particularise la suite projective/inductive précédente en prenant :

(i) $\forall n \in \mathbb{N} \qquad D_{n+1} = [D_n \to D_n] = \{f : D_n \to D_n \mid f$ est continue au sens de la topologie de Scott $\}$

(ii) $\forall n \in \mathbb{N} \qquad i_{n+1}(x) = i_n \circ x \circ j_n$

$$j_{n+1}(y) = j_n \circ y \circ i_n$$

on a même <u>l'homéomorphie</u>

$$D_\infty \simeq [D_\infty \to D_\infty]$$

Ce dernier résultat permet d'exhiber une large classe de solutions à l'équation aux espaces :

$$\exists ?\ D \qquad D \simeq [D \to D]$$

La question qui se pose, avant d'appliquer ceci à l'informatique, est : en quoi les espaces partiellement ordonnés sont-ils pertinents dans l'étude des programmes En quoi les treillis continus permettent-ils d'exprimer les concepts fondamentaux de la programmation ?

L'argument est le suivant. Il existe des programmes qui, pour quelque donnée, ne s'arrêtent pas, i.e. dénotent des fonctions partielles au sens usuel. L'idée de Scott est de compter cette abscence d'arrêt comme une valeur "moins définie que toutes les autres", notée $\perp$, au même titre que les autres valeurs du domaine. Cet élément $\perp$ est encore assimilé à l'absence d'information, lorsque par exemple on se prépare à éxécuter un programme. Ceci amène à prendre en compte, au niveau de la structure mathématique l'idée "moins définie que", qui se manifeste par une relation d'ordre partiel, et à ne considérer que des fonctions monotones croissantes pour cet ordre partiel. L'application à la programmation a été faite alors en s'appuyant sur les thèses suivantes (cf. par exemple Plotkin 1978)

(i) Tous les domaines sont des ordres partiels avec un plus petit élément.

(ii) Toute fonction calculable est monotone croissante.

On veut évaluer des appels récursifs, donc

(iii) Toute chaine ascendante du domaine converge (i.e. le domaine est un cpo).

(iv) Toute fonction calculable est continue.

Puisque un espace ordonné X est un cpo si et seulement si toutes ses parties dirigées convergent (au sens de la topologie de Scott ou, si l'on préfère au sens de l'existence des bornes supérieures dans un tel espace), le cpo peut apparaître du point de vue qui consiste à identifier $\perp$ avec l'absence d'information, comme une belle illustration de la sémantique de l'appel de procédure résursive. Cette structure, bien moins riche que le treillis continue, a suffi pour exprimer une partie de la sémantique des programmes [STO,VUI] . Les cpo's les plus utilisés en théorie de la programmation sont les cpo's dits algébriques qui se rapprochent un peu plus des treillis continus. Un cpo X est <u>algébrique</u> ssi les éléments compacts $a \in X$, i.e. vérifiant $\forall$ partie dirigée $S \subseteq X$, $a \sqsubseteq \sqcup S$ implique $\exists s \in S$ tel que $a \subseteq s$, permettent de reconstruire tous les autres par borne supérieure. La notion d'algébricité sur les cpo's peut être considérée comme une trace de la notion de "calculabilité", ou mieux d'approximation, qu'on avait sur les treillis continus :

$$\forall x \in X \qquad x = \sqcup \{ y \in X : y \ll x \}$$

G. Plotkin a construit un modèle du λ-calcul, $\mathbb{T}^{\omega}$, dans le cadre des cpo's, en utilisant une gödelisation dans le style de celle du modèle P_{ω} de Scott (Scott 1976, Plotkin 1978).

Si l'on se place du point de vue de l'étude des modèles, on ne peut manquer ici de noter l'apparition d'une nouvelle structure à côté du treillis continu : le cpo.

L'utilisation du cpo de préférence au treillis continu pour la théorie des langages de programmation a été faite au nom d'un principe d'économie d'une part, et parce que le fameux élément top $(\top)$ du treillis continu ne semblait pas avoir d'interprétation bien claire pour les programmeurs [VUI] d'autre part. D.Scott explique le choix du treillis continu de la manière suivante : "Une raison pour avoir un treillis entier était le théorème de prolongement pour les fonctions continues. Si D est un treillis continu alors toute fonction continue $f : X \to D$ peut être prolongée en une fonction continue $f : Y \to D$ où Y est un sur-espace quelconque tel que $X \subseteq Y$. En effet cette propriété caractérise les treillis continus, et elle semblait très raisonnable si nous voulons des modèles riches pour les fonctions partielles. Par exemple toute fonction partielle continue $f : A \to B$ où $A,B \subseteq D$ peut être vue comme la restriction d'une $\tilde{f} : D \to D$ totale si D est un treillis continu. Si $D = D \to D$ nous pouvons voir pourquoi il est agréable de pouvoir représenter par quelque élément $u \in D$ toute espèce de fonctions continues sur tous les sous-espaces de D" (Scott 1975, pp. 355).

L'importance, du point de vue informatique, de cet argument de Scott nous paraît justifier l'approche prise plus haut pour définir les treillis continus, par le détour de la topologie de Scott. D'un point de vue mathématique, il aurait été plus économique de se donner d'emblée la "way -below relation" (relation "bien en dessous")

$$x \ll y \iff x \in \bigcap \{I \text{ idéal} \subseteq X : y \leq \bigcup I\}$$

(cf. [SCS] I-1.1 pp. 38),
(donc définir une approximation par idéaux au lieu d'une approximation topologique), et de dire :

X treillis complet est continu ssi

$$\forall x \in X \qquad x = \bigsqcup \{z \in X : z \ll x\}$$

Ces deux relations "way-below" et "way-below" topologique ne sont généralement pas équivalentes (cf. ch. II. Prop. 2.9).

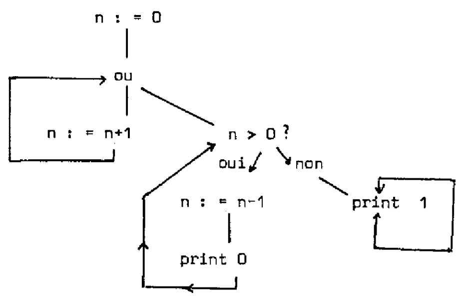
Notons encore ici que les méthodes issues des travaux de Scott ont été utilisées pour traiter des problèmes plus complexes d'informatique. On sait par exemple que pour une machine de quelque dimension l'unité centrale, l'imprimante et le lecteur de cartes fonctionnent en parallèle. Le besoin de formaliser le non-déterminisme et le parallélisme a conduit G. Plotkin [PLO] et B. Smyth [SMY] à l'introduction des powerdomains, présentés comme des cpo's de parties de cpo's. Ces powerdomains sont généralement munis de l'ordre d'Egli-Milner (qui est en fait un préordre) défini par :

$$A \subseteq_M S \text{ si et seulement si}$$

(i) tout élément de S majore un élément de A (i.e. l'information continue dans A est suffisante pour approximer l'information finale contenue dans S)

(ii) tout élément de A minore un élément de S (c'est-à-dire l'information contenue dans A est nécessaire pour approximer l'information finale contenue dans S).

Cette introduction ne résout cependant pas tous les problèmes, comme l'indique l'exemple suivant (cité dans Smyth) : soit le programme



Si X est l'ensemble des suites finies ou infinies de 0 et de 1, ordonné par l'ordre préfixe ($a \subseteq b$ ssi la suite b commence par la sous-suite a), alors l'ensemble des résultats possibles du programme précédent est

$$S_0 = \{\varepsilon\} \cup \{0^n 1^\omega \mid n \in \mathbb{N}\}$$

où ε est la suite vide.

Mais l'ordre d'Egli-Milner conduit à identifier S_0 avec l'ensemble

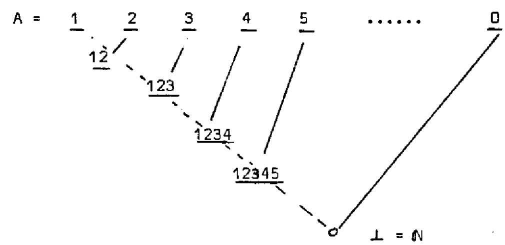
$$S_1 = \{\varepsilon\} \cup \{0^n 1^\omega \mid n \in \mathbb{N}\} \cup \{0^\omega\}$$

qui mentionne un résultat, 0^ω, qui ne sera jamais donné par ce programme.

Un autre problème est celui des types dans les langages de programmation. L'on sait que lors de la compilation-interprétation d'un programme, chaque variable possède un descripteur qui peut contenir entre autres attributs le type de cette variable (réel, complexe, booléen,...) ou sa valeur [AHO] . Le type existe donc véritablement en tant qu'objet à l'intérieur de la machine, au même titre qu'une valeur numérique. On aimerait donc calculer sur des types comme si l'on avait véritablement des objets "primitifs" du domaine [SHA,NOL] . Cela est-il possible dans le cadre des cpo et des fonctions continues, en gardant à l'esprit le fait que $\perp$ est l'élément "le moins défini" du domaine ?

Prenons un exemple :

Soit L un langage de programmation qui permet de calculer sur des entiers et possède une infinité dénombrable de constantes de types $\mathbb{N}$, $\underline{12}$, $\underline{123}$,...,$\underline{123...n}$,... pour $n \in \mathbb{N}^*$, qui correspondent au fait qu'un entier puisse appartenir à l'ensemble $\mathbb{N}$, {1,2} , {1,2,3} , etc... S'appuyant sur le fait que savoir qu'un nombre est de type {1,2,3,4} est un pas intermédiaire entre l'absence d'information et le fait qu'il est égal à 1 par exemple, on obtient le cpo suivant :



Soit le programme bien connu :

$F(n)$ = si n = 0 alors 0 sinon si n = 1 alors 1 sinon
F(n-1) + F(n-2) fsi fsi

La fonction calculée f est le plus petit point fixe de la fonctionnelle ci-dessus définie $F \to \tau\,[F]$, et

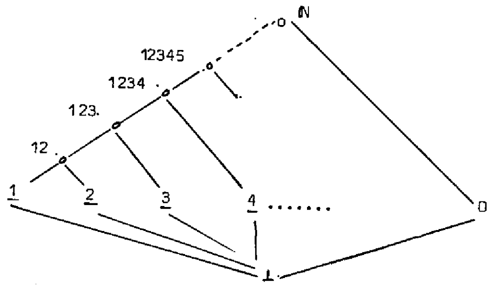
$f(\underline{12})$ = si $\underline{12}$ = 0 alors 0 sinon si $\underline{12}$ = 1 alors 1 sinon
$f(\underline{12}-1) + f(\underline{12}-2)$ fsi fsi
$= 1 \sqcap [f(\perp) + f(\perp)] = 1 \sqcap \perp = \perp$

c'est-à-dire que le plus petit point fixe, évalué en chaque type $\underline{1...n}$, ne donne aucun renseignement et ceci ne peut donc être utilisé pour faire des vérifications de type, activité pourtant normale pour le programmeur. Cela introduit de plus une dichotomie, qui, on l'a vu, n'existe pas au niveau de la compilation-interprétation, entre d'une part les types "atomiques" comme 0,1,2,... - où le plus petit point fixe se comporte normalement - et d'autre part les types "non-atomiques" comme $\underline{12}$, $\underline{123}$,... - où le plus petit point fixe ne donne généralement pas le résultat espéré.

A ceci l'on peut répondre qu'il suffit, pour que le plus petit point fixe garde toute sa valeur, d'utiliser les éléments "atomiques" comme "verrou" pour empêcher que la valeur du plus petit point fixe n'aille "s'écraser" sur $\perp$. Le prix à payer est que $\perp$ ne correspond plus à l'idée intuitive d'absence d'information.

On obtient alors la structure suivante :

A' =



qui est un treillis complet donc un cpo. Cette solution est celle qu'adoptent Shamir et Wadge, mais l'utilisation de fonctions sur A' qui sont des plus petits prolongements de fonctions définies sur le sous-cpo $\mathbb{N}_\perp = \mathbb{N} \cup \{\perp\}$ (que ces auteurs appellent "tight") requiert dans l'évaluation de $f(\underline{1234})$ de partitionner le type $\underline{1234}$ en 1 et le "reste" $\underline{234}$, qui n'est pas un élément du domaine (cf. [SHA] pp. 477-478). Cette partition n'a pas de justification théorique dans la construction.

D'autre part il existe des fonctions continues sur le sous-cpo $\mathbb{N}_\perp$ de A' dont le prolongement minimal n'est pas une fonction continue sur le cpo A' comme par exemple

$$\begin{aligned} f : x &\to 1 \qquad \text{si} \quad x \neq 0, \perp \\ 0 &\to 0 \\ \perp &\to \perp \end{aligned}$$

Cette fonction est continue sur $\mathbb{N}_\perp$ et son prolongement minimal $\bar{f} : A' \to A'$ est défini par :

$$\bar{f}(\underline{12}) = f(1) \cup f(2) = 1$$

$$\bar{f}(\underline{12\ldots n}) = f(1) \cup f(2) \cup \ldots \cup f(n) = 1 \qquad \forall n \in \mathbb{N}$$

$$\bar{f}(0) = f(0) = 0$$

$$\bar{f}(\mathbb{N}) = \bar{f}(\underline{12}) \cup \ldots \cup \bar{f}(0) = 1 \cup 0 = \mathbb{N}$$

Mais la fonction $\bar{f}$ n'est pas continue car

$$\mathbb{N} = \bar{f}(\mathbb{N}) = \bar{f}(\cup\{\underline{1\ldots n} \mid n \in \mathbb{N}^+\}) \neq$$
$$\cup\{\bar{f}(\underline{1\ldots n}) \mid n \in \mathbb{N}^+\} = \cup\{1 \mid n \in \mathbb{N}^+\} = 1$$

Ceci ne coïncide pas avec les thèses (iii) et (iv) mentionnés plus haut.

Ce bref survol peut peut-être susciter quelques questions sur la structure du domaine à utiliser (exemple de Smyth sur les powerdomains), la nature des fonctions intéressantes pour la programmation (problèmes de types), la nature de l'élément $\perp$ donc la qualité du point fixe calculé par un programme.

II. Sémantique algébrique

La sémantique algébrique [NIV,GOG] travaille sur des schémas de programmes (notion introduite par Ianov [IAN]) qui sont définis comme des systèmes d'équations sur une algèbre $M(F,V)$, où F est l'ensemble des symboles fonctionnels et V l'ensemble des variables, appelée magma par M. Nivat. On ordonne $M(F,V)$ en introduisant un élément Ω (= "l'absence d'information") plus petit que tous les autres et en prolongeant l'ordre par monotonie :

si $f(a_1,\ldots,a_i,\ldots,a_n)$ et $f(a_1,\ldots,b_i,\ldots,a_n)$ sont dans $M(F,V \cup \{\Omega\})$ et $a_i \leq b_i$, alors $f(a_1,\ldots,a_i,\ldots,a_n) \leq f(a_1,\ldots,b_i,\ldots,a_n)$

La signification d'un schéma de programme est alors définie par M. Nivat et J. Vuillemin comme étant un arbre infini approximé par des chaînes ascendantes (dénombrables) d'arbres finis, c'est-à-dire d'arbres appartenant à $M(F,V \cup \{\Omega\})$. L'ensemble de tels arbres finis et infinis définit le magma complété $M^{\infty}(F,V)$, qui possède une structure de cpo algébrique. M. Nivat décrit ainsi sa construction, inspirée, dit-il, de celle de D. Scott dans "Lattice of flow diagrams" : "Ce que nous avons surtout voulu faire, c'est expliciter l'analogie formelle entre schémas de programmes et systèmes d'équations algébriques, et de ce fait l'accent se trouve déplacé de la structure d'ordre qui sous-tend la construction de D. Scott, vers la structure de magma dont se trouve muni l'ensemble des schémas de programmes". [NIV] . L'intérêt de la sémantique algébrique est qu'elle factorise toute interprétation d'un programme (au sens de la sémantique dénotationnelle) à travers une interprétation libre dans $M^{\infty}(F,V)$ et permet en étudiant d'abord cette interprétation de faire le lien avec la théorie des langages algébriques et de développer des techniques inspirées de l'algèbre universelle (Courcelle-Nivat 1978, Goguen 1978,...).

Pour rendre compte aussi des schémas de programmes non-déterministes cette sémantique a été généralisée par M. Nivat et A. Arnold en une théorie métrique où le théorème du plus petit point fixe dans un cpo est remplacé par le théorème bien connu du point fixe d'une fonction contractante dans un espace métrique complet. Le magma $M(F,V$ se trouve ainsi muni de la distance

$$d(x,y) = 0 \quad \text{si} \quad x = y$$

$$= \frac{1}{\min\{n : \pi_n(x) \neq \pi_n(y)\}} \quad \text{sinon}$$

où $\pi_n(x)$ désigne la coupe à hauteur n de l'arbre $x \in M(F,V)$. La propriété intéressante est que le complété métrique de $M(F,V)$ pour d, i.e. l'espace des classes d'équivalence des suites de Cauchy sur $M(F,V)$, peut être identifié avec le cpo algébrique $M^\infty(F,V)$ diminué de l'élément Ω .

La fonction calculée par une procédure est alors le plus grand point fixe de quelque fonctionnelle (Arnold et Nivat 1978). De plus l'introduction de la métrique permet d'éliminer l'élément controversé Ω(ou $\perp$), en supprimant d'une part la notion d'absence d'information" et en faisant des programmes qui bouclent ce qu'ils sont en fait, des calculs infinis.

3. La théorie des algorithmes

On peut relever que la sémantique dénotationnelle a été élaborée en partie à partir de la modélisation du λ-calcul dans D_∞ , et que la sémantique algébrique prend ses racines quant à elle dans la théorie des langages algébriques [GUI] , les travaux de S. Kleene sur les fonctions récursives [KLE] , et la calculabilité au sens de Herbrand-Gödel ,[MEN] .

Une caractéristique des travaux de L. Nolin [NOL] sur la sémantique des programmes (appelés théorie des algorithmes) est l'approche expérimentale. Le but, plus ou moins lointain, de toute sémantique des programmes est de fonder une théorie de la compilation. Et de même que l'idée de base de l'étude par Scott des treillis continus semble être de disposer de la propriété de prolongement des fonctions continues et d'espaces universels en un certain sens (cf. supra), l'idée de base de la construction de Nolin semble être de construire un modèle aussi proche que possible de la compilation des programmes.

Les programmes considérés sont ceux d'un langage qui contient des déclarations de types, comme Algol 60 par exemple, et le souci primordial est celui de rendre compte des phénomènes qui se passent lors de la compilation et l'éxécution de tels programmes.

(i) Considérons la séquence pseudo-Algol suivante

```
boolean  x ;
  y : = 2;
  if  x  then  y : = y * 2  else  y : = y * 3  fi
```

et supposons qu'elle est fournie à un compilateur conversationnel. Lorsque le compilateur décodera la première instruction, il réservera de la mémoire pour un objet appelé x et dont il sait que la valeur sera booléenne, c'est-à-dire vrai ou faux (il y a donc une petite incertitude sur cette valeur). Lordqu'il décodera la deuxième instruction, il réservera de la mémoire pour un objet appelé y et dont il sait que la valeur sera 2, au début du calcul tout au moins. Mais dans les deux cas le traitement est au fond le même : à un objet x (respectivement y) on attache l'information <u>booléen</u> (respectivement 2). La déclaration doit être traitée comme une instruction d'affectation, i.e. l'objet booléen sera de la même nature que l'objet 2. D'où la

<u>Thèse 1</u> : Les objets qui apparaissent en tant que tels durant le processus de compilation doivent avoir une existence en tant qu'objets dans l'espace des dénotations (la structure interprétative) à définir

(ii) La question qui se pose maintenant est : quels sont ces objets . Réponse : il y a d'une part les types de base : <u>booléen</u>, <u>réel</u>, <u>entier</u>, etc..., d'autre part les types qui apparaissent dans les déclarations de procédures, comme par exemple :

```
entier naturel procédure p(y) entier naturel y
      début
            etc...
      fin
```

L'énoncé précédent concernant la procédure p donne une première approximation de la fonction f_p calculée par cette procédure. Il dit simplement que la fonction f_p appartient à l'ensemble des fonctions qui, restreintes à l'ensemble des entiers naturels, donnent une image inclue dans l'ensemble des entiers naturels. Ceci est noté : $f_p \in F\,\mathbb{N}\,\mathbb{N}$

si $F\,\mathbb{N}\,\mathbb{N} = \{f\ldots \mid \forall n \in \mathbb{N} \quad f(n) \in \mathbb{N}\}$

Cette approximation peut être rendue de plus en plus précise : par exemple, toujours dans le même jargon pseudo-Algol on peut dire

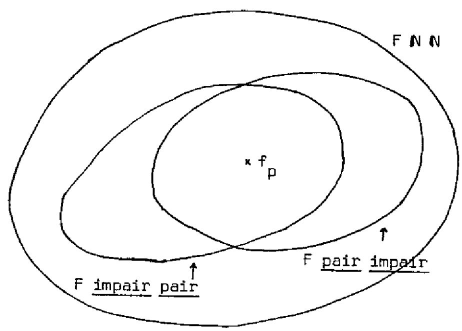
```
(pair → impair) et (impair → pair) procédure p(y)
      début
            etc...
      fin
```

ce qui nous dit que

$$f_p \in (F\ \underline{\text{pair}}\ \underline{\text{impair}}) \cap (F\ \underline{\text{impair}}\ \underline{\text{pair}})$$

i.e. $f_p \in \{f\ldots \mid \forall n \in \mathbb{N} \quad f(2n) \in 2\mathbb{N} + 1\} \cap \{f\ldots \mid \forall n \in \mathbb{N} \quad f(2n+1) \in 2\mathbb{N}\}$.

ce qui constitue évidemment une information plus précise sur f_p, puisqu'on a le diagramme



Un cas limite d'une telle spécification est le tableau qui donne pour chaque valeur de l'index une valeur du tableau :

$$\bigcap_{i \in [1,n]} F\{i\}\ \{a_i\} \quad , \text{ où } \ [1,n] = \{1,2,\ldots,n\}$$

A cet égard, il n'y a pas de différence fondamentale entre les procédures et les tableaux, ceux-ci n'étant en quelque sorte que des procédures idéales, "très bien" définies au travers de leurs spécifications. D'où la

Thèse 2 : les approximations sont faites de façon naturelle par intersections d'ensembles de la forme

$$F\ X\ Y = \{f\ldots \mid f(X) \subseteq Y\}$$

pour les fonctions calculées par les programmes et pour les tableaux.

Les types, au lieu d'apparaître en tant que rétractes comme dans la théorie de Scott, sont ainsi des objets fondamentaux de la construction.

(iii) Comme conséquence des deux exigences précédentes il s'ensuit que :

(a) si f_p est la fonction calculée par la procédure p alors

$$\{f_p\} = \bigcap \text{ quelques } F\ X\ Y\text{'s}$$

(b) si p est récursive alors f_p est un point fixe; f_p étant obtenue comme borne inférieure, c'est un plus grand point fixe. D'où la

Thèse 3 : La fonction calculée par une procédure est le plus grand point fixe de quelque fonctionnelle.

Ceci peut s'expliquer de la manière suivante. Lorsqu'on calcule, on part d'une valeur très mal définie du résultat, laquelle se trouve quelque part dans un ensemble (peut-être spécifié par une déclaration), par exemple : trouver l'entier qui est le 24è nombre de la suite de Fibonacci. L'on restreint alors de plus en plus cet ensemble par intersections, de manière à rendre cette valeur de plus en plus précise. L'information optimale - d'un point de vue déterministe - étant obtenue lorsque l'ensemble est réduit à un singleton . Il se peut aussi que l'ensemble soit vide, ce qui peut correspondre à un overflow, underflow,... On s'aperçoit d'ailleurs qu'il est plus commode de parler de parties que d'éléments.

Nous sommes maintenant prêts pour aborder les détails techniques de cette théorie. On peut tout de suite donner les grands axes de la construction : les objets seront les types au sens de la thèse 1 (appelés algorithmes par Nolin), et l'on a vu apparaître deux opérations sur ces objets : $F : (a,b) \to F\ ab$, et l'intersection (possiblement infinie). Nolin va définir un espace qui contient ces objets et qui est clos pour ces deux opérations.

La notion de base de la théorie est celle de <u>collection d'algorithmes</u>. L'auteur n'utilise pas le terme d'ensemble, car écrit-il "à première vue tout au moins, mes "ensembles" à l'exception des ensembles de base, ne peuvent être qualifiés tels dans aucune théorie des ensembles connue". ([NOL 78] pp. 267). Il ne définit pas ce qu'est une collection mais s'autorise sur ces objets les mêmes opérations que sur les ensembles, c'est-à-dire en gros : prendre l'union, l'intersection, la collection des parties d'une collection, un élément appartenant à une collection.

Si T (comme "tout") est une collection non vide, $\mathfrak{A}$ une collection de parties de T close pour l'intersection infinie, et contenant $\emptyset$ (la collection vide) et T comme éléments, on définit

(1) $\forall \{X_i\}_{i \in I} \subseteq \mathfrak{A}$ <u>l'union complétée</u> $\bar{\bigcup\limits_i} X_i$ par

$$\bar{\bigcup_i} X_i = \cap \{Z \in \mathfrak{A} : X_i \subseteq Z \quad \forall i \in I\}$$

(2) $\{X_i\}_{i \in I} \subseteq \mathfrak{A}$ est une <u>prépartition</u> de Y ssi

$$Y = \bigcup_i X_i \quad \text{et} \quad X_i \neq X_j \Rightarrow X_i \not\subseteq X_j$$

(3) $X \in \mathfrak{A}$ est <u>atomique</u> ssi $X \subseteq \bigcup_{i \in I} Y_i \Rightarrow \exists j \in I$ tel que $X \subseteq Y_j$

(4) $f : \mathfrak{A} \to \mathfrak{A}$ est <u>normale</u> ssi

$$\forall X \in \mathfrak{A} \quad f(X) = \bar{\bigcup} \{f(Y) : Y \text{ atomique } \subseteq X\}$$

(5) si $X, Y \in \mathfrak{A}$ on définit

$$F\, X\, Y = \{f : \mathfrak{A} \to \mathfrak{A} \mid f \text{ normale et } f(X) \subseteq Y\}$$

(6) $\mathfrak{A}_F$ est la plus petite collection qui contient les $F\ X\ Y$ où $Y \neq T$ et qui est close pour l'intersection infinie.

Ces préliminaires étant posés, on définit une collection d'algorithmes de la manière suivante :

Soit T une collection non vide, $\mathfrak{A}$ une collection de parties de T. La collection $\mathfrak{A}$ est une <u>collection d'algorithmes</u> ssi elle vérifie les quatre conditions :

(i) il existe une collection $\mathfrak{B}$ de parties de T contenant $\emptyset$ et T, et telle que $\mathfrak{A}$ est la plus petite sous-collection de parties de T, contenant $\mathfrak{B}$, close par intersection infinie et par l'opération

$$F : (X,Y) \to F\ X\ Y$$

avec $F\ X\ T = T$ et $F\ X\ \emptyset = \emptyset$

(ii) soit $\mathfrak{A}_B$ la clôture de $\mathfrak{B}$ par intersection infinie, amputée de $\emptyset$ et T; alors si $X \in \mathfrak{A}_B$ et si $Y \in \mathfrak{A}_F$, X et Y sont incomparables pour l'inclusion

(iii) $\forall X \in \mathfrak{A}\quad X = \cup \{Y \in \mathfrak{A} : Y \text{ atomique} \subseteq X\}$

(iv) T est atomique . □

On peut noter que cette définition énonce en même temps un problème de point fixe :

$$\mathfrak{A} = \mathfrak{A}_B + \mathfrak{A}_F$$

supposé résolu d'emblée. C'est pourquoi l'auteur, conscient de cette difficulté, s'en tient à la notion de collection, au lieu de parler d'ensembles.

La collection $\mathfrak{A}$ est munie d'une loi de composition (en fait une application au sens du λ-calcul) :

$$\forall X,Y \in \mathfrak{A}\quad X[Y] = \cap\, \{Z \in \mathfrak{A} : X \subseteq F\ Y\ Z\}$$

Si l'on oublie toutes les définitions qui précèdent pour ne considérer que la dernière, on s'aperçoit que ce n'est rien d'autre qu'une évaluation de type au sens de la logique combinatoire avec types ([HIN] pp. 75)

La loi de composition sur $\mathfrak{A}$ possède la propriété suivante :

$\forall \{Z_j\}_{j \in J} \subseteq \mathfrak{A}$ tel que $\bigcup_{j \in J} Z_j \in \mathfrak{A}$ on a

$$\Big(\bigcap_{i \in I \neq \emptyset} F\ X_i Y_i\Big)\ \Big[\bigcup_j Z_j\Big] = \bar{\bigcup_j}\ \Big(\Big(\bigcap_i F\ X_i\ Y_i\Big)\ [Z_j]\Big)$$

résultat qui semble d'écrire un phénomène analogue à la continuité sur les cpo et les treillis continue.

Dans cette théorie les données et types de bases sont interprétés comme des éléments de $\mathcal{A}_B$ (algorithmes de base), les procédures comme des éléments de $\mathcal{A}_F$ (algorithmes propres).

On peut noter que les préoccupations d'Arnold et Nivat en sémantique algébrique, et le traitement des types de données en tant qu'objets de calcul suggéré par Shamir et Wadge, rejoignent sur plusieurs points l'argumentation développée plus haut et qui constitue la justification informatique de la théorie des algorithmes.

Les relations entre ces différentes sémantiques ne semblent pas très claires et la question à laquelle nous nous intéressons est de les placer les unes par rapport aux autres. Dans ce qui suit nous proposons une construction qui nous permet d'obtenir une telle classification, et nous examinons plus particulièrement les différentes façons de reconstruire les points de l'espace à partir d'autres points.

De plus, nous obtenons des simplifications dans les constructions déjà existantes et de nouveaux résultats.

Certains auteurs, comme D. Scott et L. Nolin privilégient les treillis. D'autres comme M. Nivat, G. Plotkin, J. Vuillemin,... utilisent des structures ordonnées. La raison pour prendre une structure beaucoup plus générale, que nous appelons <u>faisceau</u> est qu'en fait à chaque point de l'espace des dénotations on associe toujours un système approximant de sous-ensembles de l'espace, qui décrit toutes les façons d'obtenir ce point par approximation, comme point limite. Ainsi un élément essentiel de nombre de structures mentionnées ci-dessus, est la possibilité de reconstruire tous les éléments de l'espace en partant uniquement d'un sous-ensemble restreint d'éléments privilégiés qui sont appelés : <u>éléments atomiques</u> (théorie des algorithmes), <u>éléments compacts</u> (cpo algébriques), <u>arbres finis</u> (magmas) . De tels éléments sont appelés dans notre approche éléments rationnels. Cette reconstruction de l'espace tout entier est faite soit par une <u>complétion métrique</u>, i.e. en prenant les classes d'équivalence des suites de Cauchy (magma métrique), soit en prenant les bornes supérieures (ou les bornes inférieures) de certaines parties (théorie des algorithmes). On pourrait appeler ce dernier mécanisme <u>complétion élémentaire</u>.

Donc schématiquement :

$$\text{éléments privilégiés (les rationnels)} \left\{ \begin{array}{lll} + \text{ complétion élémentaire} & \longrightarrow & \text{espace ordonné} \\ + \text{ complétion métrique} & \longrightarrow & \text{espace métrique complet} \end{array} \right.$$

Si l'on considère la première approche, celle de la complétion élémentaire, on voit alors émerger de façon très naturelle la notion <u>d'ordre élémentaire</u> telle qu'elle a été définie dans [NAI] . Cette notion fut utilisée pour définir la sémantique des programmes avec déclarations de types, et la fonction calculée par un tel programme fut définie comme le plus grand point fixe d'une certaine fonctionnelle.

Cependant la situation peut devenir plus complexe : les éléments rationnels peuvent disparaître et la notion d'ordre élémentaire devient insuffisante. Nous avons besoin d'une notion plus générale : celle de faisceau ordonné. Un faisceau ordonné est en gros un ordre élémentaire dans lequel les rationnels ne jouent plus de rôle privilégié, i.e. dont les fibres ne sont plus contraintes à ne contenir que des sous-ensembles de rationnels. Plus précisément un faisceau ordonné est la donnée sur un espace de toutes les façons possible de reconstruire les éléments de cet espace comme borne supérieure (ou inférieure) de certaines familles.

Nous définissons les fonctions régulières comme étant juste celles qui respectent la "complétion" que l'on avait sur l'ensemble de départ. La première raison pour prendre de telles fonctions est que nous avons à rendre compte des calculs finis, qui peuvent être exécutés d'une manière réalisable dans l'informatique pratique. La deuxième raison est de décrire la notion d'approximation utilisée pour les appels de procédure, le non-déterminisme, la vérification de type. Notre notion de fonction régulière généralise en un unique concept, très simple, les notions de fonction continue (Scott), algorithme (Nolin), "tight function" (Shamir et Wadge), fonction stable (Berry). Nous mettons, quant à nous, l'accent plus sur la notion d'approximation que sur celle de continuité au sens usuel (c'est-à-dire au sens topologique). Du reste la continuité de Scott n'est, sur les treillis continus, qu'une semi-continuité inférieure (Scott 1969, [SCS]).

L'idée que nous venons d'exprimer, à savoir le rôle central de la reconstruction des points, vient naturellement à l'esprit lorsqu'on considère la démarche d'un auteur comme Scott, par exemple, et sa non-utilisation des cpo's (Scott 1975, pp 355). Elle consiste simplement à constater que pour construire D_∞ , on utilise des espaces très particuliers où les points peuvent être obtenus par approximation. C'est donc une vision locale du problème posé par l'équation $D \simeq [D \to D]$, qui est analogue aux méthodes utilisées en géométrie algébrique [MUM] où l'on travaille surtout aux voisinages des points. Une autre idée est celle qui consiste à se dire que l'on a résolu une équation aux espaces dans la catégorie des treillis complets (en fait des cpos) en utilisant des limites dans cette catégorie. Et de développer une théorie équationnelle des domaines, dans le cadre des catégories, où les sujets manipulés sont des espaces tout entiers. C'est donc une vision globale du problème. Elle est celle de Plotkin et Smyth [PLS] et de Hofmann 1979.

L'approche locale ou "faisceautique" que nous développons ici peut sembler plus proche des préoccupations immédiates du programmeur, à la recherche de "bons espaces" pour exprimer ses calculs, et dont les procédures, souvent, ne dépassent pas quelques niveaux de fonctionnalité.

Elle paraît aussi de nature à satisfaire le théoricien en lui permettant aussi d'exprimer l'approche globale. Il a pu sembler qu'en passant aux catégories l'aspect "reconstruction des points à l'intérieur de l'espace" était occulté et donc n'avait pas d'importance. En fait à chaque objet manipulé dans les catégories considérées on associe toujours une ou plusieurs façons d'obtenir cet objet comme limite à partir d'autres objets de la catégorie. On retrouve ainsi une notion de "reconstruction des objets à l'intérieur de la catégorie" analogue à celle de reconstruction des points. Les foncteurs utilisés sont ceux qui préservent ces limites, c'est-à-dire qui vérifient une propriété très proche de celle qu'on avait sur les fonctions. On retrouve donc la même notion.

On sait que tout ensemble partiellement ordonné peut être considéré comme une catégorie (cf. Mac Lane 1976, pp. 11); si donc on se restreint aux faisceaux ordonnés, on peut tout dire à l'intérieur de la théorie des catégories. Mais, outre qu'on gagne en généralité, il est une autre raison de rejeter une telle restriction, du moins de prime abord : c'est que l'exposé des problèmes est plus facile d'accès lorsqu'on adopte d'emblée une approche ensembliste, et qu'on n'emploie ainsi que des outils très usuels. Il nous a paru aussi plus didactique d'élucider en premier lieu le concept informatique, pour lui donner un moule mathématique convenable.

La suite de ce mémoire s'articule donc comme suit :

Le chapitre 0 contient un résumé sommaire des chapitres suivants .

Le chapitre I introduit la notion générale de faisceau et quelques-unes de ses propriétés.

Le chapitre II aborde les propriétés élémentaires des faisceaux ordonnés et en donne une classification.

Le chapitre III définit la notion de fonction régulière et examine la question de la construction d'une structure de faisceau sur ces espaces de fonctions régulières. On y présente deux méthodes utilisées, respectivement, pour la première fois par Scott et par Nolin.

Au chapitre IV on examine la question du calcul de limites de faisceaux et son lien avec la théorie équationnelle des domaines développées par Hofmann, Plotkin et Smyth. On y construit un domaine A_∞ qui contient des types en tant qu'objets et généralise le modèle E_∞ de Wadsworth.

CHAPITRE 0

Résumé

Nous reprenons les notions d'approximation définies par D. Scott 1969 treillis continus), L. Nolin 1971 (collections d'algorithmes), K.H. Hofmann 1978 (ordres approximants), A. Shamir et W. Wadge 1977 (structure de type sur un domaine), ADJ 1977 (posets inductifs), A. Arnold et M.Nivat 1978 (magmas métrique) etc..., en les présentant dans un cadre général qui permet de les placer les unes par rapport aux autres et d'étudier leurs propriétés. Nous espérons ainsi jeter les bases d'une théorie unifiée des modèles de sémantique des langages de programmation.

I. Faisceaux

La première notion d'approximation que nous examinons est celle qui permet d'obtenir des points comme limites d'autres points.

I.1. Notion de faisceau

Soit X un ensemble, $\mathfrak{F}(X)$ l'ensemble des familles d'éléments de X.

Un faisceau sur X est un couple $< \lambda, \beta >$ de fonctions :

- $\lambda : \mathfrak{F}(X) \to X$ partielle surjective notée presque toujours, si $(f : I \to X) \in \mathfrak{F}(X)$, $\lim\limits_{i \in I} f(i)$
- $\beta : X \to \mathcal{P}(\mathfrak{F}(X))$ injective

vérifiant la propriété de Fubini d'une part

$$\forall\, \omega \in \mathfrak{F}(X), \quad \omega : I \times J \to X,$$

si $\lim\limits_{i \in I} (\lim\limits_{j \in I} \omega(i,j))$ existe

(resp. si $\lim\limits_{j \in J} (\lim\limits_{i \in I} \omega(i,j))$ existe) alors

$\lim\limits_{(i,j) \in I \times J} \omega(i,j)$ existe aussi et on a l'égalité

$$\lim_{i \in I} (\lim_{j \in J} \omega(i,j)) = \lim_{(i,j) \in I \times J} \omega(i,j)) \text{ (resp.} \lim_{j \in J}(\lim_{i \in I} \omega(i,j)) = \lim_{(i,j) \in I \times J} \omega(i,j))$$

et la propriété de reconstruction des points d'autre part :

$$\forall x \in X \quad \forall \sigma \in \beta(x) \quad x = \lambda(\sigma) = \lim(\sigma) \quad \square$$

La valeur $\beta(x)$ de la fonction β au point x est appelée fibre de x.

La topologie de Scott sur un espace partiellement ordonné, les collections d'algorithmes de Nolin, les structures de type sur un domaine au sens de Shamir et Wadge, le magma métrique d'Arnold et Nivat, la notion de convergence utilisée dans Manna et Shamir 1977, la notion de convergence au sens de Kuratowski-Painlevé, l'utilisation qui est faite du λ-calcul en sémantique formelle des langages de programmation, définissent des structures de faisceau.

Les faisceaux sont clos par produit et coproduit et on peut également définir des notions de sous-faisceau et de faisceau-quotient.

I.2. Classes de faisceaux

Soit X un ensemble muni d'une structure de faisceau $<\lambda,\beta>$.

L'ensemble

$$N(X) = N_{\lambda,\beta}(X) = \{u \in X : \exists x \in X, \ \exists \sigma \in \beta(x), \ \sigma : I \to X, \ \exists i \in I \quad u = \sigma(i)\}$$

est appelé noyau du faisceau. Un élément $u \in X$ est rationnel ssi

$$\forall \sigma \in \beta(u) \quad \sigma : I \to X, \quad \nexists\, i \in \mathrm{Dom}(\sigma) \quad u = \sigma(i)$$

Le faisceau $<\lambda,\beta>$ est élémentaire ssi son noyau ne contient que des éléments rationnels.

Le faisceau $<\lambda,\beta>$ est ordonné ssi X est partiellement ordonné et λ prend les bornes supérieures (resp. les bornes inférieures) des familles d'éléments.

Le faisceau ordonné $<\lambda,\beta>$ est ordonné monique (au sens restreint) ssi il existe une fonction $s : X \to \mathcal{P}(X)$ croissante telle que

$$\forall x \in X \qquad \beta(x) = \{id_{s(x)} : s(x) \to s(x)\} .$$

Pour tout x $s(x)$ est appelé spectre de x. On note $a < b$ ssi $a \in s(b)$ (a approxime b).

Proposition 1.1. :

Les faisceaux ordonnés moniques (au sens restreint) dont les spectres sont des sections commerçantes du noyau coïncident avec les connections de Galois

$$X \underset{s}{\overset{\lambda}{\rightleftarrows}} T \subseteq \mathcal{P}(X)$$

t.q. $\lambda \circ s = id_X$.

I.3. Fonctions régulières

Soit X un ensemble muni d'un faisceau $<\lambda,\beta>$, Y muni d'un faisceau $<\lambda',\beta'>$. Une fonction $f : X \to Y$ est régulière au point x ssi
$\forall \sigma \in \beta(x) \quad f(x) = \lambda'(f(\sigma))$.

Les fonctions continues au sens de Scott, les algorithmes au sens de Nolin, les fonctions "tight" de Manna et Shamir 1977, etc... sont des fonctions régulières.

L'identité et les fonctions constantes sont toujours régulières. Les projections d'un produit de faisceaux sont régulières.

Une fonction de plusieurs variables est régulière ssi elle est régulière en chacune de ses variables (Prop. 1.4).

On note $[X \to Y]$ l'espace des fonctions régulières de X dans Y.
Si Y est ordonné on munit $[X \to Y]$ de l'ordre point par point (extensionnel)
$\forall f,y \in [X \to Y] \quad f \subseteq g$ ssi $\forall x \in X \quad f(x) \sqsubseteq g(x)$

Si X et Y sont des faisceaux, un faisceau sur l'espace $[X \to Y]$ est fidèle ssi $\forall f \in [X \to Y]$ $\forall \sigma \in \beta(f)$ (avec $\sigma : I \to [X \to Y]$, $i \to f_i$

$$\forall x \in X \quad \lim_{i\,Y} f_i(x) = f(x).$$

Si les espaces fonctionnels sont munis de faisceaux fidèles alors l'abstraction est une fonction faiblement régulière. ($g : A \to B$ est faiblement régulière ssi $\forall x \in A$ $\forall \sigma \in \beta(x)$ $\lim_B f(\sigma)$ existe $\Rightarrow f(x) = \lim_B(f(\sigma))$).

Par contre si les espaces fonctionnels sont munis de faisceaux fidèles, l'application (ou évaluation) est toujours régulière.

II. Faisceaux ordonnés

Tout faisceau ordonné élémentaire définit un ordre élémentaire (cf. Nait-Abdallah 1978) et réciproquement. Tout cpo algébrique possède une structure de faisceau ordonné monique (f.o.m) élémentaire (2.8).

Si s_1 et s_2 sont des fonctions de X dans $\mathcal{P}(X)$ définissant des f o m sur X, on dit que le f o m défini par s_1 est plus fin que le f o m défini par s_2 ssi

$$\forall x \in X \qquad s_1(x) \subseteq s_2(x).$$

Si X est un ensemble partiellement ordonné, l'ensemble des f o m (au sens restreint) sur X ordonné par la relation "être plus fin que" a une structure de semi-treillis complet dont le "bottom" est constitué par le faisceau

$$s : x \to \downarrow x = \{y \in X : y \sqsubseteq x\}$$

(Prop. 2.11)

Un faisceau $\langle \lambda, \beta \rangle$ sur un ensemble X est <u>interpolable au point $e \in X$</u> ssi $\forall x \in X$ on a

$$\forall \sigma \in \beta(x)$$
$$[\exists i \in \mathrm{Dom}(\sigma)\ e = \sigma(i)] \iff [\exists \ell \in \mathrm{Dom}(\sigma),\ u = \sigma(\ell),$$
$$\forall \tau \in \beta(u) \quad \exists j \in \mathrm{Dom}(\tau) \quad e = \tau(j)] .$$

Dans le cas monique (restreint) ceci se réduit à :

$$e \in s(x) \iff \exists\, u \in s(x) \qquad e \in s(u).$$

Les treillis continus, les faisceaux (ordonnés) élémentaires sont interpolables en tout point approximant. Les faisceaux (ordonnés) sont interpolables en tout point rationnel.

Un faisceau <u>ordonné</u> est <u>stable au point $x \in X$</u> ssi $\forall y \in X$, $\forall a \in N(X)$ (<u>noyau</u> de X) $a \sqsubseteq x \subseteq y$ implique l'équivalence des deux assertions suivantes :

(i) $\exists\, \sigma \in \beta(x) \quad \exists\, i \in \mathrm{Dom}(\sigma) \quad a = \sigma(i)$

(ii) $\exists\, \tau \in \beta(y) \quad \exists\, j \in \mathrm{Dom}(\tau) \quad a = \tau(j)$.

Dans le cas monique ceci se réduit à

$$\forall y \in X \qquad x \sqsubseteq y \implies s(x) = ({\downarrow} x) \cap s(y).$$

Le faisceau SW est stable; le faisceau N est stable ssi les rationnels sont incomparables entre eux.

On s'aperçoit alors que tout f o m qui est une connection de Galois est stable. Tout f o m sur un espace dirigé X qui n'est pas une connection de Galois possède un point d'instabilité (lemme 2.16). En ce qui concerne les f o m la stabilité et l'interpolabilité se relient de la façon suivante.

<u>Lemme 2.17</u> : Soit X un poset muni d'un f o m,

$x \in X$ et ${\uparrow} x = \{z \in X : x \in s(z)\}$.

Si les spectres des $y \in {\uparrow} x$ sont dirigés et si le f o m est stable sur ${\uparrow} x$, alors le f o m est interpolable au point x. □

La notion de treillis continu au sens de Scott s'insère de façon élégante dans ce cadre.

<u>Proposition 2.18</u> : Soit X un poset, $\mathcal{P}(X)$ l'ensemble de ses parties ordonné par inclusion, munis tous deux de leur topologie de Scott, $s : X \to \mathcal{P}(X)$ définissant un f o m sur X. On a :

(i) Si $s = s_1 : x \to \cap\{I \in Id_{cv}\, X : x \sqsubseteq \cup I\}$

et si le faisceau est interpolable, alors s est une fonction continue en chaque point pour les topologies de Scott de X et $\mathcal{P}(X)$.

(ii) Si s est continue en chaque point pour les topologies de Scott de X et $\mathcal{P}(X)$ alors

$$\forall x \in X \quad s(x) \subseteq \cap \{I \in Id_{cv} X : x \in \cup I\} \quad \square$$

III. Espaces de fonctions régulières

Toute fonction régulière sur un f o m est monotone. Et si X est un f o m (au sens restreint), Y un sup semi treillis complet, alors toute fonction régulière $f : N(X) \to Y$ admet un unique prolongement régulier (3.5)

La régularité de la composition de deux fonctions régulières exige des conditions particulières (3.7), mais on peut définir la "composition régulière" suivante

$$o : N[X \to Y] \times N[Y \to Z] \times N(X) \to Z$$

$$o = \lambda f \in N[X \to Y].\lambda g \in N[Y \to Z].\ \lambda u \in N(X).g(f(u))$$

qui généralise la composition usuelle et tient mieux compte des structures de faisceaux particulières qu'on manipule. De plus, dans le cas donné, si X est un f o m élémentaire, si $[X \to Y]$ et $[Y \to Z]$ sont des f o m fidèles et élémentaires, si Z est un espace partiellement ordonné complet sous condition, alors l'application o est régulière sur l'espace $[X \to Y] \times [Y \to Z] \times X$ muni du faisceau produit. En particulier la composition o fournit, à partir de fonctions régulières, des fonctions régulières (Prop. 3.8). L'associativité de o est assurée dans certains cas (3.9).

III.1. Faisceaux sur des espaces de fonctions régulières

Si X et Y sont des faisceaux ordonnés tels que Y est normalisé, le faisceau de la convergence simple sur $[X \to Y]$, lorsqu'il existe, est défini par la relation

$$f < g \iff (\forall x \in X \quad f(x) < g(x)) \iff$$
$$(\forall x \in X \quad f(x) \in s(g(x)))$$

Le problème de munir un espace de fonctions régulières sur des faisceaux ordonnés, d'une structure de faisceau, se pose alors sous sa formulation la plus générale. Pour résoudre ce problème dans un certain nombre de cas pertinents en informatique, on examine quelques méthodes inspirées par les échelons de Scott, les échelons de Nolin, les faisceaux continus. La construction appelée "powerdomain" par Smyth et Plotkin, vue sous cet angle, prend un relief particulier et se relie à la sémantique dénotationnelle d'Arnold et Nivat.

Si D et D' sont des faisceaux ordonnés $e \in D$, $e' \in D'$ on définit l'échelon de Scott (ou échelon inférieur) $[e,e']_m$ (avec $m \subseteq e'$) par

$$[e,e']_m \quad x \mapsto \begin{cases} e' & \text{si } \forall \sigma \in \beta(x)\ \exists i \in \mathrm{Dom}(\sigma) \quad e = \sigma(i) \\ m & \text{sinon} \end{cases}$$

et l'échelon de Nolin (ou échelon supérieur) $\langle e,e'\rangle_m$ (avec $e' \subseteq m$) par :

$$\langle e,e'\rangle_m : x \mapsto \begin{cases} e' & \text{si } x \subseteq e \\ m & \text{sinon.} \end{cases}$$

Tout échelon de Nolin est une fonction régulière.

α. Méthode des échelons de Scott

a) Cas monique : Si D et D' sont des faisceaux ordonnés moniques, toute fonction régulière $f : D \to D'$ minorée est borne supérieure d'une famille d'échelons de Scott (3.12), et un échelon de Scott $[e,e']_m$ est régulier si et seulement si D est interpolable en e. D'où l'on tire, si D est interpolable, une structure de faisceau sur $[D \to D']_*$, avec

$[D \to D']_* = \{f \in [D \to D'] \ : f$ est minorée$\}$,

dont les éléments approximants sont les échelons de Scott. Et par la même occasion, si nous prenons :

$$[D \to D']_{**} = \{f \in [D \to D'] : \exists m(f) \in D' \text{ tq } \forall x \in D \quad m(f) < f(x)\}$$

nous obtenons un faisceau de la convergence simple sur $[D \to D']_{**}$ (Corollaire 3.14).

Cette construction peut être itérée dans les conditions suivantes :

Proposition 3.16. : Soit D,D' des f o m. Supposons D interpolable et $[D \to D']_{**}$ comme ci-dessus. Munissons $[D \to D']_{**}$ du faisceau de la convergence simple et supposons de plus que D' vérifie les conditions suivantes

(i) il est stable

(ii) ses spectres sont des sup de mi treillis.

Alors le f o m $[D \to D']_{**}$ est interpolable et vérifie les deux conditions précédentes (i) et (ii).

En fait si D est élémentaire (ce qui est un cas particulier d'interpolabilité) on peut imposer, dans les conditions du Corollaire 3.14. certaines propriétés de canonicité à la décomposition des fonctions minorées en échelon de Scott (Prop. 3.17)

b) Cas général : La méthode se déduit du cas monique. Si D et D' sont des faisceaux ordonnés, l'échelon de Scott $[e,e']_m$ est régulier ssi D est interpolable au point e (3.18). Si la fibre $\beta(x)$ de chaque $x \in D$ possède un élément minimal au sens des prolongements (3.19), alors toute fonction régulière $f : D \to D'$ minorée est borne supérieure d'une famille d'échelons de Scott.

On peut opérer de manière analogue en utilisant la notion de faisceau ordonné stable : Si D et D' sont des faisceaux ordonnés tq D' est stable, toute fonction $f : D \to D'$ régulière, monotone, minorée est borne supérieure d'une famille d'échelons de Scott.

D'où une structure de faisceau sur cette classe de fonctions régulières en rajoutant l'interpolabilité de D (Proposition 3.21). On note qu'on obtient ainsi un faisceau monique sur l'espace des fonctions, ce qui nous ramène au cas précédent si nous voulons itérer.

β. Méthode "Continuité de l'espace des images"

a) Cas monique : Si D et D' sont deux f o m interpolables, où D' est continu au sens de Scott et contient $\perp$, alors l'espace $[D \to D']$ possède une structure de faisceau continu et contient $\perp$.

b) Cas général : Dans le cas général entre en jeu une autre façon de généraliser la notion d'interpolabilité restreinte aux f o m ; nous l'appelons l'interpolabilité globale.

Un faisceau sur un ensemble X est globalement interpolable au point $e \in X$ ssi pour tout $u \in X$ nous avons

$$[\forall \tau \in \beta(u) \quad \exists k \in \mathrm{Dom}(\tau) \quad e = \tau(k)] \iff$$
$$\exists x \in X \{ (\forall \sigma \in \beta(x) \;\exists i \in \mathrm{Dom}(\sigma) \quad e = \sigma(i)) \wedge$$
$$\wedge (\forall \tau \in \beta(u) \;\exists j \in \mathrm{Dom}(\tau) \quad x = \tau(j))\}$$

Il vient alors :

Proposition 3.24. : Soit D un faisceau ordonné interpolable et globalement interpolable, D' un faisceau continu au sens de Scott, interpolable et possédant $\perp$. Alors l'espace $[D \to D']$ des fonctions régulières monotones, ordonné extensionnellement, possède une structure de faisceau continu et contient $\perp$. □

Y. Méthode des échelons de Nolin :

Si D et D' sont deux faisceaux ordonnés, $[D \to D']_{max}$ l'espace des fonctions régulières de D dans D' qui sont majorées et monotones croissantes, alors toute fonction de $[D \to D']_{max}$ est borne inférieure d'une famille d'échelons de Nolin (3.27). Si D est élémentaire, on peut donner une définition plus "canonique" de cette famille (3.28), analogue à ce que l'on avait dans le cas des échelons de Scott.

III.2. Sémantique d'un langage de programmation avec déclarations de types, et rétractes

Il a été indiqué (cf. [NOL] , [SHA]) que les types peuvent être approchés en tant qu'objets de calcul. Nous utilisons la notion de faisceau ordonné pour préciser et développer ce point de vue et construire une sémantique du plus grand point fixe des programmes typés

$$F\ \alpha\ \beta\ :\ G(x) \Leftarrow \tau\,[G]\ (x)$$

Appelons encore τ la fonctionnelle associée (par composition régulière) au terme $\tau\,[G]$.

Proposition 3.39 : Soit X est un f o m élémentaire, distributif, complet sous condition, $f_o \in [X \to X]$. Alors si la suite $\{\tau^n(f_o)\}_{n \in \mathbb{N}}$ est convergente sur $D \subseteq X$, pour le faisceau associé à la topologie de Scott pour l'ordre inverse sur $[X \to X]$, $\prod_n \tau^n(f_o)$ est solution sur D de l'équation fonctionnelle $g = \tau(g)$. Si $f_o = \lambda x.T$, $\prod_n \tau^n(f_o)$ est le plus grand point fixe sur D de τ . □

On appelle domaine commode un f o m élémentaire X, complet sous condition, possédant T et possédant la propriété de distributivité :

$$\forall a \in X \quad \forall S \subseteq X \quad a \sqcap (\sqcup\{s : s \in S\}) = \sqcup\{a \sqcap s : s \in S\}$$

A la déclaration de type dans la définition récursive ci-dessus, on associe une transformation fonctionnelle :

$N_{a,b}$: $\tau \to \lambda f.\,\lambda x.\,< a,b \sqcap \tau(f)(x) >_T(x)$ où a (resp. b) interprète α (resp. β) qui décrit le filtrage opéré par la vérification de types dans un langage comme ALGOL ou FORTRAN.

Si X est un domaine commode et si $[X \to X]$ est muni d'un faisceau fidèle, $N_{a,b}$ transforme des fonctionnelles régulières en des fonctionnelles régulières (Prop. 3.40).

La fonction calculée par le programme

$$F\ \alpha\ \beta :\ G(x) \Leftarrow \tau\ [G\ ](x)$$

sera alors le plus grand point fixe de la fonctionnelle régulière $\tau' = N_{a,b}(\tau)$.

Quelle relation y a-t-il avec la théorie des rétractes de D. Scott ?

Si X est un faisceau, appelons rétraction sur X toute fonction régulière $r : X \to X$ tq. $r \circ r = r$; r définit alors le rétracte $r(X)$.

Supposons que tous les espaces fonctionnels sont munis de faisceaux fidèles.

Lemme 3.42. : Soit X un faisceau, Y un domaine commode $a \in X$, $b \in Y$. Alors

(i) la transformation $N_{a,b}$ est une rétraction sur l'espace des transformations régulières

(ii) il n'existe pas de rétractions $r : X \to X$, $t : Y \to Y$ tq. $\forall f \in [X \to Y]$ $N_{a,b}(f) = t \circ f \circ r$ (i.e. tq. $N_{a,b} = t \circ\!\!\to r$)

La rétraction $N_{a,b}$ et le rétracte de Scott $u \circ\!\!\to v$ jouent donc un rôle distinct.

IV. Faisceaux sagittaux et limites (théorie globale)

Une deuxième forme de la notion d'approximation est celle qui permet de reconstruire des espaces comme limites d'autres espaces. Soit U un univers fixé. Les faisceaux considérés auront tous leur support dans U.

Si X et Y sont des faisceaux, une fonction $f : X \to Y$ régulière est fortement régulière si et seulement si

$$\forall x \in X\ \ \forall \sigma \in \beta(x)\ \ (f \circ \sigma) \in \beta(f(x))$$

La catégorie Fais qui a pour objets les faisceaux et pour morphismes les fonctions fortement régulières a des équaliseurs, coequaliseurs, produits et coproduits. Elle est donc complète et co-complète.

Il en est de même pour les catégories de faisceaux Foen, Ups, Dcont, DAlg,... Ceci indique comment obtenir certains objets comme limites (ou co-limites) de diagrammes. Nous appelons faisceau sagittal cette généralisation de la notion de faisceau. La théorie des catégories n'intervient ici que pour nous fournir un langage commode.

L'exponentiation de faisceaux peut se faire de diverses manières (Prop. 4.4. 4.5.).

Pour manipuler convenablement la notion de type nous utilisons une double structure de faisceau sur un espace, ou <u>bifaisceau</u>.

Il existe une exponentiation particulière sur les bifaisceaux qui est obtenue avec les échelons de Nolin :

<u>Proposition 4.20.</u> : Soit <u>BFt</u> la catégorie dont les objets ont pour faisceau inférieur une structure de <u>Foen</u>, et faisceau supérieur une structure de <u>D Alg</u>, et sont munis d'un τ , et dont les flèches sont les fonctions régulières simples pour le faisceau inférieur.

Alors

$$N : \underline{Foen} \times \underline{BFt} \to \underline{D\ Alg}$$
$$(X,Y) \to [X \to Y]$$

(défini sur les objets par : $[X \to Y]$ est muni du faisceau algébrique défini au moyen de Prop. 2.18 par la fonction

$$\forall f \in [X \to Y] \quad s : f \to s(f)$$

$s(f) = \{\langle e,e'\rangle_\tau : e' \geq f(e)$, e rationnel et e' compact pour le faisceau supérieur de Y$\}$) est un foncteur. □

En utilisant la notion de bifaisceau commode (qui ne sont rien d'autre que des domaines commodes munis d'un faisceau supérieur qui est dans <u>D Alg</u>), on construit un domaine avec types A_∞ , généralisant le domaine E_∞ de Wadsworth :

<u>Théorème 4.27</u> : Soit D un bifaisceau commode, alors le schéma d'approximation de Wadsworth fournit dans <u>Set</u> une limite projective A_∞ qui est telle que

(i) $A_\infty = D + \varprojlim \Delta_k$, la limite des Δ_k pouvant être calculée dans <u>D Alg</u>, ce qui munit A_∞ d'une structure de faisceau

(ii) Si $[A_\infty \to A_\infty]_P$ est l'espace des fonctions régulières partielles de A_∞ dans A_∞ alors il existe une bijection

$$\varprojlim \Delta_k \leftrightarrow [A_\infty \to A_\infty]_P .$$

En particulier A_∞ contient l'espace de toutes les fonctions régulières de A_∞ dans A_∞ . □

Notations et définitions générales

Un <u>univers</u> est un ensemble $\mathcal{U}$ possédant les proprietes suivantes :

(i) $x \in u \in \mathcal{U} \Longrightarrow x \in \mathcal{U}$

(ii) $u \in \mathcal{U}$ et $v \in \mathcal{U}$ impliquent $\{u,v\}$, $\langle u,v \rangle$ et $u \times v \in \mathcal{U}$

(iii) $x \in \mathcal{U}$ implique $P(x) \in \mathcal{U}$ et $\bigcup x \in \mathcal{U}$

(iv) Si $\omega = \{0,1,2,\ldots\}$ est l'ensemble des ordinaux finis, alors $\omega \in \mathcal{U}$

(v) Si $f : a \to b$ est une surjection telle que $a \in \mathcal{U}$ et $b \subset \mathcal{U}$ alors $b \in \mathcal{U}$

Si X est un ensemble et $\mathcal{U}$ un univers, on appellera <u>ensemble des familles d'éléments de X</u> l'ensemble de fonctions

$$F(X) = \{f : I \to X \mid I \in \mathcal{U}\}.$$

Un ordre partiel $\leq$ sur un ensemble X est une relation binaire qui est reflexive, transitive et antisymétrique. Si X est un ensemble partiellement ordonné (ou poset), pour tout $x \in x$ nous définissons la <u>section commencante principale</u> :

$$\downarrow x = \{y \in X : y \leq x\}$$

Une partie $S \subseteq X$ est <u>dirigée</u> ssi

$$\forall y,x \in S \quad \exists z \in S \quad z \geq x,y$$

Un <u>idéal</u> $I \subseteq X$ est une partie dirigée telle que $\forall x \in I \downarrow x \subseteq I$. Si X est un poset, a chaque partie $S \subseteq X$ nous associons, si elle existe, sa <u>borne supérieure</u> $\bigsqcup S$ (resp. sa <u>borne inférieure</u> $\bigsqcap S$). Ceci définit des fonctions partielles $\bigsqcup$, $\bigsqcap$. Chaque fois que nous considérons un idéal, nous supposons qu'il est <u>convergent</u>, i.e. a une borne supérieure. Nous notons $Id_{cv}(X)$ l'ensemble des idéaux convergents de X.

CHAPITRE I

FAISCEAUX

°°°

I - NOTION DE FAISCEAU

Intuitivement nous allons définir un faisceau comme la description de toutes les façons possibles de reconstruire les éléments d'un espace comme limites. Dans la suite on appellera $\mathfrak{F}(X)$ l'ensemble des familles d'éléments de X; i.e. des applications $\sigma : J \to X : j \mapsto x_j$. ($X$ est ici un ensemble).

DEFINITION : Soit X un ensemble. Un faisceau sur X est un couple de fonctions $\langle \lambda, \beta \rangle$

$\lambda : \mathfrak{F}(X) \to X$ (partielle) surjective (parfois notée lim)

$\beta : X \to \mathcal{P}(\mathfrak{F}(X))$ injective

où λ vérifie :

1. la propriété "à la Fubini" suivante :

$\forall$ famille $\{\omega_{ij}\}_{i \in I, j \in J} \subseteq X$ si on a que

$\lambda(\{\lambda(\{\omega_{ij} : j \in J\}) : i \in I\})$ existe (respectivement
$\lambda(\{\lambda(\{\omega_{ij} : i \in I\}) : j \in J\})$ existe) alors la valeur
$\lambda(\{\omega_{ij} : i \in I, j \in J\})$ existe aussi et on a l'égalité

$$\lambda(\{\omega_{ij} : i \in I, j \in J\}) = \lambda(\{\lambda(\{\omega_{ij} : j \in J\}) : i \in I\})$$
$$(\text{resp. } \lambda(\{\omega_{ij} : i \in I, j \in J\}) = \lambda(\{\lambda(\{\omega_{ij} : i \in I\}) : j \in J\}))$$

2. la propriété de reconstruction des points suivante :

$$\forall x \in X \quad \forall \sigma \in \beta(x) \quad x = \lambda(\sigma)$$

Remarque 1 : On sait qu'en théorie de l'intégration le théorème de Fubini s'énonce de la manière suivante :

si $(x,y) \mapsto f(x,y)$ est une fonction mesurable et non-négative sur le pavé $\Delta = \{(x,y) = a \leq x \leq b,\ \ c \leq y \leq d\}$, alors l'existence de l'intégrale itérée

$$\int_a^b \left[\int_c^d f(x,y)dy\right] dx$$

implique que $f(x,y)$ est sommable sur Δ et on a l'égalité

$$\int_a^b \left[\int_c^d f(x,y)dy\right] dx = \iint_\Delta f(x,y)dx\ dy.$$

L'analogie des rôles joués par le signe $\int$ et le symbole λ est à l'origine de notre terminologie. $\square$.

Remarque 2 : La définition précédente n'utilise que des flèches. Si on se restreint aux fonctions λ qui vérifient la propriété de Fubini, on peut l'exprimer alors en disant que la diagramme suivant est commutatif :

$$\begin{tikzcd} X \arrow[rr, "\beta"] \arrow[dr, "\{\ \}"'] & & \mathcal{P}(\mathfrak{F}(X)) \arrow[dl, "\mathcal{P}\lambda"] \\ & \mathfrak{F}(X) & \end{tikzcd}$$

avec $\{\ \} : x \mapsto (\{x\} \mapsto \{x\})$ et $\mathcal{P}\lambda(S) = \{\lambda(s) : s \in S\}$. $\square$

Terminologie : $\forall x \in X$ $\beta(x)$ est la fibre de x, un élément $\sigma \in \beta(x)$ est un système approximant de x. L'ensemble

$N_{\lambda,\beta}(X) = \cup\{\mathrm{Im}(\sigma) : \sigma \in \beta(x),\ x \in X\}$ est le noyau de faisceau $\langle\lambda,\beta\rangle$. La fonction λ (ou lim) sera souvent appelé limite, et on notera

$$\lambda(\{s_i : i \in I\}) = \lim_{i \in I} s_i$$

comme en analyse réelle par exemple (s'il y a ambiguité sur l'espace concerné, on notera

$$\lim_X = \lim : \mathfrak{F}(X) \to X$$

Ainsi la propriété de Fubini peut s'écrire plus simplement :

$\forall$ famille $\{\omega_{ij}\}_{i \in I, j \in J} \subseteq X$ dès que $\lim_i (\lim_j \omega_{ij})$ (respectivement $\lim_j (\lim_i \omega_{ij})$) existe alors $\lim_{ij} \omega_{ij}$ existe aussi et on a l'égalité.

On appellera <u>ω-faisceau</u> un faisceau dont les systèmes approximants sont des <u>suites</u>, i.e. des applications $\sigma : \mathbb{N} \to X$

Si $x \in X$, on peut définir un préordre sur les systèmes approximants $\sigma \in \beta(x)$ par :

$$\sigma \leq \sigma' \iff \mathrm{Im}(\sigma) \subseteq \mathrm{Im}(\sigma')$$

En passant à l'ordre partiel quotient, s'il existe un plus grand élément, on appellera <u>spectre de x $s(x)$</u> ce plus grand élément (dans des conditions que nous éclaircissons dans la remarque 3 suivante).

Un représentant quelconque de cette classe d'équivalence, sera appelé système <u>approximant spectral</u> (en abrégé <u>sas</u>) de x et nous noterons <u>$ss(x)$</u> un tel objet.

<u>Exemples</u> : Quelques exemples de faisceaux sont donnés par :

(i) <u>topologie de Scott sur un espace partiellement ordonné</u> : Soit X un poset muni de la topologie de Scott. Alors on peut définir canoniquement sur X le faisceau appelé <u>faisceau S</u>

$\lambda : \mathcal{G}(X) \to X$ par

$\sigma \mapsto \mathrm{Im}(\sigma) \xrightarrow{\sqcup} \sqcup(\mathrm{Im}(\sigma)) = \lambda(\sigma)$ où σ est une famille, dans le cas où la borne supérieure $\sqcup(\mathrm{Im}(\sigma)) = \sqcup\{x_j : j \in J\}$ $(x_j = \sigma(j)\ \forall j)$ existe

$\beta : X \to \mathcal{S}(\mathcal{G}(X))$

$x \mapsto \{S : S$ est une famille dirigée d'éléments de X et $x = \sqcup S\}$

λ est surjective, β injective et la propriété de reconstruction des points est vérifiée. (En fait la propriété de reconstruction des points entraîne toujours l'injectivité de β et la surjectivité de λ ; ces deux propriétés ont été incluse dans la définition pour aider l'intuition). De plus $\lambda = \sqcup \circ \mathrm{Im}$ possède bien la propriété de Fubini, car

$$\bigsqcup_{ij} \omega_{ij} = \bigsqcup_i (\bigsqcup_j \omega_{ij}) = \bigsqcup_j (\bigsqcup_i \omega_{ij})$$

En effet $a \sqsupseteq \bigsqcup_{ij} \omega_{ij} \Leftrightarrow \forall i \ \forall j \quad a \sqsupseteq \omega_{ij}$, et

$$a \sqsupseteq \bigsqcup_i \Big(\bigsqcup_j \omega_{ij}\Big) \Leftrightarrow \forall i \quad a \sqsupseteq \bigsqcup_j \omega_{ij} \Leftrightarrow$$

$$\forall i \ \forall j \quad a \sqsupseteq \omega_{ij} \Leftrightarrow a \sqsupseteq \bigsqcup_{ij} \omega_{ij} .$$

Donc, si $\omega : (i,j) \mapsto \omega(i,j) = \omega_{ij}$

$a = \bigsqcup_{ij} \omega_{ij} \Leftrightarrow a \sqsupseteq \bigsqcup_{ij} \omega_{ij}$ et c'est le plus petit

$\Leftrightarrow a \sqsupseteq \bigsqcup_i \big(\bigsqcup_j \omega_{ij}\big)$ et c'est le plus petit

$\Leftrightarrow a = \bigsqcup_i \big(\bigsqcup_j \omega_{ij}\big) \Rightarrow \lambda_{(i,j) \in I \times J}(\omega) = \lambda_{i \in I}(\lambda_{j \in J} \cdot \omega)$

NB1 : Dans l'exemple précédent on note que la limite λ n'utilise que la partie $\mathrm{Im}(\sigma)$ (image de σ) des systèmes approximants σ. On aurait alors pu se contenter, en simplifiant la notion de faisceau, de prendre

$$\lambda : \mathcal{P}(X) \to X$$
$$\beta : X \to \mathcal{P}(\mathcal{P}(X))$$

avec la même définition modulo la factorisation par Im.

NB2 : Noter que ceci a été fait pour un poset X muni de la topologie supérieure de Scott. On peut faire la même chose si X est muni de la topologie inférieure de Scott, en remplaçant $\sqcup$ par $\sqcap$ et les parties dirigées par les parties filtrantes. (On obtient alors le faisceau Sinv).

(ii) collections d'algorithmes de Nolin : (faisceau N)

Supposons qu'il existe un ensemble X qui ait la structure d'une collection d'algorithmes au sens de Nolin, munie de prépartitions comme spectres (cf. version non publiée de la théorie de Nolin (cours de DEA)). Alors on peut prendre :

$\lambda : \mathfrak{S}(X) \to X$ (faisceau N)

$X \sqsupseteq S \mapsto \bar{\sqcup} S$ (union complétée sur X)

$\beta : X \longrightarrow \mathcal{P}(\mathcal{F}(X))$

$x \longmapsto \{S \subseteq X : S$ est une prépartition de x en éléments atomiques$\}$.

La propriété de reconstruction des points est donnée d'emblée. La propriété de Fubini se déduit du cas précédent en notant que X muni de l'inclusion est un poset. (Ne pas oublier que les éléments de X sont des parties). On peut noter ici que $\forall x \in X$, la fibre $\beta(x)$ est réduite aux sas $ss(x)$. En effet : soient $\sigma, \sigma' \in \beta(x)$, $S = \mathrm{Im}(\sigma)$, $S' = \mathrm{Im}(\sigma')$.

Alors $\forall s \in S \quad s \subseteq x = \bigcup_{s' \in S'} s' \Rightarrow \forall s \in S \ \exists s' \in S' \quad s \subseteq s'$

car s est atomique. De même

$$\forall s' \in S' \qquad \exists s \in S \qquad s' \subseteq s$$

Donc
$$\forall s_1 \in S \qquad \exists s' \in S' \ \exists s_2 \in S \qquad s_1 \subseteq s' \subseteq s_2$$

$$\text{d'où} \qquad s_1 \subseteq s_2$$

Comme S est une prépartition (voir la définition dans les prolégomènes) il vient $s_1 = s_2 = s'$. Donc $S = S'$, donc

$$\forall x \in X \quad \forall \sigma, \sigma' \in \mathcal{F}(X) \ , \sigma, \sigma' \in \beta(x) \Rightarrow \mathrm{Im}(\sigma) = \mathrm{Im}(\sigma')$$

On a Noyau $(X) = \{$ atomiques $\}$.

Noter qu'on peut aussi définir, au sens de Proلég. 3 pp. 30 (clause (iii) le <u>faisceau</u> $N_2 = \langle \lambda, \beta' \rangle$ où

$$\beta' : x \mapsto \{\{y \in X : y \text{ atomique} \subseteq x\}\},$$

l'ensemble $\in \beta'(x)$ étant indexé par lui-même.

(iii) <u>structure de types sur un domaine, d'après Shamir et Wadge</u> : <u>(faisceau SW)</u>.

Soit D un domaine, i.e. un cpo ; $\hat{D}$ une structure de types ainsi définie par Shamir et Wadge :

$\hat{D}$ est une partie de $\mathcal{P}(D)$ telle que

(i) $\quad \forall d \in D \qquad \downarrow d \in \hat{D}$

(ii) $\quad D \in \hat{D}$

(iii) $\quad \hat{D}$ est clos par intersection infinie

(iv) $\forall x \in \hat{D}$ $x = \downarrow x$ et x contient les bornes supérieures de ses parties dirigées (i.e. est fermé pour la topologie de Scott).

On peut noter que sur un poset, tous les fermés au sens de la topologie de Scott sont des sections commençantes (i.e. des lowersets) car sinon soit $F \subseteq D$ fermé, soit $u \in \downarrow F$ tel que $u \notin F$, alors $u \in \complement F$ qui est ouvert. Comme $\complement F$ est une section finissante (upper-set), il vient $\exists f \in F$ $f \in \complement F \Rightarrow$ absurde.
D'autre part

$$\forall d \in D \qquad \bar{d} = \downarrow d \quad \text{(fermeture d'un point)}$$

D'où si on note $F(D)$ = le treillis des fermés de D au sens de la topologie de Scott, ordonné par inclusion, l'énoncé de la définition précédente peut se simplifier en disant qu'une structure de type $\hat{D}$ est un sous-treillis complet de $F(D)$ clos par intersection infinie, contenant D et les fermetures des points de D.

Ceci détermine le faisceau (noter qu'on peut "remplacer" ici $\mathfrak{F}(D)$ par $\mathcal{G}(D) \subseteq \mathfrak{F}(D)$) :

$$\begin{array}{ll} \lambda : \mathfrak{F}(\hat{D}) \to \hat{D} & \text{(faisceau SW)} \\ \quad S \longmapsto \sqcup S = \bar{\cup} S & \text{(union complétée)} \\ \beta : \hat{D} \longrightarrow \mathcal{G}(\mathfrak{F}(\hat{D})) & \\ \quad x \mapsto \{\{\downarrow d : d \in x\}\} , \quad \{\downarrow d : d \in x\} & \text{étant indexé par lui-même.} \end{array}$$

Ici encore la fibre de chaque point est réduite à un seul élément, le spectre.

(iv) <u>magma métrique d'Arnold et Nivat 1978</u> :

Soit X le magma métrique <u>complété</u> noté $\overline{M}(F,V)$ par ces auteurs. On a le <u>faisceau AN</u>

$$\begin{array}{l} \lambda : \mathcal{G}(X) \to X \\ \quad (x_n)_{n \in \mathbb{N}} \longmapsto \lim_n x_n \quad \text{si la suite } (x_n) \text{ est convergente} \\ \beta : X \to \mathcal{G}(\mathfrak{F}(X)) \\ \quad x \to \{(y_n)_{n \in \mathbb{N}} : (y_n) \text{ est de Cauchy et } x = \lim_n y_n\} \end{array}$$

Montrons la propriété de Fubini :

Supposons que $\lim\limits_{n} (\lim\limits_{m} w_{nm}) = a$ i.e. existe.

Ceci implique-t-il

$$a = \lim_{m}(\lim_{m} w_{nm}) = \lim_{n,m} w_{nm} \quad ?$$

l'hypothèse est équivalente à :

$$\forall \varepsilon > 0 \quad \exists N_\varepsilon \quad n > N_\varepsilon \implies d(a, \lim_{m} w_{n,m}) < \varepsilon$$

$$\iff \forall \varepsilon > 0 \quad \exists N_\varepsilon \quad n > N_\varepsilon \implies d(a, b_n) < \varepsilon \quad \text{avec}$$

b_n vérifiant :

$$\forall \eta > 0 \quad \exists M_\eta \quad m > M_\eta \implies d(b_n, w_{nm}) < \eta$$

Soit maintenant $\varepsilon' > 0$ tel que $\varepsilon = \frac{\varepsilon'}{3}$ et $\eta = \frac{\varepsilon'}{3}$. Alors

$$d(a, w_{nm}) < d(a, b_n) + d(b_n, w_{nm}) < \frac{2\varepsilon'}{3} < \varepsilon' .$$

q.e.d.

Donc w_{nm} converge vers a.

On a donc bien une structure de <u>ω-faisceau</u> sur X

(v) <u>la notion de convergence utilisée dans Manna et Shamir 1977</u> :

<u>(faisceau MS)</u> :

Si $(x_n)_{n \in \mathbb{N}}$ est une suite sur l'espace X, on dira que (x_n) converge vers $a \in X$ ssi

$$\exists n_0 \in \mathbb{N} \quad \forall n \geqslant n_0 \quad x_n = a$$

i.e. ssi la suite est stationnaire en a. Si on ne s'occupe pas du contenu de l'espace X (un cpo plat dans le cas présent), ce cas est un cas particulier du précédent : il suffit de voir que X est métrique pour la distance discrète :

$$d(x,y) = 1 \quad \text{si} \quad x \neq y \text{ , et } \quad 0 \quad \text{si} \quad x = y$$

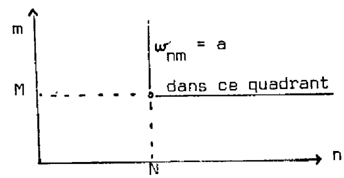
Les seules suites convergentes pour cette distance sont les suites stationnaires et l'espace est évidemment complet.

La notion de faisceau permet donc de relier ces deux constructions, apparemment sans rapport.

Que peut-on dire de la réciproque de la propriété de Fubini dans ce cas particulier ?

$$a = \lim_{m,n} w_{n,m} \Leftrightarrow (\exists N,M \quad (n,m) \geq (N,M) \Rightarrow w_{NM} = w_{nm})$$

Fixons $n = N$; on peut dire de w_{Nm} qu'elle est stationnaire à partir du rang M, et vaut a. En fait, à part un nombre fini (N par exemple), toutes les suites $\{w_{nm} : n \text{ fixé}\}$ sont stationnaires en a. Ceci peut être visualisé par le diagramme :



Donc si (w_{nm}) est convergente, et si l'on modifie un nombre fini de sous-suites $\{w_{nm} = n \text{ fixé}\}$, on a encore une propriété de Fubini, i.e. $\lim_n(\lim_m w_{nm})$ existe et est égal à w_{nm} .

Ce faisceau a été introduit par Manna et Shamir pour généraliser le théorème du plus petit point fixe de Kleene de manière à atteindre d'autres points fixes d'une définition récursive).

On peut remarquer ici que plus généralement, si X est un espace <u>totalement discontinu</u> (en anglais : extremally disconnected) , i.e. $\mathcal{U} \subseteq X$ ouvert $\Rightarrow \bar{\mathcal{U}}$ ouvert, alors une suite $\{x_n\}_{n \in \mathbb{N}}$ converge ssi elle est stationnaire. □

(vi) <u>notion de convergence d'ensembles au sens de Kuratow ski-Painlevé (ou KP-convergence)</u> :

(cette notion dûe à Painlevé [PAI] , est étudiée par Kuratow.ski [KUR] pp. 335-344, et utilisée par Arnold et Nivat 1978.

Soit X un T_2-espace topologique, $\{A_i\}_{i \in \mathbb{N}}$ une suite de parties de X (i.e. une suite de $\mathcal{P}(X)$). On définit $\underset{i}{LI}(A_i)$ et $\underset{i}{LS}(A_i)$ ainsi

1. <u>limite inférieure</u> : $\underset{i}{LI}(A_i)$

$\forall x \in X \quad x \in \underset{i}{LI}(A_i) \Longleftrightarrow \forall\, v(x)$ voisinage ouvert de x coupe tous les A_i sauf un nombre fini d'entre eux.

2. <u>limite supérieure</u> : $\underset{i}{LS}(A_i)$

$\forall x \in X \quad x \in \underset{i}{LS}(A_i) \Longleftrightarrow \forall v(x)$ voisinage ouvert de x définit une fonction monotone croissante partout définie sur $\mathbb{N}$

$$f_{v(x)} : n \longmapsto f(n) \quad \text{tq} \quad v(x) \cap A_{f(n)} \neq \emptyset$$

(On peut noter tout de suite que $\underset{i}{LI}(A_i) \subseteq \underset{i}{LS}(A_i)$)

3. <u>limite</u> : si $\underset{i}{LI}(A_i) = \underset{i}{LS}(A_i)$ alors on pose limite de A_i =

$$\lim_i A_i = \underset{i}{LI}(A_i) = \underset{i}{LS}(A_i) \quad .$$

La question qui se pose est de savoir si la fonction (partielle) $\lim : \{A_i\} \mapsto \lim_i A_i$ possède la propriété de Fubini.

<u>LEMME</u> : Soit $(A_{ij})_{i \in \mathbb{N}, j \in \mathbb{N}}$, alors

(i) la limite inférieure LI possède la propriété de Fubini forte i.e.

$$\underset{i}{LI}(\underset{j}{LI}(A_{ij})) = \underset{ij}{LI}(A_{ij})$$

(ii) $$\underset{i}{LS}(\underset{j}{LS}(A_{ij})) \subseteq \underset{ij}{LS}(A_{ij})$$

(iii) la limite $\lim$ au sens de Kuratowski-Painlevé vérifie la propriété de Fubini faible suivante :

si $\lim_i(\lim_j A_{ij})$ et $\lim_{ij} A_{ij}$ existent alors

on a l'égalité $\lim_i(\lim_j A_{ij}) = \lim_{ij} A_{ij}$

(iv) si la famille (A_{ij}) est ascendante (resp. descendante) le long des deux indices i,j alors on a la propriété de Fubini forte suivante

$$\lim_i (\lim_j A_{ij}) = \lim_{ij} A_{ij} . \qquad \square$$

<u>DEMONSTRATION</u> : (i) $x \in \underset{i}{LI}(\underset{j}{LI} A_{ij}) \Longleftrightarrow$ tout voisinage ouvert $v(x)$ de x coupe tous les $\underset{j}{LI} A_{ij}$ sauf un nombre fini d'entre eux (soit i_o), i.e.

$$\forall i > i_o \qquad \exists y \in v(x) \cap \underset{j}{LI} A_{ij}$$

$y \in \underset{j}{LI} A_{ij} \Longleftrightarrow$ tout voisinage ouvert de y coupe tous les A_{ij} sauf un nombre fini d'entre eux, soit j_o. $v(x)$ est un tel voisin age si on prend $v(x)$ ouvert $\Rightarrow v(x)$ coupe tous les A_{ij} sauf un nombre fini d'ente eux $\Longrightarrow \forall i > i_o \; \forall_j > j_o \quad v(x)$ coupe tous les A_{ij} i.e.

$$x \in \underset{ij}{LI} \; A_{ij} \;,\quad \text{d'où} \quad \underset{i}{LI}(\underset{j}{LI} A_{ij}) \subseteq \underset{ij}{LI} A_{ij}$$

Réciproquement :

$x \in \underset{j}{LI} \; A_{ij} \Longleftrightarrow$ tout voisinage ouvert de x coupe tous les A_{ij} sauf un nombre fini $\Longrightarrow$ pour i fixé, $v(x)$ coupe tous les A_{ij} sauf un nombre fini $\Rightarrow x \in \underset{j}{LI}(A_{ij})$ et $v(x)$ coupe tous $\underset{j}{LI}(A_{ij})$ $(x \in \underset{j}{LI} A_{ij}) \Rightarrow x \in \underset{i}{LI}(\underset{j}{LI} A_{ij})$ donc $\underset{ij}{LI} A_{ij} \subseteq \underset{i}{LI}(\underset{j}{LI} A_{ij})$

$$\text{donc} \quad \underset{i}{LI}\,(\underset{j}{LI} A_{ij}) = \underset{ij}{LI} A_{ij}$$

(ii) $x \in \underset{i}{LS}(\underset{j}{LS} A_{ij}) \Longleftrightarrow \forall v(x)$ voisinage ouvert définit $f_{v(x)} : n \mapsto f(n)$ tel que

$$(\underset{j}{LS} A_{f(n)j}) \cap v(x) \neq \emptyset$$

i.e. $\quad \exists y \in (\underset{j}{LS} A_{f(n)j}) \cap v(x)$

$y \in \underset{j}{LS} A_{f(n)j} \Longleftrightarrow \forall w(y)$ voisinage ouvert de y définit $g_{w(y)} : m \mapsto g(m) \quad$ tq $\; A_{f(n)g(m)} \cap w(y) \neq \emptyset$

mais si $v(x)$ ouvert, $v(x)$ est un $w(y) \Rightarrow v(x)$ définit

$\langle f,g\rangle : (n,m) \mapsto \langle f(n),g(m)\rangle$ tq $A_{\langle f,g\rangle (m,n)} \cap v(x) \neq \emptyset$

donc $x \in \underset{ij}{LS}\, A_{ij}$ i.e. $\underset{i}{LS}(\underset{j}{LS}\, A_{ij}) \subseteq \underset{ij}{LS}\, A_{ij}$

Pour démontrer la réciproque il faudrait pouvoir passer de

$$\forall i \quad \forall j \quad \exists i' \quad \exists j' \quad v(x) \cap A_{i'j'} \neq \emptyset \quad \text{à}$$

$$\forall i \quad \exists i' \quad \forall j \quad \exists j' \quad v(x) \cap A_{i'j'} \neq \emptyset$$

ce qui n'est pas possible en général.

(iii) si $\lim\limits_{i} \lim\limits_{j} A_{ij}$ existe et si $\lim\limits_{ij} A_{ij}$ existe aussi alors on a

$$\lim_{i} \lim_{j} A_{ij} = \underset{i}{LI}(\underset{j}{LI}\, A_{ij}) = \text{(d'après (i))}$$

$$\underset{ij}{LI}\, A_{ij} = \text{(hypothèse)} \lim_{ij} A_{ij} = \underset{ij}{LS}\, A_{ij}$$

(iv) on a $x \in \underset{ij}{LS}\, A_{ij} \Longleftrightarrow \forall\, v(x)$ ouvert

$$\forall i \quad \forall j \quad \exists i',j' \quad v(x) \cap A_{i'j'} \neq \emptyset .$$

I. Soit A_{ij} croissante le long des deux indices. Il s'ensuit

$$\forall i_1 \geqslant i' \qquad \forall j_1 \geqslant j' \qquad v(x) \cap A_{i_1 j_1} \neq \emptyset$$

$$x \in \underset{ij}{LS}\, A_{ij} \Rightarrow x \in \underset{ij}{LI}\, A_{ij} \qquad \text{d'où}$$

$$\underset{ij}{LS}\, A_{ij} \subseteq \underset{ij}{LI}\, A_{ij} \Longrightarrow \underset{ij}{LS} = \underset{ij}{LI}\, A_{ij}$$

Donc si A_{ij} est une famille d'ensembles $\subseteq X$ croissante le long des deux indices, alors elle possède une limite au sens de Kuratowski-Painlevé. D'où d'après (iii), $\lim\limits_{i} \lim\limits_{j} A_{ij} = \lim\limits_{ij} A_{ij}$

2. Soit A_{ij} décroissante le long des deux indices :

$$x \in \underset{ij}{LS}\ A_{ij} \Leftrightarrow \forall i\ \ \forall j\ \ \exists i' \geq i\ \ \exists j' \geq j\ \ v(x) \cap A_{i'j'} \neq \emptyset$$

or $v(x) \cap A_{i'j'} \subseteq v(x) \cap A_{ij}$

$$\therefore \forall i\ \ \forall j\ \ v(x) \cap A_{ij} \neq \emptyset \Rightarrow x \in \underset{ij}{LI}\ A_{ij}$$

$$\Rightarrow \underset{ij}{LS}\ A_{ij} \subseteq \underset{ij}{LI}\ A_{ij} \Rightarrow \lim_{ij} A_{ij} \text{ existe.}$$

On applique encore (iii). q.e.d. □

On voit donc que si $E = 2^X$ = l'ensemble des fermés de l'espace topologique X, on a la structure de <u>ω-faisceau</u>(*) suivante sur E :

$$\lambda : \mathcal{S}(E) \to E$$

$$\{A_n\}_{n \in \mathbb{N}} \mapsto \lim_n A_n \quad \text{si} \quad \underset{n}{LI}\ A_n = \underset{n}{LS}\ A_n\text{; non défini sinon}$$

$$\beta : E \to \mathcal{P}(\mathcal{S}(E))$$

$$u \mapsto \{\{A_n\}_{n \in \mathbb{N}} : A_n \text{ est KP-convergente et } u = \lim_n A_n\}$$

$\mathcal{S}(E)$ désigne l'ensemble des suites sur E. Ce faisceau sera appelé KP, ou <u>faisceau de la KP-convergence</u>.

(vii) <u>faisceau associé au λ-calcul</u> : (<u>faisceau LC</u>)

Si Λ est le langage du λ-calcul considéré [CUR], [HIN], on sait que la β-réduction est définie par :

$$(\lambda x.M)N \triangleright_\beta [N/x]\ M$$

On considère ici que la α-conversion est une égalité.
On peut considérer alors l'ensemble $\mathcal{S}(\Lambda)$ de toutes les chaînes (suites) de β-réduction, certaines d'entre elles aboutissant à des formes normales.
Si on ne rajoute aucun point à l'infini à Λ, on a alors le <u>ω-faisceau</u> LC(+) :

(*) avec seulement la propriété de Fubini faible.

(+) Nous nous excusons auprès du lecteur de la possible confusion entre le symbole de <u>limite</u> (noté ici lim), et le symbole de λ-abstraction.

$\ell im : F(\Lambda) \to \Lambda$

$\{s_n\}_{n\in N} \to \lim\limits_n s_n$: forme normale à la quelle aboutit $\{s_n\}_{n\in \mathbb{N}}$, si elle existe

$\beta : \Lambda \to P(F(\Lambda))$

$t \to \{s \in F(\Lambda)$: la chaîne s aboutit à $t\}$

On note ici que la limite s'appelle <u>forme normale</u>. Le théorème de <u>Church-Rosser</u> du λ-calcul ne fait rien d'autre qu'exprimer une <u>propriété</u> de Fubini très forte du faisceau LC .

(viii) faisceau associé à la notion d'approximation directe dans le λ-calcul de Wadsworth [WAD] et Lévy [LEV] : (faisceau WL)

On considère l'espace X des $\omega\beta$-formes normales [LEV] , pp 148, complété par idéaux (i.e. en prenant (i) $\Omega \leq$ tout le reste (ii) le prolongent $\subseteq$ de $\leq$ par monotonie (iii) rajoutant les points à l'infini des chaines ascendantes), d'une façon analogue à ce qu'on fait dans le magma M(F,V) . X est alors un poset et possède une structure de faisceau S (associé à la topologie de Scott, cf exemple (i)). Ce faisceau $< \lim_1, \beta_1 >$ sur X "prolonge" le faisceau $< \lim, \beta >$ donné à l'exemple (vii) sur l'espace $\Lambda \subseteq X$, en un certain sens. On l'appellera faisceau WL.

Remarque : On peut noter ici que, du point de vue informatique, le champ de la notion de faisceau ne se limite pas seulement à ce qui est appelé d'habitude sémantique "dénotationnelle", "algèbrique", théorie des algorithmes (cf. exemples plus loin et Prolégomènes). En effet la notion de faisceau englobe aussi la sémantique opérationnelle des programmes (cf. les travaux du laboratoire I.B.M. de Vienne, et par exemple Naït Abdallah 1976 pour une bibliographie). Si X est l'espace des formes normales et de leurs approximations (les ω-formes normales de Lévy [LEV] par exemple), $x \in X$ et $\sigma \in \beta(x)$, l'application σ décrit alors un calcul de x, le résultat de ce calcul étant donné par $\lambda(\sigma)$. Ce que nous appelons ici limite correspond alors à la forme normale du λ-calcul et des autres systèmes de réécriture (λ^*-calcul [NAI 76] , schémas de programmes [NIV] , etc...).

Dans la définition des faisceaux les systèmes approximants $\sigma \in \beta(x)$ sont des familles, i.e. des fonctions $\sigma : j \mapsto x_j = \sigma(j)$, $j \in J$, alors que les approximations sont faites au moyen de parties dirigées (chaines ascendantes) dans la théorie de Scott, et au moyen d'ensembles dans la théorie de Nolin. On peut considérer qu'un ensemble est une classe d'équivalence de familles

$$\sigma : j \mapsto x_j \sim \sigma' : i \mapsto x_i \iff \mathrm{Im}(\sigma) = \mathrm{Im}(\sigma') \iff \{x_j : j \in J\} = \{x_i : i \in I\}$$

ou encore que c'est une famille identité, c'est-à-dire un ensemble indexé par lui-même.

Dans l'approche de Scott comme dans celle de Nolin les limites sont prises par bornes (supérieures ou inférieures) dans un espace ordonné (cf. Prolégomènes et exemples plus haut). Ceci est un cas où la fonction limite considérée est assez grossière, car elle ne prend pas en considération toute l'information contenue dans un système approximant σ (en particulier les différentes étapes de ce système d'approximation) mais seulement l'information contenue dans l'image $Im(\sigma)$ de σ, et $Im(\sigma)$ est un ensemble. Ainsi si la limite est prise par borne supérieure (ou inférieure) d'une partie d'un ensemble ordonné, i.e. en prenant $\sqcup Im(\sigma)$ ou $\sqcap Im(\sigma)$, la structure particulière de σ importe peu(*) . Dans ce cas il revient au même de définir un faisceau sur un ensemble partiellement ordonné comme étant un couple $<\lambda, \beta>$ où :

$$\lambda : \mathcal{P}(X) \longmapsto X \qquad \lambda \in \{\sqcup, \sqcap\}$$

$$\beta : X \longmapsto \mathcal{P}(\mathcal{P}(X))$$

i.e. de ne regarder les systèmes d'approximation σ qu'à travers leurs images $Im(\sigma) \subseteq X$. Ceci est une notion restreinte de faisceau (cf. infra).

Cette dualité famille/ensemble nous paraît un point important pour bien comprendre l'approche de divers auteurs. Elle se retrouve entre la définition donné par Plotkin et Smyth des powerdomains (présentés comme des ensembles de parties), et qui est une tentative de faire de la sémantique du parallélisme un cas particulier de la théorie des cpo's, aboutissant à des difficultés que nous avons mentionnées dans les Prolégomènes, et l'approche à ces mêmes powerdomains, que nous développons plus loin, faisant ainsi le lien avec la théorie métrique d'Arnold et Nivat. (cf. Chap. III pp. 65-69). En dehors de leur nouvelle façon de construire des limites, l'élément essentiel de l'approche de ces deux auteurs est la restitution aux systèmes d'approximation σ, qu'ils appellent calculs (infinis), de leur statut de fonction. La limite d'un système approximant est

---

(*) à condition de ne pas raffiner la dichotomie convergence / non convergence.

est appelée par eux "valeur d'un calcul".

Fait et définition : produit de faisceaux

Soit $<\lambda,\beta>$ un faisceau sur X, et $<\lambda',\beta'>$ un faisceau sur Y, alors on définit sur le produit cartésien $X \times Y$ les deux fonctions

$$\lambda^* : \mathcal{S}(X \times Y) \to X \times Y$$

$$\lambda^*(S_1 \times S_2) = <\lambda(S_1), \lambda'(S_2)>$$

$$\lambda^*(\{x\} \times S_2) = <x, \lambda'(S_2)>$$

$$\lambda^*(S_1 \times \{y\}) = <\lambda(S_1), y>$$

chaque fois que c'est défini

$$\beta(<x,y>) = \{\sigma_1 \times \sigma_2 : \sigma_1 \in \beta(x) \cup \{x\},\ \sigma_2 \in \beta'(y) \cup \{y\}\} \setminus \alpha(x,y)$$

où $\alpha(x,y) = \emptyset$ si $<x,y> \in s(x) \times s(y)$ ($s(x), s(y)$ étant des spectres de x et y)

$= \{<x,y>\}$ sinon

Le couple $<\lambda^*, \beta^*>$ est un faisceau sur $X \times Y$ et est appelé produit des deux faisceaux $<\lambda,\beta>$, $<\lambda',\beta'>$. La construction se généralise à un nombre quelconque de faisceaux. □

Notons encore une propriété de la classe F de tous les faisceaux.

Fait et définition : coproduit de faisceaux

Soit $<\lambda,\beta>$ un faisceau sur X, $<\lambda',\beta'>$ un faisceau sur Y, alors on définit sur le coproduit (i.e. la réunion disjointe) d'ensembles $X \amalg Y$ les fonctions

(i) $\lambda^* = \lambda \amalg \lambda'$ (i.e. λ^* est l'unique prolongement de λ et λ' à $\mathcal{S}(X \amalg Y)$ tel que $\mathrm{Dom}(\lambda^*) = \mathrm{Dom}(\lambda) \amalg \mathrm{Dom}(\lambda')$.

(ii) $\beta^* = \beta \amalg \beta'$ (i.e. β^* est l'unique prolongement de β et β' à $X \amalg Y$).

Le couple $\langle \lambda^*, \beta^* \rangle$ est une structure de faisceau sur $X \amalg Y$, appelé <u>coproduit</u> (ou <u>somme</u>) <u>des faisceaux</u> $\langle \lambda, \beta \rangle$ et $\langle \lambda', \beta' \rangle$. La construction se généralise à un nombre quelconque de faisceaux. □

Pour des raisons de tradition nous noterons parfois $X+Y$ le coproduit $X \amalg Y$. (Notons que dans le cas des cpos cette somme <u>n'est pas</u> la somme traditionnelle; ceci tient au rôle spécial du $\perp$ dans ces structures).

<u>Notion de sous-faisceau</u>; <u>définition</u> :

Soit $\langle \lambda, \beta \rangle$ un faisceau sur X, $\langle \lambda', \beta \rangle$ un faisceau sur Y, avec $X \subseteq Y$. On dira que X est un <u>sous-faisceau</u> de Y ssi

(i) $\forall x \in X \qquad \sigma \in \beta(x) \Rightarrow \sigma \in \beta'(x)$

(ii) $\forall \sigma \in \mathfrak{F}(X) \subseteq \mathfrak{F}(Y) \qquad \lambda(\sigma) = \lambda'(\sigma)$ i.e. la fonction partielle λ' prolonge la fonction partielle λ. □

<u>Notion de faisceau quotient</u>; <u>définition</u> :

Soit X un ensemble muni d'un faisceau $\langle \lambda, \beta \rangle$. Soit $\sim$ une relation d'équivalence sur l'ensemble X. Cette relation induit une relation sur l'ensemble $\mathfrak{F}(X)$ des familles de X par

$$\forall \sigma, \sigma' \in \mathfrak{F}(X) \qquad \sigma \sim \sigma' \iff \mathrm{Dom}(\sigma) = \mathrm{Dom}(\sigma') \quad \text{et}$$

$$\forall_j \in \mathrm{Dom}(\sigma) \quad \sigma(j) \sim \sigma'(j)$$

On dira que la relation $\sim$ est <u>régulière</u> ssi elle respecte les limites i.e. :

$$\sigma \sim \sigma' \quad \Rightarrow \lambda(\sigma) \sim \lambda(\sigma')$$

<u>Fait</u> : Toute relation d'équivalence $\sim$ régulière sur X définit une structure de faisceau sur $X/\sim$ appelée structure de faisceau quotient. □

Soit $X^* = X/\sim$ l'espace des $\sim$-classes d'équivalence de X.

Notons $x \sim u \iff x \in [u]$ (la $\sim$ classe de u). On a

$$\mathfrak{F}(X^*) = \mathfrak{F}(X)/_{\sim} \quad \text{en posant}$$

$$\forall \sigma \in \mathfrak{F}(X) \quad [\sigma] = \lambda j \in \mathrm{Dom}(\sigma).\ [\sigma(j)]$$

(où $\lambda x \in A.f(x)$ dénote la fonction $x \mapsto f(x)$).
On peut définir sur X^*, $\forall [\sigma] \in \mathfrak{F}(X^*)$:

$$\begin{cases} \lambda^*([\sigma]) = [\lambda(\sigma_1)] & \text{où} \quad \sigma_1 \in [\sigma] \\ [\sigma] \in \beta^*([x]) \iff [\sigma] = [\tau] & \text{où } \tau \in \beta(u),\ u \in [x] \end{cases}$$

$\lambda^* : \mathfrak{F}(X^*) \to X^*$ est bien définie à cause de la régularité de $\sim$.
Vérifions la reconstruction des points :

$$\forall [\sigma] \in \beta^*([x]) \quad \text{on a}$$

$$\lambda^*([\sigma]) = [\lambda(\tau)] = (\text{si } \tau \in \beta(u),\ u \in [x]) = [u] = [x] = [\lambda(\sigma)]$$

Vérifions la propriété de Fubini :

$$\lambda^*_{i \in I}(\lambda^*_{j \in J}[w]_{i,j}) = (\sim \text{ sur } \mathfrak{F}(X^*)) = \lambda^*_{i \in I}(\lambda^*_{j \in J}[w_{i,j}]) =$$

(ce qui précède)

$$\lambda^*_{i \in I}([\lambda_{j \in J}(w_{ij})]) = [\lambda_{i \in I}(\lambda_{j \in J}\, w_{ij})] =$$

$$[\lambda_{(i,j) \in I \times J}\, w_{ij}] = \lambda^*_{i,j}[w_{ij}] = \lambda^*_{i,j}[w]_{i,j}$$

Le couple $\langle \lambda^*, \beta^* \rangle$ définit donc bien un faisceau sur $X^* = X/_{\sim}$
Nous appellerons ce faisceau le <u>faisceau quotient</u> et on pourra le noter

$$\langle \lambda^*, \beta^* \rangle = \langle \lambda, \beta \rangle/_{\sim}$$

□

Nous aurons besoin des notions de sous-faisceau et de faisceau quotient dans l'étude des catégories de faisceaux Fais, Foen, Dalg, BF, BFc, etc... (cf. chapitre IV) . Dans le cas ordonné monique la notion de faisceau quotient doit recevoir une forme légèrement différente (cf. Chapitre IV, démonstration de 4.3).

II - CLASSES DE FAISCEAUX :

On voit que la notion de faisceau est très générale. Pour une étude effective et les applications à l'informatique, on doit la particulariser de diverses manières. Posons quelques définitions.

Eléments rationnels : définition : Soit $<\lambda,\beta>$ un faisceau sur un ensemble X, un élément $u \in X$ est dit rationnel ssi $\forall \sigma \in \beta(u)$ $u \in \mathrm{Im}(\sigma)$ i.e. u appartient à chaque famille qui l'approxime.

Exemple : dans une structure de type sur un domaine $\hat{D}$, toute fermeture $\bar{d} = \downarrow d$ d'un point $d \in D$ est un élément rationnel du faisceau $<\lambda,\beta>$ de l'exemple (iii) ci-dessus (faisceau SW).

Faisceaux élémentaires : définition : Un faisceau $<\lambda,\beta>$ sur X est élémentaire ssi son noyau $N_{\lambda,\beta}(X) = \cup \{\mathrm{Im}(\sigma) : \sigma \in \beta(x),\ x \in X\}$ ne contient que des éléments rationnels. □

C'est le cas des exemples (ii) (collections d'algorithmes), (iii) (domaine de types de Shamir et Wadge), (v) ci-dessus.

Faisceaux moniques : définition : Un faisceau $<\lambda,\beta>$ est monique ssi la fibre de chaque élément est réduite aux systèmes approximants spectraux de cet élément, i.e.

$$\forall x \in X \qquad \forall \sigma_1, \sigma_2 \in \beta(x) \qquad \mathrm{Im}(\sigma_1) = \mathrm{Im}(\sigma_2)$$

et

$$\forall x,y \qquad x \neq y \Rightarrow \forall \sigma \in \beta(x) \quad \forall \sigma' \in \beta(y)$$

on a $$\mathrm{Im}(\sigma) \neq \mathrm{Im}(\sigma') \ .$$ □

Remarque importante :

On peut noter que chaque faisceau monique définit de façon unique un couple de fonctions s, λ^* $\quad X \overset{\lambda^*}{\underset{s}{\leftrightarrows}} \mathcal{P}(X)$, $x \mapsto s(x)$ appelée <u>fonction spectre</u> et $\lambda^* : S \mapsto \lambda(S) = \lambda(\mathrm{Im}(\sigma))$ si $S = \mathrm{Im}(\sigma)$, $\sigma \in \beta(x)$ tel que $\forall x \in X \quad x = \lambda(s(x))$ i.e. $\lambda \circ s = id_X$, $s(x)$ étant la valeur commune des $\mathrm{Im}(\sigma)$, $\sigma \in \beta(x)$. □

Les définitions des faisceaux que nous avons données ne font pas référence à la structure interne des espaces, i.e. aux propriétés des éléments (c'est ce qui nous avait permis de faire du faisceau utilisé par Manna et Shamir 1977 un cas particulier de celui utilisé par Arnold et Nivat 1978). Elles s'appliquent donc dans d'autres circonstances. Nous allons considérer deux cas (non disjoints) : les espaces ordonnés et les espaces topologiques.

<u>Faisceaux ordonnés</u>; <u>définition</u> :

Un faisceau $\langle \lambda, \beta \rangle$ sur X est <u>ordonné</u> ssi

(i) X a une structure d'espace partiellement ordonné et λ se factorise $\lambda = \Psi_\lambda \circ \mathrm{Im}$ où $\Psi_\lambda \in \{\sqcup, \sqcap\}$.

(ii) $\mathcal{S}(X)$ est ordonné par inclusion si $\Psi_\lambda = \sqcup$, et inclusion inverse si $\Psi_\lambda = \sqcap$.

<u>Faisceaux ordonnés moniques</u> : <u>définition</u> :

Un faisceau $\langle \lambda, \beta \rangle$ sur un ensemble X est ordonné monique (en abrégé fom) ssi :

(i) c'est un faisceau ordonné

(ii) il est monique, i.e. définit un $X \overset{\Psi_\lambda}{\underset{s}{\leftrightarrows}} \mathcal{P}(X)$

(iii) sa fonction spectre vérifie :

$$x \sqsubseteq y \Rightarrow \begin{cases} s(x) \subseteq s(y) & \text{si } \Psi_\lambda = \sqcup \\ s(y) \subseteq s(x) & \text{si } \Psi_\lambda = \sqcap . \end{cases}$$ □

La classe des fom est close par produit et coproduit.

En fait de nombreux faisceaux ordonnés moniques définissent des connections de Galois particulières.

Plus précisément :

<u>Définition</u> : <u>Connections de Galois</u> ([MCL] pp. 93-94)

Soient S et T deux ensembles partiellement ordonnés.

Un couple d'applications $S \underset{g}{\overset{d}{\leftrightarrows}} T$ est une connection de Galois (ou une adjonction) entre les deux posets S et T ssi :

(i) d et g sont monotones

(ii) $g(x) \supseteq y \iff x \supseteq d(y) \quad \forall x \in S,\ y \in T$.

g est appelé l'<u>adjointe supérieure</u> et d <u>l'adjointe inférieure</u>.

Dans le cas présent soit $\langle \lambda, \beta \rangle$ un faisceau ordonné monique sur X, $N(X)$ son noyau et $T = \mathfrak{P}(N(X))$ l'ensemble des parties du noyau. Supposons pour simplifier les énoncés que $\lambda \circ \mathrm{Im} = \sqcup$.

Alors :

<u>1.1. PROPOSITION</u> : (i) Tout faisceau ordonné monique dont les spectres sont des sections commençantes (lowersets) du noyau définit une connection de Galois

$$X \underset{s}{\overset{\lambda^*}{\leftrightarrows}} T = \mathfrak{P}(N(X)) \subseteq \mathfrak{P}(X)$$

(ii) L'ensemble des faisceaux ordonnés moniques tels que $\forall x \in X \quad s(x) = \downarrow_{N(x)} s(x)$, quotienté par la relation d'équivalence $\langle \lambda, \beta \rangle \sim \langle \lambda', \beta' \rangle \iff \forall x \quad s(x) = s'(x)$, a pour classes d'équivalence toutes les connections de Galois $X \underset{s}{\overset{\lambda^*}{\leftrightarrows}} T \subseteq \mathfrak{P}(N(X))$ telles que $\lambda^* \circ s = id_X$.

<u>DEMONSTRATION</u> : (i) s et λ^* sont monotones par définition, et $\lambda^*(s(x)) = x$ (faisceau) i.e. $\lambda(y) \subseteq x$

$y \subseteq s(x) \Rightarrow \lambda^*(y) \subseteq \lambda^*(s(x)) = x$

Réciproquement :

$\lambda^*(y) \sqsubseteq x \iff \sqcup y \sqsubseteq x$ i.e.

$\forall a \in y \quad a \sqsubseteq x$ et $a \in N(X) \Rightarrow$ (les spectres sont des sections du noyau) $\Rightarrow a \in s(x) \Rightarrow y \subseteq s(x)$.

Il s'ensuit que $X \underset{s}{\overset{\lambda^*}{\leftrightarrows}} T = \mathcal{P}(N(X))$ est une connection de Galois.

(ii) soit $X \underset{s}{\overset{\lambda^*}{\leftrightarrows}} T$ une connection de Galois telle que $\lambda \circ s = id_X$.

C'est une $\sim$-classe de faisceaux. $X \underset{s}{\overset{\lambda^*}{\leftrightarrows}} T \subseteq \mathcal{P}(X)$ donne un couple de flèches et

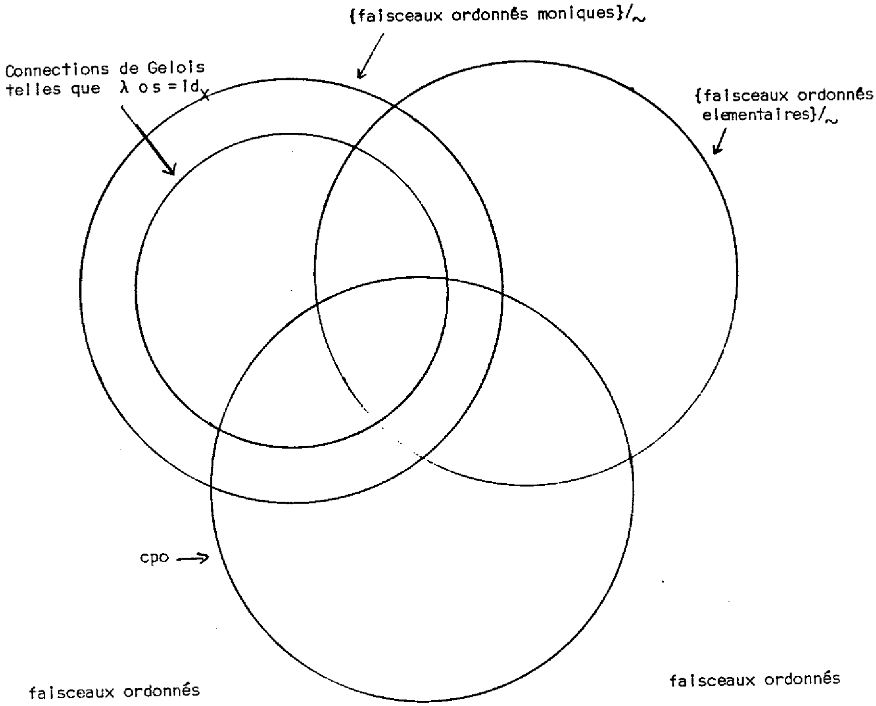
1. λ^* est surjective et s injective à cause de $\lambda^* \circ s = id_X$
2. la propriété de Fubini est trivialement vérifiée (cf. exemples §I).
3. $\forall x \in X \qquad x = id_X(x) = \lambda^*(s(x))$ par hypothèse on a donc la propriété de reconstruction des points. C'est donc une $\sim$-classe d'équivalence de faisceaux.
4. ils sont ordonnés moniques car X et T sont des posets et s est monotone

Enfin les spectres sont des sections commençantes du noyau i.e.

$$\forall x \in X \qquad s(x) = \{a \in N(X) : a \sqsubseteq x\}$$

en effet $\forall a \in N(X) \quad a \sqsubseteq x \Rightarrow \lambda(\{a\}) \sqsubseteq x \Rightarrow$ (connection de Galois) $\{a\} \subseteq s(x)$ i.e. $a \in s(x)$. □

On a un énoncé analogue au précédent si $\lambda = \sqcap$. On peut relever aussi que tous les fom élémentaires définissent des connections de Galois (par monotonie de s).



Ici les cpo's sont supposés munis soit du faisceau associé à la topologie de Scott, soit d'autres faisceaux (cf. Chapitre II, Proposition 2.8).

Diagramme sur les faisceaux ordonnés

(N.B. : pour l'intersection avec les cpo et les faisceaux ordonnés les plus généraux, on a supposé que les $\sim$-classes d'équivalence étaient représentées par un élément particulier).

Sur une notion restreinte de faisceau ordonné

Dans ce qui précède concernant les faisceaux ordonnés, on a pu remarquer un jeu constant entre les faisceaux eux-mêmes et leurs classes d'équivalence par la relation

$$< \lambda, \beta > , < \lambda', \beta' > \text{ donnés sur un même } X$$
$$< \lambda, \beta > \equiv < \lambda', \beta' > \iff (\Psi_\lambda = \Psi_{\lambda'}) \quad \text{et}$$

si on définit $\forall x \in X$

$$\mathrm{Im} \circ \beta : x \mapsto \{\mathrm{Im}(\sigma) : \sigma \in \beta(x)\}$$

alors $\mathrm{Im} \circ \beta = \mathrm{Im} \circ \beta'$.

Si on injecte $\mathcal{P}(X)$ dans $\mathfrak{F}(X)$ par :

$i : \mathcal{P}(X) \to \mathfrak{F}(X)$, $S \mapsto \lambda s \in S.\ s = i(S)$

(i.e. "les parties sont des familles indexées par elles-mêmes"), alors chaque $\equiv$-classe d'équivalence de faisceaux contient un unique faisceau de la forme :

$$\lambda_0 : \mathcal{P}(X) \to X$$
$$\beta_0 : X \to \mathcal{P}(\mathcal{P}(X))$$

Chaque fois que l'on considère un faisceau ordonné, si on ne raffine pas la dichotomie convergence/divergence il est souvent suffisant de le considérer à travers sa $\equiv$-classe d'équivalence.
Nous le ferons souvent par la suite, parfois sans mention expresse. Mais la notion de famille est quand même nécessaire. Nous la verrons réapparaître au Chapitre IV sous la forme de diagramme.

<u>Remarques bibliographiques sur les faisceaux ordonnés au sens restreint</u> :

La première référence à la notion de faisceau ordonné [(1)] que nous connaissons est celle de G.N. Raney, 1952 [RAN 1] . L'auteur définit : si L est un treillis complet, un élément $x \in L$ est complètement $\sqcup$-irréductible ssi

$$\forall S \subseteq L \qquad x \leq \sqcup S \qquad \Rightarrow \exists y \in S \qquad x \leq y$$

(ce qui est une façon de dire que x est fini, ou compact, pour l'ensemble $\mathcal{P}(L)$ des parties de L). Il énonce alors : un treillis complet est isomorphe à un anneau complet d'ensembles si et seulement chacun de ses éléments est borne supérieure de quelque ensemble d'éléments $\sqcup$-irréductibles. (C'est donc un faisceau).

D'autres références sont Raney 1953 [RAN 2] S. Papert 1958 [PAP] , et G. Bruns 1961 et 1962 [BRU] , s'occupant toutes de la théorie des <u>treillis complètement distributifs</u>. Dans tous ces travaux, la notion de faisceau ordonné est utilisée en tant qu'instrument mais non étudiée en tant que telle.

Dans [ADJ] Wright, Wagner et Thatcher, généralisant des travaux de Guessarian 1975, Courcelle et Nivat 1976 , Markovsky et Rosen 1976,..., définissent et étudient plusieurs notions voisines de celles développées plus haut. Dans [SCS]K.H. Hofmann définit une notion <u>d'ordre approximant</u> qui correspond aux faisceaux moniques ordonnés dans le cas où les spectres sont des idéaux et l'espace un treillis complet, et s'en sert pour étudier les treillis continus.

Les espaces considérés par ADJ sont des posets P. Un <u>subset-system</u> est une application $Z : P \mapsto Z(P) \subseteq \mathcal{P}(P)$ telle que

(i) $\exists P$ tq $Z(P) \neq \{\emptyset\}$

(ii) si $f : P \to P'$ est monotone alors

$$\forall S \in Z(P) \qquad f(S) \in Z(P')$$

Les éléments de $Z(P)$ sont dits <u>Z-ensembles</u>.

L'objet qui, première vue, se rapproche le plus des subset-system, dans le cadre des faisceaux est une application $\tilde{\beta} : X \mapsto \cup \{ \beta(x) : x \in X \}$ qui à

[(1)] au sens restreint de la remarque précédente.

chaque espace X muni d'un faisceau ordonné, $\lambda = \sqcup$, fait correspondre la réunion de toutes les fibres de ses éléments. On suppose en plus que β est injective et totale, ce qui est une condition plus forte.

Un poset P est <u>Z-complet</u> ssi

$$\forall \sigma \in Z(P) \qquad \sqcup \sigma \in P$$

i.e. tous les systèmes approximants (où Z-ensembles convergent). Dans notre construction nous imposons

$$\forall x \in X \qquad \forall \sigma \in \beta(x) \quad x = \lambda(\sigma)$$

i.e. les Z-ensembles ont des limites, on sait les classer $\tilde{\beta}(X) = \sqcup\{\beta(x) : x \in X\}$ et s'en servir pour approximer tous les éléments de l'espace. C'est donc une condition plus forte. (On pourrait dire que du point de vue de l'informatique, un des avantages de l'approche de ADJ , en considérant des systèmes qui ne convergent pas, serait de pouvoir inclure au niveau de la construction la notion de calcul qui ne converge pas; mais cette notion reste mal investiguée).

Une fonction $f : P \to P'$ est <u>Z-continue</u> si et seulement si $\forall \sigma \in Z(P)$ tq $\sqcup\sigma$ existe , $f(\sqcup\sigma) = \sqcup f(\sigma)$.
Ici nous définissons les fonctions $f : P \to P'$ régulières de façon analogue En particulier toutes les fonctions régulières sont Z-continues pour $Z = \tilde{\beta}$

Un élément $p \in P$ est <u>Z-compact</u> ssi $\forall S \in Z(P)$ tp $\sqcup S$ existe $p \sqsubseteq \sqcup S \Rightarrow \exists s \in S \quad p \sqsubseteq s$. On peut voir que dans un faisceau ordonné $\langle \sqcup, \beta \rangle$, tout élément $\tilde{\beta}$-compact est rationnel, car :
p compact $\Leftrightarrow \forall S \in \tilde{\beta}(X) \quad p \sqsubseteq \sqcup S \Rightarrow \quad \exists s \in S \quad p \sqsubseteq s$
donc $p = \sqcup S \Rightarrow p \sqsubseteq \sqcup S \Rightarrow$ (compacité) $\exists s \quad p \sqsubseteq s$. Comme on a

$$s \sqsubseteq \sqcup S = p \Rightarrow s \sqsubseteq p \sqsubseteq s \Rightarrow p = s$$

il vient $\forall S \in \tilde{\beta}(X) \quad p = \sqcup S \Rightarrow \exists s \in S \quad p = s$ donc p est rationnel.

Donc dans un faisceau ordonné, la rationalité est une notion plus générale que la compacité. Mais dans un treillis complètement distributif les deux notions coïncident, i.e. tout rationnel est compact :

$$p \sqsubseteq \sqcup S \Rightarrow p = \sqcup \{s \sqcap p : s \in S\} \Rightarrow \text{(rationnalité)}$$

$$\exists s \in S \quad p = s \sqcap p \quad \text{i.e. } \exists s \in S \quad p \sqsubseteq s \quad .$$

Ceci peut être généralisé de la façon suivante :

Soit $\langle \lambda, \beta \rangle$ un faisceau ordonné (avec $\lambda = \sqcup$); on dira qu'il est <u>$\sqcap$-régulier</u> si et seulement si

$$\forall S \in \tilde{\beta}(X) \quad \forall x \in X \qquad x \sqcap (\sqcup S) = \sqcup \{x \sqcap s : s \in S\} .$$

Dans un faisceau $\sqcap$-régulier les éléments compacts et les éléments rationnels coïncident. On pourra voir plus loin que dire qu'un faisceau est $\sqcap$-régulier revient à dire que la fonction $\lambda y.\ x \sqcap y : y \rightarrow x \sqcap y$ est régulière. (On retrouvera cette fonction plus loin dans les <u>domaines commodes</u>).

Un cas très fort de $\sqcap$-régularité est évidemment donné par les treillis complets complètement distributifs. Un autre exemple de faisceau $\sqcap$-régulier est donné par les <u>treillis inter-continus</u> (meet-continuous) (cf. [SCS]).

Un poset P est <u>Z-inductif</u> ssi $\forall p \in P \quad \exists \sigma \subseteq$ Z-core $p = \sqcup \sigma$, ce qui correspond à notre notion de faisceau élémentaire.

Notons encore que si l'on définit

$\forall x \in P \qquad x \mapsto \{S \in Z(P) : x * \sqcup S\} = \beta(x)$, alors $\{x\} \in s(x)$ dans tous les cas en vertu de la propriété (ii) de la définition des subsets-systèm - ce qui est une des motivations, selon nous, du passage aux Z-compacts pour définir des rationnels [(1)]. Si P est un cpo algébrique, et x n'est pas compact, alors $s(x)$ ne donne pas uniquement les façons d'approximer x au moyen d'éléments finis (Ceci peut sembler gênant dans certains cas, et rejoint un problème soulevé plus haut où on a aussi vu que le fait de mettre un élément dans sa propre fibre, facilite parfois la démonstration de certains énoncés).
Mais cette contrainte très forte nous priverait par ailleurs des faisceaux que nous construirons plus loin sur les espaces de fonctions, et qui nous semble-t-il jouent un rôle important en informatique. (cf. Chapitre III).

De ce point de vue la notion de support $\sigma(x)$ d'un élément x définie par ADJ correspondrait mieux à notre notion de spectre, mais alors la relation entre les notions de régularité et celle de Z-continuité dans un poset Z-inductif reste à investiguer.

(1) En particulier Noyau (P) = P.

En résumé il nous semble que l'on peut dire en simplifiant que ADJ s'interesse à une théorie de la construction des faisceaux ordonnés moniques élémentaires, dans laquelle les définitions données ici ne sont qu'un résultat final. Donc en ce qui concerne cette classe de faisceaux, le travail présenté ici se situe en aval des recherches de ADJ. Notre approche est plus générale au sens suivant : elle englobe d'autres faisceaux, et débouche sur des solutions aux problèmes du passage à l'espace des fonctions, et des applications à la programmation.

D'autre part la notion de système approximant, en tant que fonction, autorise une analyse plus fine des modèles.

Faisceaux moniques topologiques : Définition

Un faisceau monique $X \underset{s}{\overset{\lambda}{\rightleftarrows}} \mathcal{P}(X)$ est un faisceau topologique ssi X et $\mathcal{P}(X)$ sont des espaces topologiques et s et λ sont continues. □

Un exemple de faisceau monique topologique est fourni par les constructions de Scott, ou encore par la construction de l'algèbre (magma) complétée $M^{\infty}(F,V)$. Plus précisèment, un résultat un peu plus général que le suivant sera démontré dans la suite de ce mémoire (chapitre II, Corollaire 2.15) :

1.7. PROPOSITION : Soit X un treillis complet et $\mathcal{P}(X)$ l'ensemble de ses parties ordonné par inclusion, munis de leurs topologies de Scott, et $\lambda = \sqcup$. Alors il existe un faisceau monique $X \underset{s}{\overset{\lambda}{\rightleftarrows}} \mathcal{P}(X)$ qui est un faisceau topologique ssi X est un treillis continu . □

III - FONCTIONS REGULIERES

DEFINITION : Fonction régulière sur un faisceau

Soit X un ensemble muni du faisceau $\langle \lambda, \beta \rangle$ et Y muni du faisceau $\langle \lambda', \beta' \rangle$. Une fonction $f : X \to Y$ est régulière au point $x \in X$ si et seulement si

$$\forall\ \sigma \in \beta(x) \qquad f(x) = \lambda'(f(\sigma))$$

i.e. ssi le diagramme suivant commute

$$\begin{tikzcd} x \arrow[r, "f"] \arrow[d, squiggly] & f(x) \arrow[d, "\lambda'"] \\ \beta(x) \ni \sigma \arrow[r, "\mathfrak{S} f"] & f(\sigma) \end{tikzcd}$$

(Cela implique en particulier que f est définie sur le spectre $s(x)$ de x).

Une fonction $f : X \to Y$ est régulière sur X ssi elle est régulière en chaque point, i.e. le diagramme suivant commute :

$$\begin{tikzcd} X \arrow[rr, "\beta"] \arrow[d, "f"'] & & \mathcal{P}(\mathfrak{S}(X)) \arrow[d, "\mathcal{P}(\mathfrak{S}(f))"] \\ Y \arrow[dr, "\{\ \}"'] & & \mathcal{P}(\mathfrak{S}(Y)) \arrow[dl, "\mathcal{P}\lim_Y"] \\ & \mathfrak{S}(Y) & \end{tikzcd}$$ □

Noter que dans cette définition seule la partie λ' du faisceau sur Y est utilisée.

Exemples de fonctions régulières sur des faisceaux :

(i) Si X et Y sont des cpo's comme ci-dessus, i.e. munis du faisceau associé à la topologie de Scott, on a $f : X \to Y$ régulière au pointe $x \in X$ ssi

$$\forall\ S \quad \text{dirigée} \quad \text{tq} \quad x = \sqcup S\ , \quad f(x) = \sqcup f(S)$$

i.e. ssi f est continue au point x au sens de la topologie de Scott (ou si l'on préfère, des cpo's).

(ii) Soient X, Y des modèles ensemblistes de collections d'algorithmes, munis du faisceau exhibé plus haut. Alors $f : X \to Y$ est régulière au sens des faisceaux ssi elle définit un <u>algorithme</u> au sens de Nolin.

<u>DEMONSTRATION</u> : 1. Tout algorithme définit une fonction régulière (Nolin 1974) :

Soit $x \in X$, $s(x) = \{y_i\}_{i \in I}$ $(\Rightarrow x = \bigcup_{i \in I} y_i)$

et A un algorithme de X dans Y, c'est-à-dire

$$A = \bigcap_{x \in X} F \times A[x]$$

Alors $\forall u \in Y \quad A[x] \subseteq u \iff A[\bigcup_i y_i] \subseteq u \iff$

$A \subseteq F\,(\bigcup_i y_i)\,u \iff$

$A \subseteq \{f : X \to Y \text{ normale} \mid f(\bigcup_i y_i) \subseteq u\}$

$\iff A \subseteq \{f : X \to Y \text{ normale} \mid \forall i \ \ f(y_i) \subseteq u\}$

$\iff A \subseteq \bigcap_i \{f : X \to Y \text{ normale} \mid f(y_i) \subseteq u\}$

$\iff A \subseteq F\,y_i\,u \quad \forall i \in I \iff$

$\iff A[y_i] \subseteq u \quad \forall i \in I \iff \bar{\bigcup_i} A[y_i] \subseteq u$

Donc $A[x] = \bar{\bigcup_i} A[y_i]$, c'est-à-dire la fonction $x \mapsto A[x]$ est une fonction régulière.

2. Réciproquement soit $f : X \to Y$ une fonction régulière; faisons-lui correspondre l'algorithme

$$a_f = \bigcap_{x \in X} F \times f(x)$$

Il est clair d'après le point 1. que $x \mapsto a_f[x]$ est entièrement déterminé par son graphe sur les atomiques. Il suffit donc de vérifier que $\forall t$ atomique $f(t) = a_f[t]$. Soit t atomique

$$a_f[t] = \bigcap\{z \in Y : a_f \subseteq F\,t\,z\} =$$

$$\bigcap\{z \in Y : \forall h\ (\forall x \in X \ \ h(x) \subseteq f(x)) \Rightarrow h(t) \subseteq z\} =$$

$$= \bigcap\{z \in Y : \forall h \quad h \subseteq f \Rightarrow h(t) \subseteq z\} =$$

$$= \bigcap\{z \in Y : h \text{ majorée par } f \Rightarrow h(t) \subseteq z\} =$$

$$= \bigcap\{z \in Y : f(t) \subseteq z\} = f(t) \quad .$$

Les fonctions f et $x \mapsto a_f[x]$ sont identiques. On peut donc identifier, au niveau de l'évaluation, les fonctions régulières de X dans Y et les algorithmes de X dans Y.

(iii) Soit $\hat{D}$ une structure de type au sens de Shamir et Wadge. Ces auteurs définissent

$$f : \hat{D} \to \hat{D} \quad \text{monotone croissante est "}\underline{\text{tight}}\text{" ssi}$$
$$\forall x \in \hat{D} \qquad f(x) = \bigsqcup_{d \in x} f(\bar{d})$$

Sachant que $\bar{d} = \downarrow d$, ceci n'est rien d'autre que la régularité de f pour le faisceau exhibé plus haut.

(iv) Si on reprend la notion de convergence de Manna et Shamir, on a ([MAS] , pp. 9)

$$f : A \to B \ \ \underline{\text{continue}} \ \ \text{ssi} \ \forall \ \text{suite convergente} \ (x_j) \ \text{on a}$$
$$f(\lim_j x_j) = \lim_j f(x_j)$$

ce qui encore une instantiation de la régularité.

<u>Notation</u> : Si X et Y sont deux faisceaux, on note

$$[X \to Y] = \{ f : X \to Y \mid f \ \text{ régulière sur } X \} .$$

<u>1.2.</u> <u>LEMME</u> : Soient X, Y deux faisceaux. Alors

(i) l'identité $\lambda x.x : x \mapsto x$ est régulière

(ii) toute fonction constante $\lambda x.c : x \mapsto c$ est régulière.

(iii) si $\{f_i\}_{i \in I}$ est une famille de fonction régulières au point $x \in X$ et telle que les limites $\lim_i f_i(x)$ et $\forall \sigma \in \beta(x)$ $\lim_\sigma(\lim_i f_i(\sigma))$ existent dans Y, alors la fonction partielle

$$y \mapsto \lim_i f_i(y)$$

est régulière au point x (notée $\lambda y. \lim_i f_i(y)$) .

<u>DEMONSTRATION</u> : (i) (ii) évidents

(iii) soit $\sigma \in \beta(x)$, alors

$(\lambda y.\ \lim_i f_i(y))(x) = \lim_i f_i(x) = \lim_i(\lim_\sigma f_i(\sigma)) =$

(Fubini) $\lim_{I\times\sigma} f_i(\sigma) =$ (Fubini) $\lim_\sigma (\lim_i f_i(\sigma)) =$

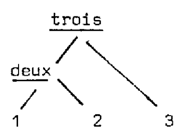
$\lim_\sigma (\lambda y.\ \lim_i f_i(y))(\sigma) = \lim(\lambda y.\ \lim_i f_i(y))\ (\sigma)$

d'où la régularité au point x. □

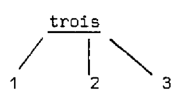
<u>Remarque sur l'identité</u> : En fait la notion de régularité de l'identité intervient dès la définition des faisceaux. Un couple de fonctions $\langle\lambda,\beta\rangle$, λ surjectif, β injectif vérifiant la propriété de Fubini est un faisceau ssi l'identité est régulière. □

Nous avons défini une classe d'objets, les faisceaux, et des flèches entre ces objets, les fonctions régulières. Avons-nous construit une catégorie ? La réponse est non, car la composition de deux fonctions régulières n'est généralement pas régulière si on prend la composition usuelle.

<u>Exemple</u> : Soit X le sup-demi treillis suivant



et Y le sup-demi treillis



On a $Y \subseteq X$. Munissons X du faisceau défini par $\lambda = \sqcup$

β : $x \mapsto \{\{x\}\}$ si $x = 1, 2, 3$

<u>deux</u> $\mapsto \{\{1,2\}\}$

<u>trois</u> $\mapsto \{\{1,2,3\}\}$

et Y du faisceau défini par la restriction de λ et β . Et soient les fonctions

$$g : Y \mapsto X \quad \text{définie par l'inclusion}$$
$$f : X \mapsto Y \quad \text{définie par } \underline{\text{deux}} \mapsto \underline{\text{trois}} \text{ et l'identité ailleurs}$$

On voit que g et f sont régulières, mais

$$(g \circ f)(\underline{\text{deux}}) = g(f(\underline{\text{deux}})) = g(f(1) \sqcup f(2)) = g(1 \sqcup 2)$$
$$= g(\underline{\text{trois}}) = \underline{\text{trois}} \neq (g \circ f)(1) \sqcup (g \circ f)(2) = 1 \sqcup 2 = \underline{\text{deux}}$$

La composée $g \circ f : x \mapsto g(f(x))$ n'est donc pas régulière au point deux. □

On n'a donc pas une catégorie si on prend la composition usuelle Mais on a quand même deux propriétés intéressantes dans la classe de tous les faisceaux (supposés bien sûr inclus dans quelques univers $\mathcal{U}$, cf. [MCL] pp. 22 ou [MAN] pp. 95-102).

1.3. LEMME : Régularité des projections

Soient X,Y deux faisceaux et $X \times Y$ leur produit, alors les deux projections $p_1 : (x,y) \mapsto x$ et $p_2 : (x,y) \mapsto y$ sont régulières

DEMONSTRATION : soit $p = p_1$ par exemple.
A-t-on $p(x,y) = x = \lim p(\sigma)$ où $\sigma \in \beta((x,y))$?
Supposons $\sigma = \sigma_1 \times \sigma_2$, $\sigma_1 \in \beta(x)$, $\sigma_2 \in \beta'(y)$, alors

$$p(\sigma) = p([\sigma_1,\sigma_2]) = \{ p(a,b) : (a,b) \in \sigma_1 \times \sigma_2 = \{a : a \in \sigma_1\} = \sigma_1$$

d'où $\lim p(\sigma) = \lim \sigma_1 = x$.

Si $\sigma_1 = \{x\}$ ou $\sigma_2 = \{y\}$, il est clair qu'on aboutit au même résultat. Donc p est régulière. qed. □

1.4. PROPOSITION : Soient X, Y, Z des faisceaux. Alors $f : X \times Y \to Z$ est régulière sur $X \times Y$ si et seulement si elle est régulière pour chacune de ses variables.

DEMONSTRATION : $\Rightarrow$:
montrons que $g_{x_0} : y \mapsto f(x_0, y)$ est régulière. Si $\sigma_2 \in \beta'(y)$ on a $g_{x_0}(y) =$ (faisceau) $= g_{x_0}(\lim \sigma_2) = f(x_0, \lim \sigma_2) =$ (définition du produit) $= f(\lim(\{x_0\}, \sigma_2)) =$ (régularité) $= \lim f(\{x_0\}, \sigma_2) = \lim f(x_0, \sigma_2) = \lim g_{x_0}(\sigma_2)$.

On fait la même chose pour $g_{y_0} : x \mapsto f(x, y_0)$.

$\Leftarrow$: soit $\sigma = \sigma_1 \times \sigma_2 \in \beta((x,y))$ où $\sigma_1 \in \beta(x)$, $\sigma_2 \in \beta'(y)$

$f(x,y) = f(\lim \sigma) = f(\lim \sigma_1, \lim \sigma_2) =$ (régularité par rapport à x) $= \lim f(\sigma_1, \lim \sigma_2) =$ (régularité par rapport à y) $=$

$\lim\limits_{\sigma_1} (\lim\limits_{\sigma_2} f(\sigma_1, \sigma_2)) =$ (Fubini) $\lim\limits_{\sigma_1 \sigma_2} f(\sigma_1, \sigma_2) = \lim\limits_{\sigma} f(\sigma) = \lim f(\sigma)$

Par ailleurs si $\sigma_1 = \{x\}$ par exemple, $\sigma_2 \in \beta'(y)$ $f(x,y) = f(\lim \sigma) = f(x, \lim \sigma_2) =$ (régularité par rapport à y) $\lim\limits_{\sigma_2} f(x, \sigma_2) = \lim\limits_{\{x\} \times \sigma_2} f(\{x\}, \sigma_2) = \lim\limits_{\sigma} f(\sigma) = \lim f(\sigma)$

Donc f est bien régulière. □

Remarque sur les produits de faisceaux et la régularité des fonctions "partielles" g_x et g_y :

D'un point de vue conceptuel on peut discuter le bien-fondé de la définition que nous avons donnée pour les produits de faisceaux. En effet, pour ce qui concerne l'informatique, pour approximer un couple de points tous deux à "l'infini" (x_∞, y_∞), cela veut dire qu'on peut supposer qu'à un certain pas d'appoximation, on a déjà un des deux points, x_∞ par exemple, et qu'on n'a plus qu'à approximer l'autre, y_∞.

On peut arguer qu'il peut y avoir des schémas d'approximation dans lesquels on ne peut avoir à chaque instant, que des approximations des deux coordonnées. La définition du faisceau produit se modifie alors ainsi :

$$\beta^*(\langle x,y\rangle) = \{\sigma_1 \times \sigma_2 : \sigma_1 \in \beta(x),\ \sigma_2 \in \beta'(y)\} \cup$$
$$\cup\ a(x,y) \cup b(x,y)$$

$$a(x,y) = \{x\} \times \beta'(y) \quad \text{si} \quad x \in N_{\lambda,\beta}(X)$$
$$= \emptyset \quad \text{sinon}$$

$$b(x,y) = \beta(x) \times \{y\} \quad \text{si} \quad y \in N_{\lambda',\beta'}(X)$$
$$= \emptyset \quad \text{sinon}$$

L'énoncé de la proposition précédente se complique alors quelque peu et donne

PROPOSITION 1.4. bis : Soient X, Y, Z des faisceaux $f : X \times Y \longrightarrow Z$ une fonction. Alors

(i) si $\forall x_o \quad g_{x_o} : y \mapsto f(x_o,y)$ et $\forall y_o \quad h_{y_o} : x \mapsto f(x,y_o)$ sont régulières alors f est régulière

(ii) Si f est régulière de $X \times Y$ dans Z, alors

$\forall x_o \in N_{\lambda\beta}(X) \quad g_{x_o} : y \mapsto f(x_o,y)$ est régulière

$\forall y_o \in N_{\lambda'\beta'}(Y) \quad h_{y_o} : x \mapsto f(x,y_o)$ est régulière

(iii) Si le faisceau Z vérifie la propriété de Fubini forte :

$$\lim_{ij} w_{ij} = \lim_i \lim_j w_{ij} = \lim_j \lim_i w_{ij}$$

$\forall$ famille w_{ij} telle que l'un des termes de l'égalité existe, alors f régulière sur $X \times Y \Rightarrow f$ régulière pour chacune de ses variables. □

DEMONSTRATION : (i) même chose que pour la proposition 1.4, en utilisant la propriété de Fubini

(ii) même chose que pour 1.4 en utilisant les modifications de la fibre.

(iii) montrons que $g_{x_0} : y \mapsto f(x_0,y)$ est régulière

$$g_{x_0}(y) = f(x_0,y) = f(\lim \sigma_1, \lim \sigma_2) \text{(si } \sigma_1 \in \beta(x),\ \sigma_2 \in \beta'(y))$$

$= \text{(régularité de } f) \lim\limits_{\sigma_1\ \sigma_2} f(\sigma_1,\sigma_2) = \text{(Fubini forte)} =$

$\lim\limits_{\sigma_2} (\lim\limits_{\sigma_1} f(\sigma_1,\sigma_2)) =$

$\lim \{(\lim\{f(u,v) : u \in \sigma_1\}) : v \in \sigma_2\} =$

$(\sigma_1 \times \{v\} \in \beta(x_0,v)$ et g_v régulière d'après (i))

$= \lim \{f(x_0,v) \ ; v \in \sigma_2\} = \lim\limits_{\sigma_2} g_{x_0}(\sigma_2)$ i.e. g_{x_0} régulière.

Même chose pour $x \longrightarrow f(x,y_0)$.qed □

Notons que la condition de (iii) est vérifiée par exemple pour les faisceaux ordonnés que nous examinons plus loin.

On voit aisément que ce qui précède généralise ce qui est déjà connu pour les treillis continus et les cpo's. Si l'on suppose maintenant que les espaces de fonctions régulières sont munis d'une structure de faisceau (cf. Chapitre III), on peut alors énoncer deux résultats intéressants : la régularité de l'application et la régularité de l'abstraction. Pour ce on a besoin d'une restriction supplémentaire.

Remarquons que toute topologie $\mathcal{T}$ sur un espace X, si elle est T_0, définit un faisceau : il suffit de prendre pour éléments des fibres les suites généralisées (1) de X convergentes pour $\mathcal{T}$ i.e. $\beta : x \mapsto \{(y_n) \subseteq X : (y_n)$ converge vers x pour $\mathcal{T}\}$. L'on voit donc que se donner un faisceau revient à se donner une notion de

(1) appelées "nets" en anglais.

convergence, et l'on sait (cf. Dugundji par exemple), que la donnée d'une notion de convergence, par contraste avec celle de fermeture, ouverts,..., ne permet pas dans le cas général de reconstruire une topologie, car elle est trop faible.

L'on sait que si X,Y sont des espaces topologiques, $(X \longrightarrow Y)$ l'espace des fonctions continues de X dans Y, alors toute topologie "raisonnable" sur $(X \longrightarrow Y)$ est telle que $f_i \longrightarrow_{(X \to Y)} f \Longrightarrow \forall x \in X$ $f_i(x) \longrightarrow_Y f(x)$ i.e. $\lim_{i(X \to Y)} f_i = f \Longrightarrow \forall x \in X \ \lim_{i\,Y} f_i(x) = f(x)$ ou encore $f = \lambda x.\ \lim_{i\,Y} f_i(x) = \lim_i f_i$. On va définir une notion analogue pour les faisceaux, en ne gardant que l'aspect "convergence" de l'assertion précédente.

<u>DEFINITION</u> : Soit X,Y des faisceaux. Alors on dira qu'un faisceau sur $[X \longrightarrow Y]$ est <u>fidèle</u> si et seulement si $\forall f \in [X \to Y]$ $\forall (f_i) \in \beta(f)$

$$\lim_{i\,[X \to Y]} f_i = f \Longrightarrow \forall x \in X \quad \lim_{i\,Y} f_i(x) = f(x)$$

i.e. $f = \lambda x\ .\ \lim_{i\,Y} f_i(x) = \lim_i f_i$

(On dira encore que le faisceau sur $[X \longrightarrow Y]$ respecte l'évaluation). □

<u>DEFINITION</u> : Soit X,Y des faisceaux ; une fonction $f : X \longrightarrow Y$ est <u>faiblement régulière</u> en $x \in X$ ssi,

$$\forall\ \sigma \in \beta(x) \quad \lim_Y f(\sigma) \text{ existe} \Longrightarrow f(x) = \lim_Y f(\sigma).$$

<u>1.5. LEMME</u> : (régularité de l'abstraction).

Soit X,Y,Z des faisceaux, $f \in [X \times Y \longrightarrow Z]$ et $g_f : x \longrightarrow \lambda y \in Y.\ f(x,y)$.

(i) Si $[Y \longrightarrow Z]$ est muni d'une structure de faisceau fidèle, alors $g_f : X \longrightarrow [Y \longrightarrow Z]$ est faiblement régulière.

(ii) Si $[X \times Y \longrightarrow Z]$ est muni d'une structure de faisceau fidèle alors $\lambda f.g_f : [X \times Y \longrightarrow Z] \longrightarrow [X \longrightarrow [Y \longrightarrow Z]]$ est une fonction faiblement régulière.

(iii) Si dans (i) et (ii) les faisceaux de $[Y \longrightarrow Z]$ et de $[X \times Y \longrightarrow Z]$ sont ceux de la convergence simple, i.e.

$\forall f \in [A \longrightarrow B]$ $\forall (f_i) \in \beta(f)$ on a

$$\lim_i{}_{[A \to B]} f_i = f \Longleftrightarrow \forall x \in A \ \lim_i{}_B f_i(x) = f(x)$$

ou bien si dans $[Y \longrightarrow Z]$ et $[X \longrightarrow [Y \longrightarrow Z]]$ toutes les parties ont une limite, alors la propriété "faiblement régulière" peut être remplacée par "régulière" dans les assertions (i) et (ii).

DEMONSTRATION :

(i) soit $f \in [X \times Y \longrightarrow Z]$, $g_f : x \longrightarrow \lambda y \in Y.f(x,y)$, $x \in X$ et $\sigma \in \beta(x)$, alors si $\lim g_f(\sigma)$ existe,

$$\lim g_f(\sigma) = \lim_\sigma \ \lambda y \in Y \ . \ f(\sigma,y) = \text{(fidélité)}$$

$\lambda y \in Y \ . \ \lim\limits_\sigma f(\sigma,y) =$ (régularité de f et prop. 1.3)

$\lambda y \in Y \ . \ f(\lim \sigma,y) = g_f(\lim \sigma) = g_f(x)$ d'où la faible régularité de g_f.

(ii) Soit $L = \lambda f.g_f$, $h \in [X \times Y \longrightarrow Z]$ et $F \in \beta(h)$, supposons que $\lim L(F)$ existe

$\lim L(F) = \lim g_F = \lim \lambda x.\lambda y.F(x,y) =$ (fidélité) $\lambda x.\lambda y.\lim F(x,y) =$ (fidélité et existence de $\lim F = h$) $= \lambda x.\lambda y.\ (\lim F)(x,y) =$ $(\lambda f.g_f)(\lim F) = L(F)$ d'où L faiblement régulière.

(iii) évident en regardant les démonstrations de (i), (ii) □

1.6. LEMME : régularité de l'application :

Soient X,Y des faisceaux, $[X \longrightarrow Y]$ muni d'une structure de faisceau fidèle ; définissons

$$A_p : [X \longrightarrow Y] \times X \longrightarrow Y \qquad A_p(f,x) = f(x)$$

Alors A_p est régulière (parfois appelée évaluation).

<u>DEMONSTRATION</u> : D'après proposition 1.4, il suffit de montrer que A_p est régulière pour chaque variable

(i) $A_p\ (f_o,.) : x \longrightarrow f_o(x)$ est régulière car f_o l'est

(ii) $A_p(.,x_o) : f \longrightarrow f(x_o)$? régulière

$$\sigma \in \beta(f) \Longrightarrow A_p(f,x_o) = A_p\ (\lim \sigma, x_o) =$$

$$(\lim_{[X \longrightarrow Y]} \sigma)(x_o) = (\text{fidélité})\ \ \lim_Y(\sigma(x_o)) =$$

$$\lim_Y(\{A_p(g,x_o) : g \in \sigma\}) = \lim_Y A_p(\sigma,x_o)$$

$\Longrightarrow A_p(.,x_o)$ est régulière.

d'où le lemme. □

CHAPITRE II

FAISCEAUX ORDONNES

°°°

Ce chapitre est consacré à l'étude d'une classe de faisceaux qui contient la plupart des structures utilisées en informatique mentionnées plus haut. Dans ces faisceaux les limites seront prises soit par borne supérieure $\sqcup$ (construction de Scott et constructions dérivées - motto : " $\sqcup$ = cumul de l'information"), soit par borne inférieure $\sqcap$ (construction de Nolin en ce qui concerne les algorithmes propres, sémantique d'Arnold et Nivat - motto : " $\sqcap$ = réduction de l'indétermination"). Le fait que $\lambda \in \{\sqcup, \sqcap\}$ implique que la propriété de Fubini (forte) est toujours vérifiée et que la fonction (point fixe) calculée par une procédure récursive sera le plus petit / plus grand point fixe de la fonctionnelle associée à la définition de cette procédure.

Nous étudions ici les faisceaux ordonnés à travers deux de leurs sous-classes : les faisceaux ordonnés élémentaires et les faisceaux ordonnés moniques.

I - <u>FAISCEAUX ORDONNES ELEMENTAIRES</u>

Des objets proches de ceux-ci ont été appelés "<u>ordres élémentaires</u>" dans [NAI 78]. La notion présentée ici est un raffinement, dans un but de clarté et de cohérence, de celle d'ordre élémentaire. En effet le schéma suivant avait alors été utilisé :

<u>DEFINITION</u> 1 : Soit X un ensemble, $\sqsubseteq$ un ordre partiel sur X, λ une fonction $\in \{\sqcup, \sqcap\}$

1. <u>fonction de densité</u> : on appellera fonction de densité toute restriction (au sens de la théorie des ensembles) de la fonction $\lambda : S \longrightarrow \lambda(S)$

2. <u>densité</u> : soit $P \subseteq X$ un sous-ensemble de X. φ une fonction de densité sur X. On dira que P est φ-dense dans X ssi $\exists x \in X \quad \exists \sigma \subseteq P \quad x = \varphi(\sigma)$

3. rationalité : soit φ une fonction de densité sur X. Un élément $x \in X$ et φ-rationnel ssi

$$\forall S \subseteq X \quad x = \varphi(S) \Rightarrow x \in S$$

DEFINITION 2 : Soit X un ensemble. Un ordre élémentaire sur X est un couple $\langle \subseteq, \varphi \rangle$ où $\subseteq$ est un ordre partiel et φ une fonction de densité, tel que l'ensemble des éléments φ-rationnels est φ-dense dans X.

Si $\langle \subseteq, \varphi \rangle$ est un ordre élémentaire sur X, alors on notera Rt l'ensemble des éléments φ-rationnels. Comme Rt est φ-dense dans X, on a :

$$\forall x \in X \qquad \exists \sigma \subseteq Rt \qquad x = \varphi(\sigma) = \lambda(\sigma)$$

i.e. pour tout $x \in X$ l'ensemble

$$\beta(x) = \{\sigma \subseteq Rt : x = \varphi(\sigma) = \lambda(\sigma)\}$$

n'est pas vide. Ceci définit un faisceau $\langle \lambda, \beta \rangle$ sur l'espace X, dont Rt est le noyau, et les fibres images de $\mathcal{P}(X) \hookrightarrow \mathcal{F}(X)$. Par définition ce faisceau est ordonné. Il est élémentaire car :

1. $u \in X$ rationnel au sens des faisceaux $\Rightarrow$

$$\forall \sigma \in \beta(u) \quad u \in \sigma \quad \Rightarrow \quad \text{(puisque } \beta(u) =$$
$$\varphi^{-1}(u) \cap \mathcal{P}(Rt)) \; (\forall \sigma \subseteq X \cap Rt \quad u = \varphi(\sigma) \Longrightarrow u \in \sigma).$$

Comme $\exists \sigma \subseteq X \cap Rt \quad x = \varphi(\sigma)$, il vient

$$\exists \sigma \subseteq X \cap Rt \quad u \in \sigma \quad \text{i.e. } u \in Rt \quad \text{i.e.}$$

u est φ-rationnel.

2. réciproquement u φ-rationnel $\Longrightarrow$

$\forall \sigma \in \beta(u) \quad u \in \sigma$ i.e. u rationnel au sens des faisceaux. Donc $Rt \subseteq N(X)$.

D'où finalement par définition du noyau $N(X) = Rt$.

Réciproquement tout faisceau ordonné élémentaire définit un ordre élémentaire : si $\langle\lambda,\beta\rangle$ est un tel faisceau, il suffit de voir que

$$\lambda : \sigma \mapsto \lambda(\sigma) \in X$$
$$\sigma : I \to X,\ i \mapsto x_i = \sigma(i)$$

λ peut être factorisée

$$\underbrace{\sigma \mapsto \mathrm{Im}(\sigma) \mapsto \psi(\mathrm{Im}(\sigma))}_{\lambda} = \lambda(\sigma)$$

Le $\Psi \in \{\sqcup, \sqcap\}$ tel que $\lambda = \Psi \circ \mathrm{Im}$ est noté Ψ_λ.

$$\beta(X) = \{\sigma \in \mathcal{F}(X) : \exists x \in X \quad \sigma \in \beta(x)\}$$
$$(\mathrm{Im} \circ \beta)(X) = \{\mathrm{Im}(\sigma) \in \mathcal{P}(X) : \sigma \in \mathcal{F}(X) \text{ et } \exists x \in X \quad \sigma \in \beta(x)\}.$$

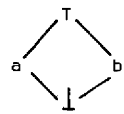
Il suffit de poser $\varphi = \Psi_\lambda|_{(\mathrm{Im} \circ \beta)(X)}$ c'est-à-dire que $(\mathrm{Im} \circ \beta)$ engendre le domaine de φ, et on a encore que $N(X) = Rt$. Un ordre élémentaire peut donc être vue comme le représentant canonique d'une classe d'équivalence de faisceaux ordonnés élémentaires.

Notons aussi que si φ est une fonction de densité, $\mathrm{Dom}(\varphi)$ correspond à la notion de subset-system complet au sens de [ADJ].

N.B. Dans la suite on considère souvent les faisceaux ordonnés élémentaires modulo cette factorisation, i.e. à travers les ordres élémentaires représentant leur $\equiv$-classe d'équivalence (cf. chapitre I). □

Noter qu'une fonction de densité φ étant donnée les éléments φ-rationnels ne sont pas toujours φ-denses comme on peut le voir dans les exemples suivants : □

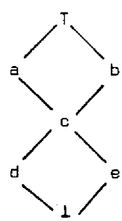
(i) X est le treillis complet

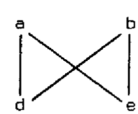


$\lambda = \cup$ et la fonction de densité a pour domaine $\mathrm{Dom}(\varphi) = \mathcal{P}(X) - \{\{a\},\{a,\bot\}\}$. Les éléments φ-rationnels sont $\bot$, a, b et T est le seul élément non rationnel. Mais $\forall S \in \mathcal{P}(X)$ $a \neq \varphi(S)$, donc $<\subseteq,\varphi>$ n'est pas un ordre élémentaire sur X.

(ii) De même si on a $\psi = \cup$ et s : X = l'intervalle ouvert de nombres réels $]0,1[= \{x \in \mathbb{R} : 0 < x < 1\}$ ordonné de faiçon usuelle, alors il n'existe aucun element ψ-rationnel, puisque tout x est borne supérieure de l'ensemble des éléments <u>strictement</u> plus petits que lui - Par conséquent les éléments ψ-rationnels ne sont pas ψ-denses dans X .

D'autre part l'ensemble des éléments rationnels d'un treillis complet n'est pas nécessairement un (sup-) semilattice. Ainsi

X =  et $\varphi = \cup$

a pour rationnels 

qui n'est pas un (sup-) semilattice.

2.1. LEMME : Soit X un ensemble ordonné par $\subseteq$, si la fonction de densité φ est partout définie et $\varphi = \bigcup$, alors $\forall x \in X$ x est φ-rationnel si et seulement si il existe un unique $b \in X$ tel que $b \subseteq x$ et

$$]b,x[= \{c \in X : b \subsetneq c \subsetneq x\} = \phi \qquad \square$$

DEMONSTRATION :

$\Longrightarrow$: x φ-rationnel $\Longleftrightarrow \forall S \subseteq X$ $x = \varphi(S) \Rightarrow x \in S$.

Soit $S_1 = \{y \in X : y \subseteq x \text{ et } y \neq x\}$, $x \notin S_1$

$\bigcup S_1 \neq x$ sinon (x rationnel) $x \in S_1$

soit $b = \bigcup S_1$ (existe car φ partout définie)

$(\forall y \in S_1 \quad y \subseteq x) \Rightarrow \bigcup S_1 = b \subseteq x \Longrightarrow b \in S_1$

$\Longrightarrow S_1 = {\downarrow} b$. D'où ${\downarrow} x = \{x\} \cup {\downarrow} b$ et $]b,x[= \{y : b \subsetneq y \subsetneq x\} = \phi$

Le b est unique par construction car si b' est tel que $b' \subseteq x$ et $]b',x[= \phi$ alors $b' \in S_1 = {\downarrow} b$ i.e. $b' \subseteq b$ i.e. $]b',x[= \phi$ ssi $b = b'$.

$\Longleftarrow$: réciproquement : soit $x \in X$ tel que $\exists!\, b \in X$ $b \subseteq x$ et $]b,x[= \phi$ i.e. ${\downarrow} x = \{x\} \cup {\downarrow} b$. Supposons x non rationnel i.e. $\exists S \subseteq X$, $x = \bigcup S$ et $x \notin S$.

$x = \bigcup S \Longrightarrow S \subseteq {\downarrow} x = \{x\} \cup {\downarrow} b \Longrightarrow S \subseteq {\downarrow} b$

puisque $x \notin S \Longrightarrow \bigcup S \subseteq b \subsetneq x \Rightarrow \bigcup S \subsetneq x$

contradiction $\Rightarrow x$ rationnel. $\square$

On a un résultat dual pour $\varphi = \bigcap$.

2.2. LEMME : (i) Si l'ensemble ordonné X est noethérien (i.e. toute chaine ascendante est stationnaire) et si la fonction de densité $\varphi = \bigcap$ est surjective, alors les éléments φ-rationnels sont φ-denses dans X.

(ii) Dualement si X est artinien et $\varphi = \bigcup$ surjective alors les éléments φ-rationnels sont φ-denses dans X.

DEMONSTRATION : Soit $\varphi = \sqcap$. On veut montrer

$$\forall x \in X \quad \exists \sigma \subsetneq Rt \quad x = \sqcap \sigma$$

1er cas : $x \in Rt$, alors $\sigma = \{x\}$

2e cas : $x \notin Rt \Longrightarrow S \subseteq X \quad x = \sqcap S$ et $x \notin S$.

Il faut maintenant examiner les éléments de S, s'ils sont tous rationnels on a fini, sinon soit $x_1 \in S$ et $x_1 \notin Rt$. Alors

$$\exists S' \subseteq X \quad x_1 = \sqcap S_1 \text{ et } x_1 \notin S'$$

(on a nécessairement $x \subsetneq x_1$), et nous itérons le processus avec S_1. On construit ainsi une famille de chaines ascendantes

$$x \subsetneq x_1 \subsetneq x_2 \subsetneq \dots \text{ etc}$$

comme X est noethérien, toutes ces suites sont stationnaires et leurs limites sont rationnelles par définition. L'ensemble de ces limites augmenté des éléments rationnels rencontrés durant ce processus fournit une décomposition de l'élément initial x, i.e. quelque $\sigma \subsetneq Rt$ tel que $x = \sqcap \sigma$. □

Remarque : Comme le traitement pour $\lambda = \sqcup$ et $\lambda = \sqcap$ est symétrique, nous supposons à partir de maintenant que $\lambda = \sqcup$, sauf mention expresse.

Quelques constructions simples d'ordres élémentaires :

(i) o.e. trivial : Si X est un ensemble ordonné par $\subsetneq$, soit la fonction de densité $\varphi = \sqcup|_{\{\text{singletons}\}}$. Alors tout élément de X est rationnel et on a un o.e.

(ii) o.e. canonique : Si X est un ensemble ordonné par $\subsetneq$, soit $\varphi = \sqcup$. Si les rationnels sont denses alors on a un o.e sur X dit o.e canonique.

(iii) o.e. induit : Si W est un ensemble ordonné par $\leqslant$, $X \subseteq W$ un sous-ensemble ordonné par l'ordre $\subseteq$ induit par $\leqslant$, soit la fonction de densité

$$\varphi : \mathcal{P}(X) \to X$$

φ : $S \mapsto$ borne sup. de S dans W si elle existe et $\in$ X, non défini sinon. Si les rationnels sont denses alors on a un o.e sur X. □

2.3. Fait : Soit X un sup-similattice qui a une structure d'o.e canonique. Si X n'a que des rationnels, alors X est totalement ordonné.

DEMONSTRATION : Soit $x, y \in X$, X est un sup-similattice $\Longrightarrow x \cup y \in X$, X a un o.e canonique pour $\lambda = \cup \Longrightarrow x = x \cup y$ ou $y = x \cup y$ i.e. les deux éléments sont ordonnés $\Longrightarrow$ X est totalement ordonné. □

Notons encore que la classe des faisceaux ordonnés élémentaires, et celle des ordres élémentaires, sont closes par produit cartésien et coproduit (somme).

Exemples : Soit $X = \mathbb{R}$ l'ensemble des nombres réels et $P_1(\mathbb{Q}) = \{U \subseteq \mathbb{Q} : U$ est majoré mais non borné supérieurement dans $\mathbb{Q}\}$. Soit $\varphi = \cup|_{P_1(\mathbb{Q})}$ la fonction de densité. Les éléments rationnels sont alors exactement les nombres rationnels ; leur densité dans $\mathbb{R}$ résulte de la construction de Cantor-Dedekind que nous simulons ici. Nous avons ainsi un faisceau ordonné élémentaire. Donc dans un foe les éléments rationnels peuvent être considérés comme ceux qui sont finis, rationnels, commensurables, les éléments non rationnels étant "non finis", irrationnels, incommensurables. Dans cet exemple $\forall x \in X$, le fibre $\beta(X)$ de x contient toutes les suites de Cauchy croissantes de rationnels convergeant vers x. Cette fibre contient aussi un plus grand élément pour l'inclusion qui est le segment inférieur (ou section commençante de rationnels) $\{y \in \mathbb{Q} : y \leqslant x\}$ □

Cet exemple est à l'origine de notre terminologie : les nombres rationnels sont exactement les éléments qui sont rationnels au sens de ce faisceau ordonné. □

(ii) le faisceau P_ω : $X = P_\omega = \{S : S \subsetneq \mathbb{N}\}$ est ordonné par inclusion $\subsetneq$ et muni de l'union $\cup$. Le couple $\langle \subseteq, \cup \rangle$ définit un faisceau ordonné élémentaire (foe canonique) sur P_ω. Les éléments rationnels sont l'ensemble vide ϕ, et les singletons $\{n\}$, $n \in \mathbb{N}$. Les rationnels forment un semi-treillis plat (ou cpo discret). Le lemme 2.1 est trivialement vérifié. Le faisceau P_ω est une connection de Galois (trivialement). (cf. Scott 1976).

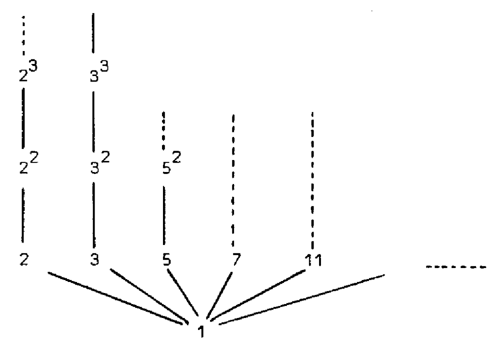
(iii) Soit $X = \mathbb{N}^* =$ l'ensemble des entiers naturels non nuls ordonné par

$$p \subsetneq n \iff p \text{ divise } n \;(\iff n \text{ multiple de } p).$$

Soit la fonction de densité

$$\varphi = \cup : \{x_1, \ldots, x_n\} \longmapsto \text{plus petit commun multiple de } x_1, \ldots, x_n$$

Les éléments rationnels sont alors 1 et les p^m où p est premier et m un entier. Ils sont -denses (théorème de factorisation de Gauss). On a donc ainsi un o.e $\langle \subseteq, \cup \rangle$ sur $\mathbb{N}$ (foe canonique). L'ensemble des rationnels a une structure plus complexe que dans l'exemple numéro (i) (semi-treillis complet) :



On a encore ici une connection de Galois.

(iv) Le faisceau N : soit X un modèle ensembliste d'une collection d'algorithme $\mathcal{A}$ (s'il existe) ordonné par inclusion. X forme un treillis complet. Notant que 1. les éléments atomiques au sens de Nolin sont compacts au sens des prepartitions, 2. tout élément compact au sens d'un faisceau est rationnel pour ce faisceau, 3. que les éléments atomiques sont $\cup$-denses dans $\mathcal{A}$ (cf. Prolégomènes), il vient que le couple $\langle \subseteq, \cup \rangle$ définit un faisceau ordonné élémentaire sur X. (En fait ici on peut noter que la distributivité entraîne l'équivalence de la compacité et de la rationnalité). Ce faisceau est un foe induit, et le fait 2.3 permet d'éclairer un ressort de la théorie des algorithmes : le jeu entre l'union U et l'union complétée $\bar{U}$. On sait que :

PROPOSITION (Nolin) : ϕ, T, les singletons et les algorithmes propres sont atomiques. $\square$

Il s'ensuit que si l'oe était canonique, l'espace des algorithmes serait linéairement ordonné.

Le faisceau N n'est pas une connection de Galois en général (cf. Lemme 2.16).

(v) le faisceau SW : Soit $X = \hat{D}$ une structure de type au sens de Shamir et Wadge. Le faisceau $\langle \lambda, \beta \rangle$ donné au chapitre I est évidemment ordonné élémentaire. Les rationnels sont les $\downarrow d$, $d \in D$; Rt est isomorphe au cpo D par

$$d \mapsto \downarrow d : D \to Rt$$
$$x \mapsto \cup x : Rt \to D$$

En fait Rt peut être considéré comme une copie de D dans $\hat{D}$. $\square$

DEFINITION : Un faisceau $\langle \lambda, \beta \rangle$ sur X est normalisé ssi $\forall x \in X \; s(x) \in \beta(x)$, i.e. la fibre de x contient le spectre de x.

2.4. Fait : Pour tout foe $\langle \subseteq, \varphi \rangle$ sur X il existe un foe $\langle \subseteq, \bar{\varphi} \rangle$ normalisé sur X tel que (i) $\bar{\varphi}$ est un prolongement de φ

(ii) le noyau de $\langle \subseteq, \varphi \rangle$ coïncide avec celui de $\langle \subseteq, \bar{\varphi} \rangle$.

DEMONSTRATION : il suffit de prendre

$$\bar{\varphi} : S \mapsto \mathcal{P}(S) \quad \forall S \in \mathrm{Dom}(\varphi)$$
$$s(x) \mapsto x \quad \forall x \in X$$

DEFINITION : foe plat : Soit X un foe. On dira que X est plat ssi pour tous rationnels $x,y \in Rt(X)$ on a :

$$x \sqsubseteq y \Leftrightarrow (x=y) \text{ ou } (x=\perp)$$

où $\perp$ désigne (s'il existe) le plus petit élément de X.

DEFINITION : finitude pour une famille d'ensembles.

Soit X un foe, $r \in X$, F une famille de parties de X possèdant des bornes supérieures. Alors r est fini (ou compact) pour F ssi

$$\forall S \in F \quad r \sqsubseteq \sqcup S \quad \exists s \in S \quad r \sqsubseteq s.$$

Ceci généralise une notion bien connue lorsque F est l'ensemble des parties dirigées de X, et correspond à la Z-compacité d'ADJ.

2.5. LEMME : Si X est un foe plat dont les rationnels sont finis pour la famille d'ensembles $F = \bigsqcup_{x \in X} \beta(x)$ (i.e. finis pour le faisceau), alors :

(i) $\forall x \in X$ la fibre de x contient au plus deux éléments (i.e. $|\beta(x)| \leq 2$)

(ii) si $\exists\, S_1, S_2$ distincts $\in \beta(x)$, alors X possède un élément $\perp$ et S_1, S_2 ne diffèrent que par cet élément

(iii) si $x,y \in X$ sont tels que

$\beta(x) = \{s(x)\}$, $\beta(y) = \{s(y)\}$, alors $x \sqsubseteq y \Longrightarrow$

$(\perp \in X \text{ et } s(x) - \{\perp\} \subseteq s(y))$ ou $(\perp \notin X \text{ et } s(x) \subseteq s(y))$ □

Cela veut dire qu'il manque très peu de choses à X pour être un f.o.m.

<u>DEMONSTRATION</u> : Soit $x \in X \quad S_1, S_2 \in \beta(x)$

$$\forall a \in S_1 \quad a \sqsubseteq \sqcup S_2 \Longrightarrow \exists u(a) \in S_2 \quad a \sqsubseteq u(a)$$

$$\forall b \in S_2 \quad b \sqsubseteq \sqcup S_1 \Longrightarrow \exists t(b) \in S_1 \quad b \sqsubseteq t(b)$$

Comme X est un foe plat.

$\forall a \in S_1 \quad a = \perp$ (s'il existe) ou $a \in S_2$

$\forall b \in S_2 \quad b = \perp$ (s'il existe) ou $b \in S_1$

d'où les clauses (i) et (ii) du lemme.

Maintenant si $\beta(x) = \{s(x)\}$, $\beta(y) = \{s(y)\}$ et $x \sqsubseteq y$, on a

$$\forall a \in s(x) \quad a \sqsubseteq \sqcup s(y) \Longrightarrow \exists b \in s(y) \; a \sqsubseteq b$$

1er cas : $\perp \notin X \Longrightarrow a = b \in s(y)$ i.e.

$s(x) \subseteq s(y)$

2e cas : $\perp \in X$: 2 sous-cas $a = \perp$

$a \neq \perp \Longrightarrow a = b \in s(y)$

$\therefore s(x) - \{\perp\} \subseteq s(y)$ □

Nous avons considéré la notion de reconstruction des éléments au moyen des rationnels, celle de spectre peut être utilisée de la même manière. Plus précisément, nous allons examiner la forme la plus pure (ou minimale si l'on préfère) de f.o. normalisé, celle de faisceau ordonné monique.

II - FAISCEAUX ORDONNES MONIQUES

Nous utilisons d'abord une approche analogue à celle de Hofmann [SCS] I-1.9, 1.10.

2.6. PROPOSITION : Soit X un ensemble partiellement ordonné, $\mathbb{F}$ l'ensemble des parties non vides de X, $\mathcal{S}$ l'ensemble de toutes les fonctions monotones croissantes $s : X \mapsto \mathbb{F}$ telles que $\forall x \in X \; s(x) \subseteq \downarrow x$. Soit R l'ensemble des relations binaires $\prec$ sur X telles que

(i) $x \prec y \Longrightarrow x \leq y$

(ii) $x \prec y$ et $y \subseteq z \Longrightarrow x \prec z$

(iii) $\forall x \in X \quad \exists y \in X \quad y \prec x$

Alors il existe une bijection entre $\mathcal{S}$ et R qui associe à chaque $s \in \mathcal{S}$ la relation $\prec_s$ donnée par $x \prec_s y \Longleftrightarrow x \in s(y)$. Son inverse est donnée par

$$R \ni \prec \longmapsto s_\prec : X \to \mathbb{F}$$
$$x \mapsto \{y \in X : y \prec x\} \quad .$$

□

DEMONSTRATION : Soit $s \in \mathcal{S}$ et $x \prec_s y \Longleftrightarrow x \in s(y)$. Alors

(i) $x \prec_s y \Longrightarrow x \in s(y) \subseteq \downarrow y \Longrightarrow x \leq y$

(ii) $x \prec_s y$ et $y \leq z \Longrightarrow x \in s(y) \subseteq s(z) \Longrightarrow x \prec_s z$

(iii) $\forall x \in X \quad s(x) \neq \emptyset \Longrightarrow \exists y \in s(x)$ i.e. $y \prec_s x$.

Réciproquement soit R et $s\ (x) = \{y \in X : y \prec x\}$ alors $s_\prec(x) \subseteq \downarrow x$ par (i), s est monotone par (ii) et $Im(s\ ) \in \mathbb{F}$ par (iii). Donc $s_\prec \in \mathcal{S}$. La bijection est évidente.

□

TERMINOLOGIE : si $x \prec y$ on dira que x approxime y.

Exemples : (i) la relation "way-below" (bien en dessous) dans un treillis complet définie par :

$$x \ll_i y \iff x \in \cap \{ I \in Id_{cv} X \ : y \sqsubseteq \sqcup I \}$$

est un élément de R (cf. [SCS] pp 39, Prop. 1.2)

(ii) le faisceau N_2 : $x \prec y \iff x$ est atomique et $x \sqsubseteq y$

(iii) le faisceau SW : $x \prec y \iff \exists d \in D \quad x = \downarrow d$ et $x \sqsubseteq y$.

Noter que les éléments de R ne sont généralement pas des ordres au sens usuel : ce ne sont même pas des préordres puisque $(v \prec v \iff v \in X)$ est faux. Ce sont des ordres stricts cependant.

Les éléments de R seront appelés ordres auxiliaires (l'expression est empruntée à [SCS] bien que nos objets soient plus généraux), et ceux de $\mathcal{S}$ les fonctions-spectrales. Les parties s (x) seront appelées les $\prec$-spectres.

Il est clair qu'un ordre auxiliaire est un faisceau ordonné monique (en abrégé fom ou faisceau s'il n'y a pas d'ambiguïté)

$$X \underset{s}{\overset{\sqcup}{\leftrightarrows}} \mathcal{P}(X)$$

si et seulement si $\forall x \in X \quad x = \sqcup(s(x))$.

La classe des fom est close par produit cartésien et coproduit.

2.7. PROPOSITION : (i) un ordre auxiliaire $\prec$ est un fom

$$X \underset{s_\prec}{\overset{\sqcup}{\leftrightarrows}} \mathcal{P}(X)$$

si et seulement si $\forall x \in X \quad x = \sqcup(s_\prec(x))$.

(ii) Tout foe permet de définir un fom, qui est de plus une connection de Galois, en prenant la fonction spectrale

$$s : x \mapsto s(x) = \{ v \in Rt : v \sqsubseteq x \}$$ □

<u>DEMONSTRATION</u> : évident.

<u>Exemple</u> : Soit X = le magma complété $M^{\infty}(F,V)$ et l'ordre auxiliaire

$$x \prec y \Longleftrightarrow x \text{ est un arbre fini qui approxime } y.$$

$\prec$ définit un fom, les éléments rationnels sont les arbres finis, et les non-rationnels sont les arbres infinis.

$$\forall x \in X \quad s_{\prec}(x) = \{ y \in X : y \text{ est un arbre fini } \sqsubseteq x \}.$$

On a de plus dans $M^{\infty}(F,V)$ une agréable propriété : aucun rationnel n'est limite d'une infinité d'éléments tous différents de lui-même. Mieux : entre deux rationnels distincts comparables. On ne peut trouver qu'un nombre fini de rationnels. (On retrouve cette propriété dans les domaines concrets de Kahn-Plotkin). □

Il apparait que les foe dont les fibres sont réduites au spectre sont à peu près aux fom ce que les treillis algébriques sont aux treillis continus. Nous allons préciser ce point.

<u>2.8. PROPOSITION</u> : Tout cpo algébrique possède une structure de fom élémentaire.

<u>DEMONSTRATION</u> : Il suffit de prendre

$$x \prec y \Longleftrightarrow x \text{ est fini et } x \sqsubseteq y$$

On a un faisceau puisque le cpo est algébrique. Les éléments rationnels sont exactement les éléments finis. □

<u>DEFINITION</u> : Soit X un ensemble partiellement ordonné, s_1, s_2 deux fonctions de $\mathcal{P}(X)$ dans X définissant des fom sur X.

On dit que le faisceau défini par s_1 est <u>plus fin</u> (ou <u>moins grossier</u>) que le faisceau défini par s_2 si et seulement si

$$\forall x \in X \quad s_1(x) \subseteq s_2(x)$$

On note alors $s_1 \leq s_2$ □

<u>2.9. LEMME</u> : Soit X un poset muni de deux fom s_1 et s_2 tels que s_1 est plus fin que s_2. Alors toute fonction $f : X \to Y$ régulière au point x pour le faisceau s_1 est régulière au point x pour le faisceau s_2 □

<u>DEMONSTRATION</u> : $x \in X \quad f(x) = \sqcup \{f(v) : v \in s_1(x)\}$

$$\forall w \in s_2(x)\ f(w) \subseteq f(x) \Longrightarrow \{f(w) : w \in s_2(x)\}$$

est majorée par $f(x)$. Mais $s_1(x) \in s_2(x)$

$\Longrightarrow$

$$\underbrace{\{f(v) : v \in s_1(x)\}}_{\text{borne sup. } f(x)} \subseteq \underbrace{\{f(w) : w \in s_2(x)\}}_{\text{majoré par } f(x)}$$

$\Longrightarrow$

$$f(x) = \sqcup\{f(w) : w \in s_2(x)\}$$ □

On en déduit en particulier que les fonctions normales de Nolin $f : \mathcal{A}_0 \to \mathcal{A}_0$ sont toutes Scott-continues.

<u>DEFINITION</u> : Soit X un poset. Une famille $\mathcal{F}$ de parties de X est <u>isotone</u> ssi

(i) $\forall S \in \mathcal{F} \qquad \sqcup S \in X$

(ii) $\forall S_1, S_2 \in \mathcal{F} \quad \sqcup S_1 \subseteq \sqcup S_2 \Longrightarrow S_1 \leq S_2$ □

On a un résultat analogue aux topologies :

2.10. LEMME : Pour toute famille isotone $\mathfrak{F} \subseteq \mathcal{P}(X)$, il existe un faisceau ordonné monique le moins fin (ou le plus grossier) s tel que

$$\forall x \in X \quad \exists S \in \mathfrak{F} \quad x = \sqcup S \Longrightarrow s(x) = S$$ □

DEMONSTRATION : immédiat. Il suffit de prendre la fonction spectre

$$s : x \mapsto \begin{cases} S \text{ si } \exists S \in \mathfrak{F} \quad x = \sqcup S \\ \downarrow x \quad \text{sinon} \end{cases}$$

2.11. LEMME : Soit X un ensemble partiellement ordonné. Alors l'ensemble des fom sur X ordonné par $\leqslant$ a une structure de semi-treillis complet dont le "bottom" est constitué par le faisceau :

$$s : x \mapsto \downarrow x = \{y \in X : y \sqsubseteq x\}$$ □

DEMONSTRATION : immédiat. Si $\{s_i\}_{i \in I}$ est une famille de fom,

$$(\sqcap_i s_i) : x \mapsto \bigcup_{i \in I} s_i(x)$$ □

III - FAISCEAUX ORDONNES INTERPOLABLES

DEFINITION : 1) Un faisceau $\langle \lambda, \beta \rangle$ sur un ensemble X est interpolable au point $e \in X$ si et seulement si pour tout $x \in X$, on a

$$\forall \sigma \in \beta(x)$$
$$(\exists i \in \mathrm{Dom}(\sigma) \quad e = \sigma(i)) \Longleftrightarrow (\exists l \in \mathrm{Dom}(\sigma),$$
$$u = \sigma(l) \text{ tq } \quad \forall \tau \in \beta(u) \quad \exists j \in \mathrm{Dom}(\sigma) \quad e = \tau(j))$$

(c'est-à-dire e approxime x selon un certain systeme σ, si et seulement si il existe, un $u \in X$ qui approxime x selon ce même système, tel que e approxime u selon tous les systèmes.

2) Un faisceau $\langle \lambda, \beta \rangle$ sur un ensemble X est <u>interpolable</u> ssi il est interpolable en tout point. □

Dans le cas où le faisceau est <u>ordonné monique</u>, la définition précédente se ramène à (puisque σ = s(x) modulo Im) :

$$e \in s(x) \Longleftrightarrow \exists u \in s(x) \quad e \in s(u)$$

ou encore avec les notations d'ordres auxiliaires

$$e \prec x \Longleftrightarrow \exists u \quad e \prec u \prec x$$

Ceci permet de prolonger immédiatement la définition aux <u>ordres auxiliaires</u>.

On note aussi que tous les points d'interpolabilité sont des éléments du noyau, ce qui indique que l'interpolabilité est une façon très forte de tenir compte de la structure du noyau au niveau du faisceau.

<u>Exemple</u> : Les treillis continus sont interpolables (Scott).

<u>2.12. Fait</u> : (i) Les faisceaux ordonnés élémentaires sont interpolables.

(ii) les faiceaux ordonnés sont interpolables en tout point rationnel.

Ainsi les faisceaux N et SW sont interpolables. L'interpolabilité apparait comme une propriété importante dans l'étude des fo. Elle intervient dans l'étude des relations existant entre deux façons (équivalentes dans les treillis continus) de définir la continuité d'un poset au sens de Scott (proposition 2.13), et dans l'utilisation des échelons de Scott pour la construction de faisceaux sur les espaces de fonctions régulières (cf. Chapitre III - §III-1).

Noter que si s_1 et s_2 définissent des foms avec s_1 plus fin que s_2, l'interpolabilité de s_1 entraîne celle de s_2 (pour des raisons évidentes).

<u>2.13. PROPOSITION</u> : Soit X un point muni de sa topologie de Scott. Soient les ordres auxiliaires sur X

$$x \ll_t y \Longleftrightarrow y \in \mathrm{Int}(\uparrow x)$$

$$x \ll_i y \Longleftrightarrow x \in \bigcap \{I \in \mathrm{Id}_{cv}X : y \sqsubseteq \sqcup I\}$$

($\ll_t$ est associé à la topologie, et $\ll_i$ aux idéaux). Alors :

(i) $x \ll_t y \Longrightarrow x \ll_i y$ i.e. $\forall y \quad s_t(y) \subseteq s_i(y)$.

(ii) Si l'ordre auxiliaire défini par $\ll_i$ est interpolable, alors $x \ll_i y \Longrightarrow x \ll_t y$.

(iii) Si les $\ll_i$-spectres sont des supsemilattices et si $\ll_i$ définit un fom sur X, alors ce fom est interpolable.

(iv) Si $\ll_t$ définit un fom sur X et si ses spectres sont dirigés alors ce fom est interpolable et $x \ll_t y \Longleftrightarrow x \ll_i y$.

(v) Si X est un sup-semilattice, alors les $\ll_t$-spectres sont des sup-semilattices et de sections commençantes □

(N.B : On sait que sur les treillis complets $\ll_t$ définit un fom si et seulement si $\ll_i$ définit un fom).

<u>DEMONSTRATION</u> :

(i) "l'approximation topologique entraîne l'approximation par idéaux": $x \ll_t y \Longleftrightarrow y \in \mathrm{Int}(\uparrow x)$

$\Longleftrightarrow$ $\exists$ un ouvert $\mathcal{U}(y)$ tel que (i) $y \in \mathcal{U}$,

(ii) $\mathcal{U} = \uparrow \mathcal{U}$, (iii) $\forall S \subseteq X$ dirigée convergente

$\bigsqcup S \in \mathcal{U} \implies S \cap \mathcal{U} \neq \emptyset$, (iv) $\forall b \in \mathcal{U}\ b \sqsupseteq x$.

Soit I idéal convergent tel que $y \sqsubseteq \bigsqcup I$ alors $\bigsqcup I \in \mathcal{U}(y)$ i.e. $I \cap \mathcal{U}(y) \neq \emptyset$ d'où $x \in I$ puisque I est une section commençante. Donc $x \ll_t y \implies (\forall I \in Id_{cv}X \ \ y \sqsubseteq \bigsqcup I \implies x \in I)$ i.e. $x \ll_i y$.

(ii) réciproquement soit $x \in X$, et

$\hat{\uparrow} x = \{y \in X : x \ll_i y\}$. Il suffit de montrer que $\hat{\uparrow} x$ est ouvert.

(1.) $x \ll_i z \sqsubseteq u \implies x \ll_i u$ donc $\hat{\uparrow} x$ est une section finissante.

(2.) Soit $S \subseteq X$ dirigée et convergente telle que $x \ll_i \bigsqcup S$. Trouver un $s \in S$ tq $x \ll_i s$. Comme $\ll_i$ est interpolable, on a un $w \in X$ tq $x \ll_i w \ll_i \bigsqcup S$, et

$\forall I \in Id_{cv} X \quad w \sqsubseteq \bigsqcup I \implies w \in I$

Prenons $I = \downarrow S$, alors il existe $s \in S \ \ w \sqsubseteq s$ par définition. Donc $x \ll_i w \sqsubseteq s$, d'où $x \ll_i s$, i.e. $s \in \hat{\uparrow} x$. L'ensemble $\hat{\uparrow} x$ est donc ouvert.

(iii) Utilisant un argument de [SCS] on pose

$I_z = \{u \in X : \exists y \ \ u \ll_i y \ll_i z\}$. I_z est un idéal car :

1. les $\ll_i$-spectres sont des idéaux

$(\forall x \in X \quad s_{\ll_i}(x) = \bigcap \{I \in Id_{cv}X : x \sqsubseteq \bigsqcup I\})$

et une intersection d'idéaux est toujours une section commençante et

2. $\forall a,b$ tq $a \ll_i x$ et $b \ll_i x$. $\exists c \in s_{\ll_i}(x)$ $c = (a \sqcup b)$ (supsemilattice) donc les spectres sont dirigés.

3. D'où : $u_1 \ll_i y_1 \ll_i z$ et $u_2 \ll_i y_2 \ll_i z \Rightarrow$

$\Longrightarrow u_1 \sqcup u_2 \ll_i y_1 \sqcup y_2 \ll_i z$ en appliquant le point 2. et proposition 2.6. (ii).

Maintenant soit $\bigsqcup I_z = z'$, alors $z = z'$ car sinon $z = \bigsqcup\{y \in X : y \ll_i x\} \Longrightarrow \exists\, y \ll_i z \quad y \not\sqsubseteq z'$, mais alors

$$y = \bigsqcup\{u \in X : u \ll_i y\} \Longrightarrow \exists\, u \ll_i y, u \not\sqsubseteq z'$$

et $u \ll_i y \ll_i z$ i.e. $u \in I_z$ - contradiction.

Supposons alors $x \ll_i z = \bigsqcup I_z$. Ceci $\Longrightarrow x \in I_z$ par définition de $\ll_i \Longrightarrow \exists\, y \quad x \ll_i y \ll_i z$ par définition de I_z.

(iv) Soit $x \ll_t z$. Notons

$s(z) = s_{\ll_t}(z)$. Alors

$x \ll_t z = \bigsqcup s(z) \Longleftrightarrow \bigsqcup s(z) \in \mathrm{Int}(\uparrow x)$

$s(x)$ dirigé $\Longrightarrow$ (ouvert) $s(z) \cap \mathrm{Int}(\uparrow x) \neq \phi$

$\Longrightarrow \exists\, a \in s(z) \cap \mathrm{Int}(\uparrow x)$ i.e.

$\exists\, a \quad a \ll_t z$ et $x \ll_t a$ qed.

(v) $x \ll_t z \Longleftrightarrow z \in \mathrm{Int}(\uparrow x)$. Soit

$$s(x) = \{y : y \ll_t x\} = \{y : x \in \mathrm{Int}(\uparrow y)\}$$

1. $s(x)$ est une section commençante car $y \in s(x)$, $z \sqsubseteq y \Longrightarrow$

$x \in \mathrm{Int}(\uparrow y) \subseteq \mathrm{Int}(\uparrow z) \Longrightarrow z \in s(x)$

2. $s(x)$ est un sup-simillattice car

$y, z \in s(z) \Longrightarrow x \in \mathrm{Int}(\uparrow y) \cap \mathrm{Int}(\uparrow z) =$

$\mathrm{Int}((\uparrow y) \cap (\uparrow z)) = \mathrm{Int}(\uparrow(y \sqcup x)) \Longrightarrow$

$y \sqcup z \in s(x)$. $\square$

On appellera $x \ll_t y$ "<u>x est topologiquement bien en dessous de y</u>", et $x \ll_i y$ "<u>x est bien en dessous de y</u>". On notera que même sur un treillis complet elles ne sont pas forcément équivalentes : il suffit de prendre

$$X = \{\perp\} \cup (\mathbb{N} \times [0,1]) \cup \{T\}$$

avec $\uparrow(n,r) = (\{n\} \times [r,1]) \cup \{(p,1) : p \gg n\} \cup \{T\}$.

On a $(1,0) \ll_i T$ mais $\neg((1,0) \ll_t T)$.(Exemple cité dans [SCS]). On vérifie bien $s_t(T) \subsetneq s_i(T)$

<u>2.14. COROLLAIRE</u> : Si X est un sup-semilattice et si $\ll_i$ est un fom sur X, alors

(i) $x \ll_t y \Longleftrightarrow x \ll_i t$ (notée $\ll$)

(ii) $a \ll x$ et $b \ll x \Longrightarrow a \cup b \in x$ et les spectres sont des sections commençantes

(iii) il est interpolable.

De façon analogue, si X est un supsemilattice et si $\ll_t$ définit un fom sur X, alors il est interpolable et ses spectres sont des supsemilattices et des sections commençantes.

Montrons encore une classe de faisceaux ordonnés moniques interpolables.

<u>2.15. PROPOSITION</u> : Soit X complet sous condition et l'ordre auxiliaire $x \ll_i y \Longleftrightarrow$

$x \in \bigcap\{I \in Id_{cv}X : y \subseteq \bigcup I\}$. Alors

(i) si $u,v \in X$ sont majorés par $W \in X$, alors il existe $u \cup v$

(ii) $u \leq x \ll_i y \leq z \Longrightarrow u \ll_i z$ ($\Longleftrightarrow$ les spectres sont des sections commençantes)

(iii) les spectres sont dirigés

(iv) $a \ll_i x$ et $b \ll_i x \Longrightarrow a \sqcup b \ll_i x$

(v) si $\ll_i$ définit un fom sur X, alors ce fom est interpolable.

DEMONSTRATION : Si u,v possèdent un majorant, alors l'ensemble des majorants est non vide et possède une borne inférieure. Soit $s(x) = \bigcap \{I \in Id_{cv}X : x \sqsubseteq \bigsqcup I\}$. L'ensemble s(x) est un idéal car 1. si $a,b \in s(x)$ $\forall I \in Id_{cv}X \;\; x \sqsubseteq \bigsqcup I$ et $a,b \in I \Longrightarrow \exists\, m_I \in I$ tq $m_I \sqsupseteq a,b$. Il suffit de prendre

$$m = \sqcap \{m_I \;:\; I \in Id_{cv}X \text{ et } x \sqsubseteq \bigsqcup I\}$$

qui existe puisque $\{m_I : \ldots\}$ est minorée et non vide et $m \in s(x)$, $m \sqsupseteq a,b$.
2. C'est une section commençante car intersection de sections commençantes.

Puisque s(x) est un idéal, $\forall a,b \in s(x)$

$\exists\, m \in s(x) \quad m \sqsupseteq a,b \Longrightarrow a \sqcup b$ existe par (i) et il vient $a \sqcup b \sqsubseteq m \Longrightarrow a \sqcup b \in s(x)$.

L'interpolabilité du fom résulte du même argument que pour Prop. 2.13 (ii) $\square$

En fait l'interpolabilité des faisceaux des deux propositions précédentes met en oeuvre un fait plus général, qui est la stabilité.

DEFINITION : Un fom sur X est stable au point $x \in X$ si et seulement si

$$\forall y \in X \quad s \sqsubseteq y \Longrightarrow s(x) = (\downarrow x) \cap s(y) \qquad \square$$

Cela veut dire en gros que s(x) est un point de passage obligé pour la croissance de tous les spectres s(y), $y \sqsupseteq x$, et qu'on ne peut "grimper" dans l'ordre $\sqsubseteq$ qu'en rajoutant des approximants $\not\sqsubseteq x$.

Exemples de fom stables :

(i) le faisceau SW est stable en tout point car

$s(x) = \{\downarrow d \;:\; d \in x\}$ et $x \sqsubseteq y \Longrightarrow$

$(\downarrow x) \cap s(y) = (\downarrow x) \cap \{\downarrow d : d \in y\}$

comme $\downarrow y = \{u \in \hat{D} : u \subseteq y\}$

$(\downarrow x) \cap s(y) = \{\downarrow d : d \in y$ et $\downarrow d \subset n\}$ =

(y est une section commencante d'où $\downarrow d \subseteq x \Leftrightarrow d \in n$)

$= \{\downarrow d : d \in x$ et $d \in y\} = (x \subseteq y) = \{\downarrow d : d \in x\} = s(x)$

i.e., le faisceau SW est stable.

(ii) Le faisceau N est stable si et seulement si les rationnels sont incomparables entre aux - (En particulier un faisceau N stable est un foe plat). En effet:

$\Rightarrow$: x,y atomiques distincts $\Rightarrow s(x) = \{x\}$, $s(y) = \{y\}$ (les spectres ci sont les prepartitions). Supposons $x \subseteq y$ alors $s'(y) = \{x,y\}$ n'est pas une prepartition, car $x \subsetneq y$, d'où

$\{x\} = s(x) \neq s(y) \cap \downarrow x = \{y\} \cap \downarrow_x = \phi$

contradiction. Done stabilité $\Rightarrow$ les chaînes de rationnels sont toutes réduites a un element.

$\Leftarrow$ reciproquement si les rationnels sont incomparables entre eux, on a w atomique et $w \subseteq x$ et $w \notin s(n) \Rightarrow$

$\Rightarrow w \subseteq \bigcup\{u : u \in s(x)\} \Rightarrow \exists\ u \in s(x) \quad w \subseteq u$

$w = u \in s(x)$ contradiction. Donc

$$\forall\ x,y \in X \quad x \subset y \Rightarrow s(x) = (\downarrow x) \cap s(y)$$

Ce qui decoule en fait de ce que N est une connection de Galois. □

On voit donc qu'il existe des faisceaux interpolables qui ne sont pas stables. Mais comme on le verra plus loin la stabilité permet d'exhiber une large classe de fom interpolables. Précisons d'abord les relations entre les faisceaux stables et les connections de Galois.

2.16. LEMME :

(i) Tout fom qui est une connection de Galois est stable.

(ii) Tout fom sur un espace dirigé X, qui n'est pas une connection de Galois vérifie la condition

$$\exists x \in X \text{ tq le fom n'est pas stable en } x$$

(iii) sur un poset dirigé X les fom stables coïncident avec les connections de Galois tq $\sqcup \circ s = id_X$.

DEMONSTRATION :

(i) les spectres sont des sections commençantes du noyau, d'où la stabilité

(ii) Par hypothèse $\exists x \in X \quad s(x) \neq \downarrow_{N(X)} x$

i.e. $\exists x \in X \quad \exists u \in N(X) \quad u \subseteq x$ et $u \notin s(x)$ mais $u \in N(x) \Longrightarrow \exists w \in X$ $u \in s(w)$

X dirigé $\Longrightarrow \exists\, y \supseteq x, w \quad s(y) \supseteq s(x), s(w)$

$u \in s(w) \subseteq s(y) \Longrightarrow u \in s(y)$.

Donc $u \notin s(x)$ et $u \in (\downarrow x) \cap s(y)$

i.e. x point d'instabilité du fom.

□

(iii) immédiat

Ainsi pour le faisceau N :

rationnels incomparables $\Longrightarrow$ connection de Galois trivialement

rationnels comparables $\Longrightarrow$ il existe des points d'instabilité donc on n'a pas une connection de Galois.

<u>2.17. LEMME</u> : Soit X un poset muni d'un fom, $x \in X$ et $\uparrow x = \{z \in X : x \prec z\}$. Si les spectres des $y \in \uparrow x$ sont dirigés et si le fom est stable sur $\uparrow x$, alors X est interpolable au point x.

<u>DEMONSTRATION</u> : Si x est rationnel c'est évident. Sinon $x \prec y \Longrightarrow s(x) \subsetneq s(y)$. Soit $b \in s(y) - s(x)$. Comme s(y) est dirigé $\exists\, c \in s(y) \quad c \supseteq b, x$

$$s(c) = (\downarrow c) \cap s(y), \; x \in \downarrow c \text{ et } x \in s(y)$$

$\Longrightarrow \; x \in s(c)$ i.e. $x \prec c \prec y$ i.e. X interpolable au point x. □

<u>DEFINITION</u> : <u>Points de continuité d'un faisceau</u>

Soit $X \underset{s}{\overset{\sqcup}{\leftrightarrows}} \mathcal{P}(X)$ un fom, tel que X et $\mathcal{P}(X)$ soient munis d'une structure d'espace topologiques. Un point $x \in X$ sera dit point de continuité du fom ssi la fonction s est continue au point x.

Si tous les points de X sont des points de continuité, on dira que le fom est topologique pour les topologies de X et $\mathcal{P}(X)$.

<u>Exemple</u> : (1) Supposons X et $\mathcal{P}(X)$ munis de la topologie de Scott, alors $x \in X$ est un point de continuité de $X \underset{s}{\overset{\sqcup}{\leftrightarrows}} \mathcal{P}(X)$ ssi $\forall$ chaîne ascendante $\{x_i\}_{i\ I}$ telle que $x = \bigsqcup_i x_i$ on a $s(\bigsqcup_i x_i) = \bigvee_i s(x_i)$

(2) exhibons maintenant une fonction spectrale partout continue. Soit X un poset muni de la topologie de Scott, posons

$$s_t : X \longrightarrow \mathcal{P}(X)$$
$$x \longmapsto \{y : x \in \mathrm{Int}(\uparrow y)\} = \{y : y \ll_t x\}$$

(c'est la fonction spectrale de la relation "way-below" topologique).

s_t est une section commençante :

$$u \in s_t(x) \Longleftrightarrow x \in \mathrm{Int}(\uparrow u)$$

$$v \subseteq u \Longrightarrow \uparrow u \subseteq \uparrow v \Longrightarrow \mathrm{Int}(\uparrow u) \subseteq \mathrm{Int}(\uparrow v)$$

$$\Longrightarrow x \in \mathrm{Int}(\uparrow u) \subseteq \mathrm{Int}(\uparrow v) \Longrightarrow v \in s_t(x)$$

$\Longrightarrow s_t(x)$ est une section commençante.

Et s_t est partout continue i.e.

$$s(\bigvee_i x_i) = \bigvee_i s(x_i) \text{ car}$$

(i) $a \in s_t(\bigvee_i x_i) \Longleftrightarrow \bigvee_i x_i \in \mathrm{Int}(\uparrow a)$

$$\Longrightarrow \exists x_i \quad x_i \in \mathrm{Int}(\uparrow a) \Longrightarrow \exists x_i \quad a \in s_t(x_i)$$

$$\Longrightarrow s_t(\bigvee_i x_i) \subseteq \bigvee_i s_t(x_i)$$

(ii) s_t est monotone : immédiat

donc $\bigvee_i s_t(x_i) \subseteq s_t(\bigvee_i x_i)$ i.e.

$$s_t(\bigvee_i x_i) = \bigvee_i s_t(x_i)$$

De plus, puisque $s_t(x)$ est une section commençante, si s_t définit un faisceau alors c'est une connection de Galois. □

On peut donc dire dans le cas général, sur un poset X, en vue de l'application du lemme 2.17 pour obtenir l'interpolabilité, le fait que les spectres soient dirigés est ce qui "manque" aux relations $\ll_t$ (way-below topologique) et $\ll_i$ (way-below) pour être interpolables.

2.18. PROPOSITION : Soit $X \underset{s}{\overset{\cup}{\rightleftarrows}} \mathcal{P}(X)$ un fom, X et $\mathcal{P}(X)$ munis de la topologie de Scott

(i) si $\forall x \quad s(x) = s_{\ll_i}(x) = \{y \in X : y \ll_i x\}$
et si le fom est interpolable, alors chaque point de X est un point de continuité

(ii) si tous les points de X sont des points de continuité du fom, alors

$$\forall x \in X \quad s(x) \subseteq s_{\ll_i}(x) = \{y \in X : y \ll_i x\}$$

<u>DEMONSTRATION</u> :

(1) on a que (interpolabilité et prop. 2.9 clauses (i) et (ii)) $\Longrightarrow$

$$\forall x \in X \quad s_{\ll_i}(x) = s_{\ll_t}(x) = s_t(x)$$

et on voit dans l'exemple précédent que s_t est continue.

(2) Soit pour toute chaîne ascendante convergente de limite x $s(\bigsqcup_i x_i) = \bigcup_i s(x_i)$. Alors

$$v < x = \bigsqcup_i x_i \Longrightarrow v \in s(x) = s(\bigsqcup_i x_i) = \bigcup_i s(x_i)$$

$\Longrightarrow \exists x_i \; v \in s(x_i)$ i.e. $v < x_i$. Donc $v \in \bigcap\{I \in Id_{cv}X : x \subseteq \bigcup I$ car les idéaux ne sont rien d'autre que les sections commençantes engendrées par les parties dirigées et s est monotone.

D'où $v \ll_i x$. Donc

$$\forall x \in X \quad s(x) \subseteq \{y \in X : y \ll_i x\} \qquad \square$$

On peut noter aussi que si X, $\mathcal{P}(X)$ sont munis de la topologie de Scott, $s : X \longrightarrow \mathcal{P}(X)$ continue en x alors s(x) est une réunion de chaînes ascendants convergeant vers x ssi s(x) est dirigée car :

$\Longrightarrow$: si $a,b \in s(x)$ on prend une chaîne a_i passant par a et une chaîne $\{b_j\}$ passant par b ; $a, b \in s(x) = \bigcup_i s(a_i) = \bigcup_j s(b_j) \Rightarrow$

$$\exists b_j \quad a \in s(b_j) \text{ d'où } \max(b,b_j) \supseteq a,b$$

$\Longleftarrow$: trivialement.

<u>DEFINITION</u> : On dira qu'un fom sur X est continu au sens de Scott (ou simplement <u>continu</u>) ssi sa fonction spectre est

$$s : x \longmapsto s(x) = \{y \in X : y \ll_i x\}$$

On appellera <u>faisceaux algébriques</u> les faisceaux continus élémentaires.

La proposition 2.18 nous dit donc que parmi les fom interpolables tels que tous les points de l'espace sont des points de continuité pour les topologies de Scott de X et P(X), les faisceaux continus (au sens de Scott) interpolables sont les moins fins. Et que l'existence d'un faisceau s comme dans 2.18 (ii) force l'existence d'un faisceau continu au sens de Scott.

Ou encore, que X et P(X) étant munis de la topologie de Scott, la classe des faisceaux interpolables sur X, qui sont topologiques forme un sous-semi treillis complet du semi-treillis du lemme 2.11, et le bottom de ce sous-semi treillis est le fom continu de X.

On retrouvera une situation analogue à ce dernier point au chapitre III. Fait 3.11.

<u>2.19. COROLLAIRE</u> : Soit X un treillis complet et P(X) l'ensemble de ses parties ordonné par inclusion munis de leurs topologies de Scott, et $\lambda = \bigcup$. Alors $\exists$ un faisceau ordonné monique $X \underset{s}{\overset{\lambda}{\leftrightarrows}} P(X)$ qui est un faisceau topologique ssi X est un treillis continu.

<u>DEMONSTRATION</u> : immédiat par prop. 2.15 (v) et prop. 2.18.

(Nota : Notons que sur un faisceau continu toute fonction régulière en un point est Scott - continue en ce point. En effet, localement, pour toute partie dirigée S, $x = \bigcup S$, le fom défini au point x par $x \longmapsto {\downarrow} S$ est moins fin que le fom continu en x. Donc la régularité pour le second implique la régularité pour le premier (lemme 2.9)).

Il serait intéressant de savoir quelle est la structure des faisceaux topologiques lorsque l'on change la topologie de X et P(X). On pourrait par exemple prendre la topologie de Lawson (que nous retrouverons plus loin dans le problème de la séquentialité).

<u>DEFINITION</u> : Soit X un poset

(i) la <u>topologie inférieure</u> de X est engendrée par les compléments $X \setminus {\downarrow} a$, $a \in X$, des sections commençantes principales

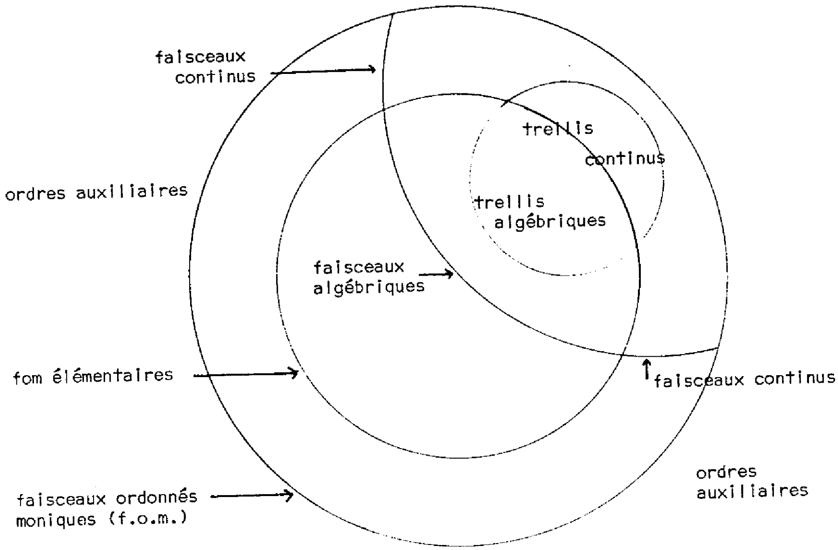
(ii) la <u>topologie de Lawson</u> de X est la topologie engendrée par les ouverts de la topologie de Scott et les ouverts de la topologie inférieure.

La classe des treillis continus est close par passage à l'espace des fonctions continues (régulières). Peut-on dire la même chose de la classe des treillis "continus au sens de Lawson" si on définit :

Soit X un treillis complet. X est continu au sens de Lawson ssi X et P(X) étant munis de la topologie de Lawson, il existe un faisceau topologique $X \underset{s}{\overset{\cup}{\leftrightarrows}} P(X)$.

Cette question, et les propriétés des faisceaux topologiques en général, n'ont pas encore été investiguées.

Le diagramme suivant montre quelques relations entre les structures considérées ici



Terminologie : On appellera faisceau algébrique tout fom continu au sens de Scott, qui est élémentaire. Si X est un fom, X et P(X) étant munis de la topologie de Scott, un point $x \in X$ est algébrique ssi (i) x est rationnel et (ii) x est un point de continuité du faisceau.

Exemples : (Toutes les structures considérées ici sont supposées munies de leur ordre habituel, sauf mention expresse) :

1. Un ordre auxiliaire qui n'est pas un faisceau

$$X = \mathbb{N}, \text{ soit } x \prec y \Longleftrightarrow (x \lneq y \text{ ou } x = y = 0).$$

La relation $\prec$ vérifie

$$x \prec y \Longrightarrow x \leq y$$

$$x \prec y \leq z \Longrightarrow x \prec z$$

$$\forall x \in X \quad \exists y \in X \quad y \prec x$$

c'est donc un ordre auxiliaire mais ce n'est pas un faisceau car :

$$\forall x \in X \quad x \neq 0 \Longrightarrow s_{\prec}(x) = \{y \in \mathbb{N} : y \lneq x\}$$

$$x = 0 \Longrightarrow s_{\prec}(0) = \{0\}$$

donc $\quad \forall x \in X \quad x = \sup(s_{\prec}(x)) \Longleftrightarrow x = 0.$

2. Un faisceau ordonné monique qui n'est pas élémentaire :

Soit $X = \mathbb{R}$ et $x \prec y \Longleftrightarrow x \lneq y$

est un ordre auxiliaire. De plus

$$\forall x \in X \quad x = \bigcup\{y \in \mathbb{R} : y \prec x\}$$

d'après les propriétés de $\mathbb{R}$. C'est donc un fom.

Cependant :

$$\forall v \in \mathbb{R} \quad v \prec v \Longleftrightarrow v \lneq v \Longleftrightarrow \text{FAUX}$$

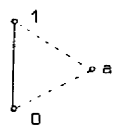
Donc l'ensemble des éléments $\prec$-rationnels est vide, et par conséquent il ne peut pas être dense dans $\mathbb{R}$. Le fom $\prec$ n'est donc pas élémentaire.

On note que ce faisceau est interpolable et stable.

3. Un fom élémentaire qui est un cpo, mais n'est pas algébrique :

Soit $X = [0,1] \cup \{a\}$ avec $a \notin \mathbb{R}$ muni de l'ordre partiel $\begin{cases} 0 \leq a \leq 1 \\ \text{ordre usuel sur } [0,1] \subseteq \mathbb{R} \end{cases}$

i.e. X est le treillis complet :



Munissons-le de la fonction de densité $\varphi = \bigsqcup\big|_{P(X)-\{1\}}$. Alors l'ensemble des éléments rationnels est :

$$Rt(X) = [0,1[\; \cup \; \{a\}$$

et 1 est le seul élément non rationnel. Nous avons évidemment un ordre élémentaire puisque

$$1 = \sup \{x : x \in [0,1[\}$$

et on a même un fom. Mais ce fom n'est pas algébrique car $[0,1[$ forme une chaîne ascendante dont la limite est 1 et

$$\bigcup\{s(y) : y \in [0,1[\} = [0,1[\; \neq s(1)$$

puisque le spectre s(1) de 1 est

$$s(1) = [0,1[\; \cup \; \{a\}$$

Les spectres ici ne sont pas dirigés.

4. Un faisceau algébrique qui n'est pas un cpo :

(i) l'ensemble $\mathbb{N}$ des entiers naturels muni de l'ordre usuel $\leqslant$ forme un faisceau continu dont tous les éléments sont rationnels. Mais ce n'est pas un cpo, puisqu'on a des chaînes ascendantes infinies qui divergent. (L'ordre $p \sqsubseteq q \Leftrightarrow p$ divise q donne une structure de même nature).

(ii) Si E est un ensemble infini, alors l'ensemble des parties propres non vides $X = P(E) - \{E,\emptyset\}$ ordonné par inclusion, muni du foe canonique satisfait aussi nos conditions.

5. Un cpo algébrique qui n'est pas un treillis algébrique : Soit $M^{\infty}(F,V)$ le magma complété [NIV]. Il est bien connu que c'est un cpo algébrique. La structure de faisceau algébrique est donnée par :

- éléments rationnels = les arbres finis
- éléments irrationnels = les arbres infinis.

6. Un faisceau non algébrique, qui n'est pas un cpo :

6.a. Un exemple avec quelques points de continuité : Nous considérons de nouveau le foe $<\xi,\varphi>$ sur $\mathbb{R}$, avec $\varphi = \bigcup \Big| \{\mathcal{U} \leq Q : \mathcal{U}$ majorée mais non bornée dans $Q\}$ que nous avons déjà donné en exemple. Les nombres rationnels sont (par vacuité) les seuls éléments φ-rationnels. Donc

$$\forall x \in \mathbb{R} \quad s(x) = \{y \in Q : y \leq x\}$$

Ce faisceau est clairement continu en tout point irrationnel, et discontinu en tout point rationnel, puisque 1 = 0,9999 .... mais

$$1 \notin \bigcup_i \{y \in Q : y \leq x_i\} \text{ où } x_i$$

est un rationnel de la forme 0,999...9.

Plus précisément soit $x \in \mathbb{R}$, $\{x_i\}_{i \in I}$ une chaîne ascendante telle que $x = \bigcup_i x_i$. Avons-nous $s(x) = \bigcup_i s(x_i)$? Il est clair que $s(x) \supset \bigcup_i s(x_i)$. Supposons $s(x) \neq \bigcup_i s(x_i)$; nous obtenons :

$$\exists u \in Q \quad u \in s(x) \text{ et } u \notin \bigcup_i s(x_i) \ \forall i$$

$\Longrightarrow \exists u \in Q \quad u > x_i \ \forall i$ et $u \leq x$.

Puisque $x = \bigcup_i x_i$, il y a deux possibilités :

(i) soit $u = x$ et ceci est impossible seulement dans le cas où x est un nombre irrationnel

(ii) soit $u \neq x$ et ceci conduit à une contradiction puisque x est la borne supérieure des x_i.

D'où la continuité du faisceau en tout point irrationnel et sa discontinuité en tout point rationnel. Aucun point de ce faisceau n'est algébrique.

6b. Un fom (élémentaire) partout discontinu :

Soit $X = \mathbb{Q}$ muni du fom induit sur Q par celui défini en 6a.

7. Un cpo qui est un faisceau continu, mais n'est ni élémentaire ni un treillis continu :

On considère les cpl (continuons partial lattices) définis par Egli, 1975 ainsi :

Un espace partiellement ordonné X est un cpl ssi

(i) X est semi-treillis complet

(ii) toute partie d'une partie dirigée à une borne supérieure

(iii) $\forall x \in X$ $\downarrow x$ est un treillis continu.

On voit que selon notre terminologie, les cpl sont des cpos complets sous condition munis d'un faisceau continu.

8. Un faisceau continu qui n'est pas un cpo, ni un cpl :

Il suffit de considérer la droite numérique $\mathbb{R}$.

CHAPITRE III

ESPACES DE FONCTIONS REGULIERES

°°°

Soient $<\lambda,\beta>$ et $<\lambda',\beta'>$ des faisceaux ordonnés sur X et Y (en fait il suffit que Y soit muni d'une fonction limite $\lambda' \in \{\sqcup,\sqcap\}$), $f : X \to Y$ une fonction (éventuellement partielle). Alors f est régulière au point $x \in X$ (resp. faiblement régulière au point $x \in X$) ssi $\forall \sigma \in \beta(x) \quad f(x) = \lambda'(f(\sigma))$ (resp. ssi $\forall \sigma \in \beta(x) \quad \lambda'(f(\sigma)$ existe $\Rightarrow f(x) = \lambda'(f(\sigma))$).

N.B. : à partir de maintenant, pour simplifier l'expression, nous confondrons souvent, s'il n'y a pas ambigüité, le nom de la structure de faisceau et le nom du support de cette structure.

DEFINITION : Ordre extensionnel (ou point par point) sur $[X \to Y]$:

$$\forall f,g \in [X \to Y] \quad f \leq g \Leftrightarrow \forall x \quad f(x) \leq g(x) \qquad \square$$

Exemples : (i) (faisceau S). Soit X un cpo discret i.e. un semilattice complet tel que $x \leq y \Leftrightarrow x = y$ ou $x = \perp$. On peut le munir d'une structure de f.o.e. canonique $<\leq,\sqcup>$ et tous les points sont rationnels.

$$\forall X \in s(x) = \{x,\perp\} \quad \text{si} \quad x \neq \perp$$
$$s(\perp) = \{\perp\}$$

(cet o.e. est en fait le faisceau S de X). Alors une fonction $f : X \to Y$, Y ordonné est continue pour la structure de cpo si et seulement si elle est régulière pour la structure de faisceau. Ceci est un cas important en theorie de la programmation mais est encore vrai pour tous les faisceaux S (cf. Chap. I pp. 49, 50). Ainsi si par exemple X est un cpo algébrique, Y ordonné, $f : X \to Y$ est continue au sens de la topologie de Scott ssi elle est régulière au sens du faisceau S de X ssi $\forall x \in X$ $f(x) = \sqcup\{f(a) : a \text{ fini} \subseteq x\}$ ssi elle est régulière au sens du

faisceau S de X ssi $\forall x \in X \quad f(x) = \bigcup\{f(a) : a \text{ fini} \sqsubseteq x\}$
ssi elle est régulière au sens du faisceau $S_{\ll_i}$ de X
$(X \underset{s_{\ll_i}}{\overset{\cup}{\rightleftarrows}} (X))$. (Ces équivalences sont faciles à démontrer). □

(ii) (faisceau P_ω)

Soit $X = Y = P\omega$ l'ensemble de toutes les parties de $\mathbb{N}$. On définit $f : P\omega \to P\omega$ continue ssi $\forall x \in P\omega$

$$f(x) = \cup\{f(e_n) : e_n \subseteq x\}$$

où les e_n, $n \in \mathbb{N}$ constituent une énumération des parties finies de $\mathbb{N}$ (on prend

$$e_n = \{k_0, k_1, \dots, k_{m-1}\}, \quad k_i < k_{i+1}, \quad n = \sum_{i<m} 2^{k_i})$$

Mais on a vu (Chap. I) que le faisceau $P\omega$ avait pour rationnels les singletons $\{p\}$, $p \in \mathbb{N}$ et $\emptyset$. La continuité de f peut donc se réécrire

$$f(x) = \cup\{f(e_n) : e_n \subseteq x\} = \cup\{f(\{p\}) : \{p\} \subseteq x\}$$

ce qui est exactement la régularité de f.

(iii) Si X est un foe avec $\lambda = \sqcup$ (resp. $\lambda = \sqcap$), X^2 le produit de X par lui-même, alors la fonction $(x,y) \mapsto x \sqcup y$ (resp. $(x,y) \mapsto x \sqcap y$) de X^2 dans X est régulière chaque fois qu'elle est définie (trivialement). □

I - <u>MONOTONICITE ET PROLONGEMENT</u>

On a vu dans les cas ci-dessus examinés que la notion de fonction régulière englobe celle de fonction continue au sens de Scott. On sait que les fonctions continues sont toutes monotones. Qu'en est-il des fonctions régulières ?

3.1. <u>LEMME</u> (monotonie des fonctions régulières)

Soient X,Y deux faisceaux ordonnés tels que X est normalisé et sa fonction spectre définit un fom. Soit $f : X \to Y$ une fonction.
Alors $\forall x,y \in X$, si f est régulière en x et y et $x \sqsubseteq y$, alors $f(x) \sqsubseteq f(y)$. En particulier si X est un fom, alors toute fonction régulière sur X est monotone sur X.

<u>DEMONSTRATION</u> : $x \sqsubseteq y$ et X est normalisé $\Rightarrow s(x) \in \beta(x)$ et $s(y) \in \beta(y)$; $s(x) \sqsubseteq s(y) \Rightarrow f(x) = \sqcup f(s(x)) \sqsubseteq \sqcup f(s(y)) = f(y) \Rightarrow$
$f(x) \sqsubseteq f(y)$. □

On note que si X et Y sont deux faisceaux ordonnés, $[X \to Y]$ l'ensemble des fonctions régulières muni de l'ordre extensionnel, et si $\{f_i\}_{i \in I}$ est une famille non vide de fonctions régulières, et si Y est un sup-semi-lattice complet alors $\{f_i\}_{i \in I}$ a une borne supérieure dans $[X \to Y]$ définie point par point par

$$(\bigsqcup_i f_i)(x) = \bigsqcup_i (f_i(x))$$

d'après le lemme 1.2.

3.2. <u>LEMME</u> : Soient X, Y deux f.o.e. tels que X est normalisé, ne contient que des rationnels et sa fonction spectre définit un fom ; soit $f : X \to Y$ une fonction. Alors f est régulière sur X ssi elle est monotone sur X.

<u>DEMONSTRATION</u> : d'après 3.1 f régulière $\Rightarrow f$ monotone.

Réciproquement f monotone $\Rightarrow \forall x \in X \quad f(x) = \sqcup f(s(x))$
puisque $x \in s(x) = \downarrow x$. □

3.3. <u>COROLLAIRE</u> : dans les conditions du lemme 3.2., pour toute fonction monotone $g : X \to Y$, la restriction de g aux éléments rationnels de X, $g|_{Rt(x)}$ est régulière.

Il est généralement faux que toute fonction régulière sur les éléments rationnels peut être prolongée en une fonction régulière sur l'espace tout entier. Par exemple soit $X = \mathbb{N}_T = \{T\} \cup \mathbb{N}$, avec $n \subseteq T$, muni de l'o.e. canonique. On a $Rt(X) = \mathbb{N}$.

Soit

$$Y = \begin{tikzcd} & \underline{\text{boolean}} \arrow[dl, dash] \arrow[dr, dash] & \\ \underline{\text{true}} & & \underline{\text{false}} \end{tikzcd}$$

muni de l'o.e. canonique; la fonction

$$f : Rt(X) \to Y \quad \text{tq} \quad \begin{array}{l} f(1) = f(2) = \text{true} \\ f(2+p) = \text{false} \quad \forall\, p > 0 \end{array}$$

est monotone sur $Rt(X)$, donc régulière, mais elle n'a pas de prolongement régulier à tout l'espace si le faisceau n'est pas monique puisque

$$\{1,2\}\ ,\ \{3,4,5,\ldots\} \in \beta(T) \Rightarrow$$

$$\overline{f}(T) = \begin{array}{l} \sqcup f(\{1,2\}) = \text{true} \\ \sqcup f(\{2+p : p > 0\}) = \text{false} \end{array}$$

avec true $\neq$ false est impossible.

Remarque : Dans le cas général, les relations entre la monotonie et la régularité ne sont pas très claires. Regardons par exemple le cas de la KP-convergence (chap I, pp 11-15).

Notations : Soient X, Y deux T_2-espaces topologiques, 2^X (resp. 2^Y) l'ensemble des fermés de X, $2^X \subseteq \mathcal{P}(X)$ (resp. de Y, $2^Y \subseteq \mathcal{P}(Y)$). Munissons 2^X et 2^Y de leurs faisceaux de la KP-convergence. Alors on sait (cf. [KUR], vol. 1. pp. 339 ; VI, 7 et 8) que pour toute suite $\{A_n\}_{n \in \mathbb{N}}$ de 2^X ou 2^Y, on a :

$$\{A_n\}_{n \in \mathbb{N}} \text{ croissante} \Rightarrow \lim_n A_n = \overline{\bigsqcup_n A_n}$$

$$\{A_n\}_{n \in \mathbb{N}} \text{ décroissante} \Rightarrow \lim_n A_n = \bigcap_n A_n$$

On dira qu'une fonction $f : 2^X \to 2^Y$ régulière pour les faisceaux de la KP-convergence de 2^X et 2^Y, est KP-régulière.

LEMME : (i) Si $f : 2^X \to 2^Y$ est à la fois KP-régulière et monotone croissante, alors elle est continue au sens de la topologie supérieure de Scott (i.e. du faisceau S de 2^X, 2^Y)

(ii) même assertion que (i) pour la topologie inférieure de Scott.

(iii) Si $g : X \to Y$ est une fonction fermée, et si son extension $g^* : 2^X \to 2^Y$ est KP-régulière $(g^*(A) = \{g(a) : a \in A\})$, alors g^* est continue au sens des deux topologies de Scott. □

<u>DEMONSTRATION</u> : (i) (ii) Soit $\{A_n\}_{n \in \mathbb{N}}$ une chaîne si $A_n \nearrow$ on a :

$$f(\lim_n A_n) = f(\overline{\bigcup_n A_n}) = (\text{reg.}) \lim_n f(A_n) = \overline{\bigcup_n f(A_n)}$$

si A_n décroit :

$$f(\lim_n A_n) = f(\bigcap_n A_n) = (\text{reg.}) \lim_n f(A_n) = \bigcap_n f(A_n)$$

(iii) la fonction g^* est monotone croissante de 2^X dans 2^Y. □

Sur les fom nous avons une propriété assez "lisse".

<u>3.4. LEMME</u> : Soit X un fom élémentaire, Y un sup-demitreillis complet. Alors toute fonction monotone $f : Rt(X) \to Y$ admet un unique prolongement régulier $\overline{f} : X \to Y$. Si Y n'est pas un sup-demitreillis complet, alors on a un unique prolongement faiblement régulier.

<u>DEMONSTRATION</u> : prendre

$$\forall x \in X \qquad \overline{f}(x) = \bigcup f(s(x))$$ □

Il est aisé de voir que ceci est une propriété d'universalité tout à fait classique en algèbre : on construit à partir de $Rt(X)$, un objet (X) qui est unique au sens suivant :

1/ on peut injecter $Rt(X)$ dans X

2/ pour tout objet Y et tout morphisme $f : Rt(X) \to Y$, il existe un unique morphisme $\overline{f} : X \to Y$ qui fait commuter le diagramme

$$\begin{tikzcd} Rt(X) \arrow[r, "i"] \arrow[d, "f"'] & X \arrow[dl, "\overline{f}"] \\ Y & \end{tikzcd}$$

Cette propriété se prolonge au cas où X est un fom et $f : N(X) \to Y$ ssi

$$\forall x \in N(X) \qquad \overline{f}(x) = f(x) \iff$$

$$\forall x \in N(X) \qquad f(x) = \sqcup f(s(x)) \iff f \text{ régulière}$$

sur $N(X)$. Donc pour appliquer 3.4 on est ramené à f régulière sur $N(X) \iff f$ monotone sur $N(X)$. Ceci permet d'énoncer

<u>3.5. PROPOSITION</u> : Si X est un fom, Y un espace ordonné, $N(X)$ le noyau de X. Alors toute fonction $f : N(X) \to Y$ régulière sur le noyau $N(X)$ telle que $\forall x \in X$ $\sqcup f(s(x))$ existe dans Y, admet un unique prolongement régulier $\overline{f} : X \to Y$. □

La restriction $f|_{N(X)}$ d'une fonction régulière $f : X \to Y$ étant régulière, il s'ensuit que dans les conditions de la proposition 3.5., il y a une bijection entre les fonctions régulières sur X et celles régulières sur $N(X)$:

$$[X \to Y] \simeq [N(X) \to Y]$$

<u>3.6. COROLLAIRE</u> : Si X est un fom élémentaire et Y un treillis complet, alors toute famille $\{f_i\}_{i \in I} \subseteq [X \to Y]$ a une borne inférieure dans $[X \to Y]$ définie par $(\sqcap_i f_i)(x) = \sqcap_i f_i(x)$ si x est rationnel. □

<u>DEMONSTRATION</u> : appliquer 3.2.

<u>II. COMPOSITION DE FONCTIONS REGULIERES</u> :

Nous avons vu plus haut (Chap. I. pp 37) que, en général les fonctions régulières ne sont pas closes par composition.

<u>3.7. PROPOSITION</u> : Soient X, Y, Z des faisceaux ordonnés tels que toute partie majorée de Z admet une borne supérieure, $f : X \to Y$ et $g : Y \to Z$

des fonctions régulières monotones croissantes. Alors la fonction composée $(g \circ f) : X \to Z$ définie par $(g \circ f)(x) = g(f(x))$ est régulière au point $x \in X$ dès que la condition suffisante :
"les éléments du noyau $N(Y)$ de Y sont finis pour l'image par f de la fibre de x" est réalisée

<u>DEMONSTRATION</u> : Nous avons $\forall \sigma \in \rho(x)$

$g(f(x)) = \sqcup \{ g(b) : b \in s(f(x))\} \sqsubseteq$ (puisque par hypothèse tout $b \in s(f(x))$ est $\sqsubseteq f(a_b)$ où $a_b \in \sigma$)

$\sqsubseteq \sqcup\{g(f(a_b)) : a_b \in \sigma \ , \ b \in s(f(x))\} \sqsubseteq \sqcup\{g(f(u)) : u \in \sigma\}$

Maintenant par monotonicité

$$\forall u \in \sigma \qquad g(f(u)) \sqsubseteq g(f(x))$$

i.e. $\sqcup\{g(f(u)) : u \in \sigma\} \sqsubseteq g(f(x))$

d'où l'égalité

$$g(f(x)) = \sqcup \{g(f(u)) : u \in \sigma\}$$

i.e. la régularité de $g \circ f$ en x. □

Cette proposition éclaire le fonctionnement de la continuité au sens de Scott pour la composition de fonctions définies sur des treillis continus ou des cpo's algébriques : l'image d'une partie dirigée est toujours une partie dirigée, et les éléments approximants sont tous finis pour les parties dirigées.

Le phénomène constaté par Prop. 3.7 n'est pas pour surprendre car on se rend compte ici que la notion de fonction $f : X \to Y$, où X et Y sont des ensembles, avec la composition usuelle, est inadéquate parce que trop générale. Au mieux (et si l'on n'examine pas les choses plus précisèment), elle fait déborder tout de suite dans la classe des fonctions monotones (cf. Shamir et Wadge, 1977). Elle ne tient pas compte du fait que l'on travaille ici sur des espaces très particuliers, les faisceaux, et le mode de calcul sur les fonctions doit nécessairement se placer dans le cadre de la théorie des faisceaux. Un faisceau sur un ensemble revient à se donner pour chaque point de l'ensemble un système d'approximation, qui permet d'accéder à ce point. Donc transformer un point x par une fonction f, revient à transformer par f le système d'approximation $\beta(x)$ de x par f, et si l'on doit encore transformer la valeur ainsi obtenue par une deuxième fonction g, alors puisqu'à chaque instant on ne peut manipuler que des approximations, il faut transformer par g l'image par f du système d'approximation $\beta(x)$ de x. Dans le cas où $\beta(x) = \{s(x)\}$, on a

$$x \overset{s}{\longmapsto} s(x) \overset{f}{\longmapsto} f(s(x)) \overset{g}{\longmapsto} g(f(s(x)))$$

Ceci revient à tenir compte de la proposition 3.5. ci-dessus. L'idée ainsi mise en oeuvre n'est rien d'autre que la vieille idée qui consiste à considérer des fonctions $\mathbb{R} \to Y$ définies comme des prolongements continus de fonctions continues $\mathbb{Q} \to Y$. Ces fonctions sont les seules qui permettent d'avoir des valeurs en des points irrationnels au moyen de valeurs en des points rationnels (finis au sens de la terminologie des programmes).

Une première formalisation de ceci est de ne s'occuper que de l'espace initial et de l'espace final : c'est la méthode "pipe-line". Si $X_1, \dots, X_{n-1}$ sont des fom, et Y un faisceau ordonné,

$$X_1 \xrightarrow{f_1} X_2 \xrightarrow{f_2} \cdots \to X_{n-1} \xrightarrow{f_n} Y$$

et si le seul but est d'obtenir une composition des f_i régulière sur X_1, en évitant les problèmes d'associativité, on peut prendre la <u>composition "pipe-line"</u> suivante

$$\forall x \in X_1$$

$$(\square(f_1, \dots, f_n))(x) = \curlyvee_{\lambda(Y)} \{(\square(f_2, \dots, f_n))(u) \ : \ u \in s(f_1(x))\}$$

(définie par récurrence), en ne transformant à chaque instant que des éléments du noyau et en ne prenant la limite qu'à la fin du processus, et en prenant X_1 monique.

Une deuxième formalisation de ceci oblige à s'occuper davantage de la nature des objets que l'on suppose.

Dans notre formalisme ceci s'écrit ainsi :

$$o : [X \to Y] \times [Y \to Z] \to (X \to Z)$$

(où $(X \to Z)$ désigne pour le moment l'espace Z^X de toutes les fonctions de X dans Z, ou encore, mettant des faisceaux sur $[X \to Y]$ et $[Y \to Z]$:

DEFINITION : Composition de fonctions au sens des faisceaux :
Soient X, Y, Z, $[X \to Y]$, $[Y \to Z]$ des faisceaux ordonnés. Alors on définit

$$o : N[X \to Y] \times N[Y \to Z] \times N(X) \to Z$$

$$o = \lambda f \in N[X \to Y] . \lambda g \in N[Y \to Z] . \lambda u \in N(X) . g(f(u)) \qquad \square$$

3.8. PROPOSITION :

(i) Si tous les espaces considérés sont munis du foe trivial, alors
$\forall f \in [X \to Y]$, $g \in [Y \to Z]$, $x \in X$
$o(f,g)(x) = g(f(x)) = (g \circ f)(x)$

(ii) si tous les espaces considérés sont munis du faisceau S , alors
$\forall f \in [X \to Y]$, $g \in [Y \to Z]$, $x \in X$
$o(f,g)(x) = g(f(x)) = (g \circ f)(x)$

(iii) si X, Y, Z sont des cpo's algébriques et si $N[X \to Y] = [X \to Y]$, $N[Y \to Z] = [Y \to Z]$ (i.e. f.o.e. triviaux), alors
$\forall f \in [X \to Y]$, $g \in [Y \to Z]$, $x \in X$
$o(f,g)(x) = g(f(x)) = (g \circ f)(x)$

(iv) si l'espace considéré est le faisceau N sur un modèle ensembliste d'une collection d'algorithmes $\mathcal{A}$, alors $\forall X, Y, Z \in \mathcal{A}$, $\forall f \in [X \to Y]$, $\forall g \in [Y \to Z]$, $x \in X$,
$o(f,g)(x) = \mathrm{Puis}[f,g,x]$
où Puis est défini par Nolin [NOL] comme

$$\mathrm{Puis} = \bigcap \{ F_3\ V_1\ V_2\ V_3 (V_2 [V_1 [V_3]]) : V_1 , V_2 , V_3 \in \mathcal{A} \}$$

(v) si $[X \to Y]$ et $[Y \to Z]$ sont des faisceaux fidèles et élémentaires, si $[X \to Y]$ est un fom , alors la fonction o est régulière en tout point

(f,g,u) rationnel pour le faisceau produit tq f et g sont monotones.

(vi) Si X est un fom élémentaire, si Y est un fom, si $[X \to Y]$, $[Y \to Z]$ sont des fom élémentaires fidèles, si Z est un poset complet sous condition, alors l'application o définie sur le produit des noyaux a un unique prolongement régulier à l'espace $[X \to Y] \times [Y \to Z] \times X$ muni du fom produit. En particulier $\forall f \in [X \to Y]$, $g \in [Y \to Z]$ on a $o(f,g) \in [X \to Z]$ i.e. la composition o fournit à partir de fonctions régulières des fonctions régulières. □

<u>DEMONSTRATION</u> :

(i) affirme simplement que o généralise la composition usuelle ce qui est évident car $N[X \to Y] = [X \to Y]$, $N[Y \to Z] = [Y \to Z]$, $N(X) = X$

(ii) ici encore $N[X \to Y] = [X \to Y]$, $N[Y \to Z] = [Y \to Z]$, $N(X) = X$

(iii) $o(f,g)(x) = \sqcup\, (o(f,g)(s(x)) =$

$\sqcup\, (\lambda f \in N[X \to Y] \,.\, \lambda g \in N[Y \to Z] \,.\, \lambda u \in N(X).g(f(u)))\ (f,g,s(x)) =$

$\sqcup\, g(f(s(x))) = \sqcup\, (g \circ f)(s(x) = (g \circ f)(x)$

puisqu'on peut appliquer 3.5 et que la composition de deux fonctions continues est une fonction continue.

(iv) même cas que (ii) (tous les algorithmes propres sont atomiques)

(v) résulte de la régularité de l'application, lemme 1.6, et proposition 1.4

(vi) résulte de (v), proposition 3.5 et $\forall x\ f(g(x)) \sqsupseteq o(g,f,x)$ □

On est donc fondé de parler de l'application o comme étant la composition régulière. La composition usuelle, qu'elle généralise (3.8(i)), est associative (et ce fait est utilisé avec profit dans les catégories). Qu'en est-il de la composition régulière ? A-t-on

$$o(o(f_1,f_2),g) = o(f_1,o(f_2,g)) \ ?$$

<u>3.9. PROPOSITION</u> : Soit X, Y, Z, U des faisceaux ordonnés avec U poset complet sous condition, et munissons les espaces de fonctions régulières de structures de faisceaux. Soit les fonctions régulières monotones

$$X \xrightarrow{f} Y \xrightarrow{g} Z \xrightarrow{h} U$$

Alors :

(i) si f, g, h, $o(g,h)$, $o(f,g)$ sont respectivement des éléments du noyau de $[X \to Y]$, $[Y \to Z]$, $[Z \to U]$, $[Y \to U]$ et $[X \to Z]$, alors pour tout $x \in X$ pris dans le noyau de X, on a

$$o(f,o(g,h)(x) \leq o(o(f,g),h)(x)$$

(ii) si tous les faisceaux fonctionnels considérés sont des fom élémentaires fidèles, si X, Y, Z sont des fom, si X et Y sont élémentaires, alors, prenant le prolongement canonique de o, on a :

$$o(f,o(g,h)) \leq o(o(f,g),h)$$

(iii) dans les mêmes conditions que (ii), si de plus, pour tout $x \in X$ rationnel $f(x)$ est rationnel, alors

$$o(f,o(g,h)) = o(o(f,g),h)$$

□

<u>DEMONSTRATION</u> : (i) Posons $f = a$, $o(g,h) = b$. Alors

$$o(f,o(g,h)) = o(a,b) =$$

$(\lambda f \in N[X \to Y] \,.\, \lambda g \in N[Y \to Z] \,.\, \lambda u \in N(X).g(f(u)))(a,b)$

$=$ (a est dans le noyau de $[X \to Y]$, et b dans le noyau de $[Y \to Z]$)

$= \lambda u \in N(X).\ b(a(u))$

Soit x dans le noyau de X, alors $o(a,b)(x) = b(a(x)) =$ (définition de a et b) $= o(g,h)(f(x)) =$ ($o(g,h)$ est régulier)

$= \sqcup\{o(g,h)(\ell)\ \ \ell \in \mathrm{Im}(\sigma),\ \ \sigma \in \beta(f(x))\} =$ (g, h, ℓ éléments du noyau)

$\sqcup\{h(g(\ell)) : \ell \in \mathrm{Im}(\sigma),\ \ \sigma \in \beta(f(x))\} =$

($\ell \leq f(x)$ et monotonie de f,g,h) $\leq \sqcup\{h(g(f(x))) : \ell \in \mathrm{Im}(\sigma) \ldots\}$

$\leq h(g(f(x)))$.

Mais f, g, x éléments du noyau $\Rightarrow$ $g(f(x)) = o(f,g,x) = o(f,g)(x)$

donc :

$$h(g(f(x))) = h(o(f,g)(x)) = \text{(si } h,\ o(f,g), x \text{ sont des}$$

éléments du noyau) $= o(o(f,g),h,x) = o(o(f,g),h)(x)$.

Donc, dans les conditions précédente, $\forall x$ dans le noyau on a :

$$o(f,o(g,h)\ (x) \subseteq o(o(f,g),h)(x)$$

(ii) se déduit de Proposition 3.8. (vi)

(iii) maintenant si pour tout x rationnel $f(x)$ est rationnel, alors $f(x) \in s(f(x))$, donc

$$o(f,o(g,h))(x) = \cup \{o(g,h,u) : u \in s(f(x))\}$$
$$= h(g(f(x))) = o(o(f,g),h)\ (x).$$

D'où la proposition. □

On voit donc que la composition régulière n'est pas dans le cas général, associative. A notre connaissance, ce fait a été relevé pour la première fois par L. Nolin pour les collections d'algorithmes (Nolin 1974).

<u>DEFINITION</u> : Si X et Y sont des foe, on appellera <u>fonctions simples</u> les $f \in [X \to Y]$ tq $\forall x$ rationnel $f(x)$ est rationnel. □

Ceci généralise la notion <u>d'algorithme simple</u> de Nolin.

Il reste que dans tous les cas, la structure particulière du faisceau considéré intervient au niveau de la composition des fonctions régulieres, et le cas le plus commode est celui où la composition usuelle et la composition régulière coïncident.

Ainsi par exemple :

Si X, Y, Z sont des T_2-espaces topologiques, 2^X, 2^Y, 2^Z munis de faisceaux de la <u>KP-convergence</u> $f : 2^X \to 2^Y$, $g : 2^Y \to 2^Z$ deux fonctions KP-régulières. A-t-on $g \circ f : 2^X \to 2^Z$ KP-régulière ?

On a : $\forall$ suite $\{A_n\}_{n \in \mathbb{N}}$ de 2^X

$$(g \circ f)(\lim_n A_n) = \quad (f \ \text{KP-régulière})$$

$$g(\lim_n f(A_n)) = \quad (g \ \text{KP-régulière})$$

$$\lim_n (g(f(A_n))) = \lim_n (g \circ f)(A_n)$$

donc $g \circ f$ est KP-régulière.

Ceci vient du fait que toutes les suites de fermés sont convergentes au sens de Kuratowski-Painlevé, et qu'une fonction régulière envoie un système d'approximation sur un système d'approximation. Il n'y a donc pas lieu d'utiliser ici une notion speciale de composition régulière.

III - FAISCEAUX SUR LES ESPACES DE FONCTIONS REGULIERES :

Si X et Y sont des faisceaux, existe-t-il une (ou plusieurs) structure(s) de faisceau naturelle(s) sur $[X \to Y]$?

Dans le cas où X et Y sont des espaces topologiques (cf. Chap.I pp. I.43), on sait que l'on peut toujours définir sur $[X \to Y]$ la notion de convergence simple, i.e.

$$f = \lim_{i \ [X \to Y]} f_i \iff \forall x \in X \quad f(x) = \lim_{i \ Y} f_i(x)$$

ou encore, posant $f_i < f \iff f_i$ approxime f, :

$$f_i < f \iff \forall x \quad f_i(x) < f(x).$$

Dans le cas général de la classe de tous les faisceaux ordonnés ceci fournit une procédure, pas toujours effective, pour construire un faisceau sur $[X \to Y]$. Plus précisément : si X et Y sont des faisceaux ordonnés (tels que Y est normalisé), définir sur $[X \to Y]$ l'ordre auxiliaire

$$f < g \iff \forall x \in X \quad f(x) < g(x)$$

(posant $f(x) < g(x) \iff f(x) \in s(g(x))$).

Si $<$ est un fom, alors on a un faisceau sur $[X \to Y]$ que nous appellerons

<u>faisceau de la convergence simple</u>. Ce faisceau est fidèle par définition, et

$N([X \to Y]) = \{f \in [X \to Y] : \exists g \quad \forall x \quad f(x) < g(x)\}$.

Ainsi si X est un foe normalisé, dont la fonction spectre est un fom, Y un autre oe, la proposition 3.9 indique l'importance des fonctions $f \in [X \to X]$ tq $\forall x$ rationnel $f(x)$ est rationnel.
i.e. $f \in [X \to X] \cap [Rt(X) \to Rt(X)]$.
Ces fonctions suffisent dans certains cas pour construire un faisceau de la convergence simple.

En effet X, Y étant des foe, si un tel faisceau existe sur $[X \to Y]$ alors son noyau

$N([X \to Y])$ est :

$$N[X \to Y] = \{f \in [X \to Y] : \exists g \in [X \to Y] \text{ tq } \forall x \quad f(x) < g(x)\} \subseteq$$

$$\subseteq \{f \in [X \to Y] : f(X) \subseteq Rt(Y)\} = [X \to Rt(Y)]$$

Comme $[X \to Rt(Y)]$ est en bijection avec $[Rt(X) \to RT(Y)]$ dès que X est un fom (proposition 3.5), il vient alors dans ce cas-là

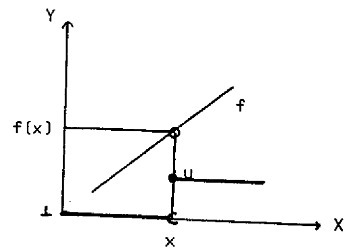
$$N[X \to Y] \subseteq [Rt(X) \to Rt(Y)] \quad .$$

Réciproquement si X est un foe normalisé dont la fonction spectre définit un fom, Y un oe, on peut définir $[X \to Y]$ la fonction

$$s : [X \to Y] \to \mathcal{P}([Rt(X) \to Rt(Y)])$$

$$s : f \to s(f) = \{g \in [Rt(X) \to Rt(Y)] : \begin{array}{l}\forall x \in X \\ g(x) < f(x)\end{array}\}$$

Si $\perp \in X$ alors $s(\perp) \neq \emptyset$ et $s(\perp) \subseteq \downarrow \perp \Rightarrow s(\perp) = \{\perp\}$ i.e. $\perp < \perp$.
Dans ce cas $s(f)$ défini ci-dessus contiendra toujours les fonctions en escalier $e_u : z \mapsto \begin{cases} u < f(x) & \text{si } x \sqsubseteq z \\ \perp & \text{sinon} \end{cases}$



et on vérifie que ceci suffit à approximer simplement f.

Cette procédure nous donne dans certains cas une structure de fom sur [X → Y]. Existe-t-il d'autres procédures qui marchent dans d'autres cas ?

Guidés par des préoccupations informatiques et l'importance des constructions de Scott, Nolin, Plotkin et Smyth pour les programmeurs, nous allons mettre en lumière les méthodes utilisées par ces auteurs pour aborder ce problème.

Dans cette section nous allons examiner plusieurs façons, non réductibles l'une à l'autre, de s'attaquer à la construction de faisceaux sur [X → Y] , utilisées (plus ou moins implicitement) dans les travaux de Scott [SCO] , Nolin [NOL], Egli [EGL] , Milner [MIL] , Plotkin [PLO] , et Smyth [SMY] . Le succès des trois premières a permis leur utilisation pour résoudre plusieurs questions d'informatique : modèle D_∞ de Scott, E_∞ de Wadworth, etc... Le demi-échec (ou demi-succès) des deux autres est un des thèmes de recherche le plus intéressant à l'heure actuelle : la modélisation du calcul parallèle. Nous pouvons les résumer ainsi :

<u>1ère méthode</u> : utilisant les échelons de Scott . X doit être un faisceau interpolable, Y un faisceau particulier [1] et les fonctions régulières

(1) voir énoncé plus loin

considérées sont minorées. Cette méthode fournit un faisceau fidèle et peut être itérée sous certaines conditions.

<u>2ème méthode</u> : utilisant la continuité de l'espace.
Cette méthode est liée à la précédente. X est un faisceau interpolable, Y est un faisceau continu au sens de Scott et possède un $\perp$; donne un faisceau continu sur $[X \to Y]$; ce faisceau est fidèle.

<u>3ème méthode</u> : utilisant les échelons de Nolin. X et Y sont des faisceaux ordonnés quelconques, et les fonctions considérées sont monotones majorées. Cette méthode fournit un fom fidèle et peut être itérée dans tous les cas.

<u>4ème méthode</u> : utilisant le préordre d'Egli-Milner, $X = N^{\omega}$, Y est un cpo; donne "presque" un faisceau sur $[X \to Y]$.

<u>5ème méthode</u> : utilisant le préordre de Smyth, analogue à la précédente.

Etablissons d'abord quelques faits préliminaires

<u>3.10. LEMME</u> : Soient A, D des fom. Supposons que $[A \to D]$ est muni du fom de la convergence simple

$$s(f) = \{g \to [A \to D] : \forall x \quad g(x) < f(x)\}$$

Alors

(i) si D est stable, $[A \to D]$ est stable,

(ii) si D est une connection de Galois, alors $[A \to D]$ est aussi une connection de Galois.

<u>DEMONSTRATION</u> :

(i) soit $h, f \in [A \to D]$, $h \geq f$. A-t-on $s(h) \cap (\downarrow f) = s(f)$?

$$s(h) \cap (\downarrow f) = \{g \in [A \to D] : \forall x \quad g(x) < h(x) \quad \text{et} \quad g(x) \subseteq f(x)\}$$

$$= \{g \in [A \to D] : \forall x \quad g(x) \in s(h(x)) \quad \text{et} \quad g(x) \in \downarrow(f(x))$$

$= \{g \in [A \to D] : \forall x \;\; g(x) \in s(h(x)) \cap (\downarrow f(x))\}$
$=$ (D stable) $\{g\ldots : \forall x \;\; g(x) \in s(f(x))\} \; = s(f)$

Donc $[A \to D]$ est stable.

(ii) A,D des fom, D connection de Galois $\Rightarrow$ $[A,D]$ muni de cv simple connection de Galois ?

i.e. $\forall f \in [A \to D] \quad s(f) = \downarrow_{N[A \to D]} f$

On a :

$$\downarrow_{N[A \to D]} f = \{g \in [A \to D] : g \subseteq f \text{ et } \exists h \; \forall x \;\; g(x) < h(x)\}$$

D connection de Galois $\Leftrightarrow \forall y \in D \quad s(y) = \downarrow_{N(D)} y$

$g(x) < h(x) \Leftrightarrow g(x) \in s(h(x)) \Leftrightarrow g(x) \subseteq h(x)$ et $g(x) \in N(D)$

donc

$$\downarrow_{N[A \to D]} f = \{g \in [A \to D] : g \subseteq f \text{ et } \exists h \;\; \forall x (g(x) \subseteq h(x) \text{ et } g(x) \in N(D))\}$$

$= \{g : \forall x \; g(x) \subseteq f(x)$ et $\exists h \;\; g \subseteq h$ et $\forall x \; g(x) \in N(D)\}$

$= \{g : \forall x (g(x) < f(x)$ et $\exists h \;\; g \subseteq h)\} =$
$= \{g : \forall x \;\; g(x) < f(x)\} \; = s(f)$

Donc $[A \to D]$ est une connection de Galois. □

3.11. <u>Fait</u> : Soient A,D des fom. Supposons qu'il existe sur $[A \to D]$ un fom fidèle s tel que $\forall f,g \in [A \to D] \;\; g < f \Rightarrow \forall x \;\; g(x) < f(x)$

Alors il existe sur $[A \to D]$ une structure de fom de la convergence simple.

DEMONSTRATION : Si s définit le fom fidèle on a :

$$\forall f \in [A \to D] \quad s \text{ fidèle } \Rightarrow s(f) \subseteq \{g : \forall x \;\; g(x) < f(x)\} \subseteq \downarrow f$$

$f = \sqcup s(f) \subseteq \sqcup\{g : \forall x \;\; g(x) < f(x)\} \subseteq \sqcup(\downarrow f) = f$

donc $<$ définie par $f < g \Leftrightarrow \forall x \;\; f(x) < g(x)$ définit un faisceau de la convergence simple. □

<u>DEFINITION</u> : Echelons :

Soient D, D' des faisceaux ordonnés, $e \in D$, $e' \in D'$. Nous définissons :

(i) si $m \leq e'$ <u>l'échelon de Scott</u> (ou inférieur)

$$[e,e']_m : x \mapsto \begin{cases} e' & \text{si } \forall \sigma \in \beta(x)\ \exists i \in \mathrm{Dom}(\sigma)\ e = \sigma(i) \\ m & \text{sinon} \end{cases}$$

(ii) si $m \geq e'$ <u>l'échelon de Nolin</u> (ou supérieur)

$$\langle e,e' \rangle_m : x \mapsto \begin{cases} e' & \text{si } x \subseteq e \\ m & \text{sinon.} \end{cases}$$

□

L'échelon $[e,e']_m$ est bornée inférieurement et l'échelon $\langle e,e' \rangle_m$ est borné supérieurement.

A notre connaissance, les échelons de Scott furent introduits pour la première fois par D. Scott 1969, avec $m = \perp$, dans la théorie des treillis continus, et les échelons de Nolin par L. Nolin 1974, avec $m = T$, dans la théorie des algorithmes. Nous notons aussi que pour des structures de faisceau plus connues, comme les réels par exemple, les fonctions en escalier sont d'un usage courant.

<u>Exemples</u> :

1/ sur l'oe $\langle \mathbb{N}, \mathrm{ppcm} \rangle$

$$[e,e']_1(x) = \begin{cases} e' & \text{si } e \text{ divise } x \text{ et } e \text{ premier} \\ 1 & \text{sinon} \end{cases}$$

$$\langle e,e' \rangle_M(x) = \begin{cases} e' & \text{si } x \text{ divise } e \\ M & \text{sinon} \end{cases}$$

2/ sur le faisceau $P\omega$

$$[e,e']_\emptyset(x) = \begin{cases} e' & \text{si } e \subseteq x \text{ et } e \text{ singleton} \\ \emptyset & \text{sinon} \end{cases}$$

$$\langle e,e' \rangle_M(x) = \begin{cases} e' & \text{si } x \subseteq e \\ M & \text{sinon} \end{cases}$$

On note que pour les échelons de Nolin

$$\langle e,e'\rangle_T \leq \langle d,d'\rangle_T \Leftrightarrow (d' = T) \quad \text{ou} \quad ((e' \in d') \wedge (d \subseteq e))$$

Nous reprendrons ce point dans notre discussion sur les rétractes à la fin de ce chapitre.

Comme les échelons sont intuitivement parmi les fonctions les plus simples qu'on puisse imaginer, une question que l'on peut naturellement se poser est : peut-on les utiliser pour reconstruire toutes les fonctions régulières ? (faiblement régulières ?).

III.1. ECHELONS DE SCOTT

DEFINITION : (i) $f \in [D \to D']$ possède un minorant sur D ssi $\exists\ m \in D' \quad \forall x \in D \quad m \leq f(x)$. Dans la suite on notera $\min(f)$ un tel minorant.

(ii) $f \to [D \to D']$ possède un minorant approximant sur D ssi $\exists\ m \in D' \ \forall x \in D \quad m < f(x)$. Dans la suite on notera $m(f)$ un tel minorant approximant.

3.12. LEMME : Si D et D' sont des fom, pour toute fonction régulière $f : D \to D'$ possédant un minorant $\min(f)$ sur D on a

$$f = \sqcup \{[e,e']_{\min(f)} : e' < f(e)\}$$

où la borne supérieure est prise dans l'ensemble des fonctions monotones de D dans D'.

DEMONSTRATION : Définissons

$f' = x \mapsto (\sqcup \{[e,e']_{\min(f)} : e' < f(e)\})(x)$

Alors

$f'(x) = \sqcup \{[e,e']_{\min(f)}(x) : e' < f(e)\}$

$\quad = (\sqcup \{e' : \exists\, e < x \quad e' < f(e)\}) \sqcup \min(f)$

$(e < x \Rightarrow e \subseteq x \quad e' < f(e) \subseteq f(x) \Rightarrow e' < f(x)$ par monotonie des spectres)

$= (\sqcup \{e' : e' < f(x)\}) \sqcup \min (f)$

$= f(x) \sqcup \min(f) = f(x).$ □

Remarque : On note dans la démonstration précédente que f est faiblement régulière ssi f' est faiblement régulière et qu'elles ont le même domaine de définition. Donc le lemme 3.12. reste valable pour les fonctions faiblement régulières.

Si $[D \to D']_{min}$ est l'ensemble des fonctions régulières $f \in [D \to D']$ minorées sur D, on a donc un fom sur $[D \to D']_{min}$:

$$s : f \to \{[e,e']_{\min(f)} : e' < f(e)\}$$

à condition que les $[e,e']_{\min(f)}$ soient des fonctions régulières minorées sur D. Comme elles sont minorées, la réponse est donnée par :

3.13. PROPOSITION : Soient D,D' des fom.
Alors la fonction échelon $[e,e']_m$ $(e \in D,\ e',m \in D',\ m \subsetneq e')$ est une fonction régulière si et seulement si le fom D est interpolable au point e.

DEMONSTRATION : $[e,e']_m$ est régulier au point $h \in D$ ssi

$$[e,e']_m(h) = \sqcup \{[e,e']_m(\ell)\ \ \ell < h\}$$
$$= \sqcup \{\text{si } e < \ell \text{ alors } e' \text{ sinon } m : \ell < h\}$$
$$= \begin{cases} e' & \text{si } \exists\, \ell < h \text{ tq } e < \ell < h \\ m & \text{sinon} \end{cases}$$

Comme $[e,e']_m(h) = \begin{cases} e' & \text{si } e < h \\ m & \text{sinon} \end{cases}$

nous devons avoir $(e' \neq m)$ $e < h$ ssi $\exists\, \ell < h$ $e < \ell < h$ ssi D interpolable au point e. D'où la proposition. □

Remarque : l'échelon $[e,m]_m$, qui est constant, est toujours régulier (Lemme 1.2)

3.14. COROLLAIRE : Si D et D' sont des fom, et si le fom D est interpolable, alors l'espace $[D \to D']_{min}$ des fonctions régulières de D dans D' minorées sur D, ordonné extensionnellement est tel que

(i) il est muni d'une structure de fom fidèle par la fonction spectre

$$s : f \mapsto \{[e,e']_{min(f)} : e' < f(e)\}$$

où min(f) décrit l'ensemble des minorants de f. Le noyau de l'espace est constitué par l'ensemble

$$\{[e,e']_a : e \in N(D),\ e' \in N(D')\ ,\ a \leq e'\}$$

(ii) si $\forall f \in [D \to D]_{min}\ \exists m(f)\ \ \forall x\ \ m(f) < f(x)$
alors l'espace peut être muni d'un fom de la convergence simple.

DEMONSTRATION :

(i) d'après proposition 3.13. lemme 3.12 et proposition 1.1.

(ii) d'après fait 3.11. □

On aurait pu se contenter dans 3.14 d'une structure de faisceau sur $[D \to D']_{min}$ la fonction spectre de (i) étant remplacée par la fibre :

$$\beta : f \mapsto \{\{[e,e']_{min(f)} : e' < f(e)\}\}$$

chaque minorant de f permettant de démarrer une "approximation", et donnant un élément de la fibre.

3.15. COROLLAIRE : le corollaire 3.14 reste valable si l'on remplace les fonctions régulières par les fonctions faiblement régulières. □

La question qui se pose maintenant est :
peut-on itérer ce processus et grimper ainsi dans l'échelle de fonctionnalité ? On sait que le pivot de ce processus est l'interpolabilité de D.

Pour quelle classe de fom l'interpolabilité est-t-elle héréditaire par passage à l'espace des fonctions régulières, si on se sert des échelons de Scott ? Le faisceau donné par Corollaire 3.14. (i) est élémentaire si D' est élémentaire, et ses rationnels sont exactement les échelons de Scott. Il est donc trivialement interpolable (Fait 2.9(i)).

Pour le faisceau de la convergence simple construit dans Cor. 3.14(ii), nous abordons ce problème par le biais de la stabilité (i.e. du Lemme 2.17) dont on sait qu'elle est héréditaire (lemme 3.10).

3.16. <u>PROPOSITION</u> : Soit D, D' des fom. Supposons D interpolable et que $\forall f \in [D \to D']_{min}$ possède un minorant approximant, i.e.

$$\exists m(f) \in D' \quad \forall x \quad m(f) \prec f(x).$$

Munissons $[D \to D']_{min}$ du faisceau de la convergence simple et supposons de plus que D' vérifie

(i) il est stable

(ii) ses spectres sont des sup-demi treillis.

Alors le fom $[D \to D']_{min}$ est interpolable et vérifie les deux conditions précédentes (i) et (ii).

<u>N.B.</u> : noter que (i) et (ii) $\Rightarrow$ D'interpolable d'après lemme 2.17.

<u>Démonstration</u> : $[D \to D']_{min}$ est stable d'après 3.10. Les spectres du D' sont des sup-demi lattices donc

$$f \prec g \quad \Rightarrow \quad f \sqsubseteq g \quad \Longrightarrow \quad s(f) \sqsubseteq s(g)$$

Donc si $f \neq g$ (sinon trivial) $s(f) \subsetneq s(g)$ d'où si $m(g)$ est le minorant approximant de g

$$\exists \, [e,e']_{m(g)} \in s(g) - s(f) .$$

On va prendre $h = f \sqcup [e,e']_{m(g)}$. Cette fonction est partout définie car (Lemme 1.2) :

$$h(x) = f(x) \sqcup [e,e']_{m(g)}(x) =$$

$$= \begin{cases} f(x) \sqcup e' & \text{si } e' \prec x \\ f(x) \sqcup m(g) & \text{sinon} \end{cases}$$

(a) $f(x) \sqcup e' \in s(g(x))$ puisque $e' \prec x \Rightarrow e' \in s(g(x))$ et $f(x) \in s(g))$ et $s(g(x))$ est un sup-semi-lattice

(b) $f(x) \sqcup m(g) \in s(g(x))$ puisque $\neg(e' \prec x) \Rightarrow$ $m(g) \prec g(x) \quad \forall x$ et $s(g(x))$ est un sup-semi-lattice.

D'où $\forall x \quad h(x) \prec g(x)$ i.e. $h \prec g$.

Donc on a $f \prec g$ et $f \sqsubseteq h \prec g$. Comme le faisceau $[D \to D']_{min}$ est stable, il vient

$$s(h) = (\downarrow h) \cap s(g), \quad f \sqsubseteq h \quad \text{et} \quad f \in s(g) \Rightarrow f \in s(h)$$

i.e., on a trouvé un h tel que $f < h < g$, d'où l'interpolabilité de $[D \to D']_{min}$. Le fait que ses spectres sont des sup-semi-lattices est évident. □

Ici encore on peut noter qu'au lieu de parler de $[D \to D']_{min}$, on aurait pu énoncer la proposition pour l'espace des fonctions régulières possédant un minorant approximant :

$$[D \to D']_m = \{f \in [D \to D'] \; : \; \exists m(f) \in D' \quad \forall x \; m(f) \prec f(x)\}$$

avec $[D \to D']_m \subseteq [D \to D']_{min}$.

La proposition 3.16 est particulièrement intéressante car ses conditions d'application (i) et (ii) sont héréditaires par passage à l'espace des fonctions régulières $[D \to D']_m$, et l'itération du processus est donc possible. Noter que les hypothèses ci-dessus sont toutes vérifiées par les treillis continus, avec $\forall f \; m(f) = \bot$, et même par les faisceaux continus $(\ll_i)$ interpolables sur des espaces ordonnés cohérents dans lesquels toute paire d'éléments majorés possède une borne supérieure. En particulier elles sont valables pour les cpl d'Egli.

Lorsqu'on est sur un fom élémentaire, le corollaire 3.14 peut recevoir une allure plus intuitive, qu'on retrouvera pour les échelons de Nolin.

3.15. : <u>PROPOSITION</u> (<u>lemme de la base</u>) :

Soit D un fom élementaire, D' un poset, $f : D \to D'$ régulière minorée, $min(f)$ un minorant de $f \neq \lambda x.\, min(f)$. Alors il existe une petite famille d'échelons

$$\{[e_i, e'_i]_{min(f)} : i \in I\} \quad \text{tq}$$

(i) $\quad \forall i \quad s(e_i) \subseteq \{e_j : j \in I\}$

(ii) $\quad i \neq j \Rightarrow e_i \neq e_j$

(iii) $\quad e_i \sqsubseteq e_j \Rightarrow e'_i \sqsubseteq e'_j$

(iv) $\forall i \quad e_i$ rationnel et $e'_i \neq \min(f)$

(v) $\quad f = \bigsqcup \{[e_i, e'_i]_{\min(f)} : i \in I\}$. □

<u>Démonstration</u> : $f : D \to D'$ régulière donnée $\Rightarrow$ elle est régulière sur $Rt(D)$. On va appliquer à $f|_{Rt(D)}$ 3.14. Prendre

$\{e_i\}_{i \in I} = \{v \in Rt(D) : v \in Dom(f) \text{ et } f(v) \neq \min(f)\}$

et $\forall i \in I \quad e'_i = f(e_i)$. La famille $\{[e_i, e'_i]_{\min(f)} : i \in I\}$ est non vide et vérifie (i) - (iv). Comme D est interpolable on a sur $Rt(D)$

$$f = \bigsqcup \{[v,w]_{\min(f)} : w < f(v)\} =$$
$$\bigsqcup \{[v, f(v)]_{\min(f)} : v \in Rt(D)\} =$$
$$\bigsqcup \{[e_i, e'_i]_{\min(f)} : i \in I\}$$

d'où la clause (v) (les bornes existent toujours puisque f est régulière).

Cette famille est minimale, car supprimer un $[e_i, e'_i]_{\min(f)}$ reviendrait à supposer que $f(e_i)$ peut être obtenu au moyen d'autres échelons par limite, donc e_i au moyen d'autre e_j comme limite $\Rightarrow e_i$ n'est pas rationnel $\Rightarrow$ contradiction. □

<u>III.1.2. - ECHELONS DE SCOTT. CAS GENERAL</u>

<u>3.18. PROPOSITION</u> : Soient D, D' des faisceaux ordonnés. Alors la fonction échelon $[e,e']_m$, ($e \in D$, $e' \in D'$, $m \in D'$, $m \subsetneq e'$) est une fonction régulière si et seulement si le faisceau ordonné D est interpolable au point e.

<u>Démonstration</u> : même démarche que pour le cas monique. On se ramène à l'équivalence

$$\forall \sigma \in \beta(x) \; \exists i \in Dom(\sigma) \quad e = \sigma(i) \iff (\exists \ell \; Dom(\sigma) \quad u = \sigma(\ell) \text{ tq } \forall \tau \in \beta(u) \; \exists j \in Dom(\tau) \quad e = \tau(j))$$

ce qui exprime exactement l'interpolabilité du faisceau D au point x. □

On ne fait donc que généraliser le cas monique.

3.19. LEMME : Si D et D' sont des faisceaux ordonnés tq $\forall x \in D$ la fibre de x a un élément minimal σ_{min} au sens suivant

$$\forall \sigma \in \beta(x) \quad \sigma \text{ est un prolongement de } \sigma_{min}$$

Alors pour toute fonction régulière $f : D \to D'$, possédant un minorant min(f) sur D, monotone ,

$$f = \sqcup \{[e,e']_{\min(f)} : \exists \sigma \in \beta(f(e)) \ \exists i \in \mathrm{Dom}(\sigma) \quad e' = \sigma(i)\}$$

où la borne supérieure est prise dans l'ensemble des fonctions monotones de D dans D'.

□

Démonstration : définissons

$$f' : x \to \sqcup \{[e,e']_{\min(f)} : \exists \sigma \in \beta(f(e)) \ \exists i \in \mathrm{Dom}(\sigma) \quad e' = \sigma(i)\}$$

Il vient

$$f'(x) = \sqcup \{[e,e']_{\min(f)} : \exists \sigma \in \beta(f(e)) \ \exists i \in \mathrm{Dom}(\sigma) \quad e' = \sigma(i)\}(x)$$

$$= \sqcup \{[e,e']_{\min(f)}(x) : \exists \sigma \in \beta(f(e)) \ \exists i \in \mathrm{Dom}(\sigma) \quad e' = \sigma(i)\}$$

$$= \sqcup \{e' : \exists e \ \forall \sigma \in \beta(x) \quad i \in \mathrm{Dom}(\sigma) \ e = \sigma(i) \quad \exists \tau \in \beta(f(e))$$
$$\exists j \in \mathrm{Dom}(\tau) \quad e' = \tau(j)\}$$

$$= \sqcup \{e' : \exists e \ldots \exists \tau \in \beta(f(e)) \ \exists j \in \mathrm{Dom}(\tau) \ e' = \tau(j)\}$$

= (reconstruction des points : e étant fixé

$$f(e) = \sqcup \{e' : \exists \tau \in \beta(f(e)) \ \exists j \in \mathrm{Dom}(\tau) \quad e' = \tau(j)\})$$

$$= \sqcup \{f(e) : \exists e \ldots\} =$$

$$= \sqcup \{f(e) : \exists e \quad \forall \sigma \in \beta(x) \quad \exists i \in \mathrm{Dom}(\sigma) \quad e = \sigma(i)\} =$$

$$= \sqcup \{f(e) : \exists e \quad \exists i \in \mathrm{Dom}(\sigma_{min}) \quad e = \sigma_{min}(i)\}$$

$= f(x)$.

□

Il existe cependant une autre façon de généraliser cette construction.

DEFINITION : faisceau ordonné stable.

Un faisceau ordonné sur un poset X est stable ssi

$\forall x, y \in X \quad \forall a \in N(X)$

$a \sqsubseteq x \sqsubseteq y \quad \Rightarrow$ les assertions suivantes sont équivalentes

(i) $\quad \exists \sigma \in \beta(x) \quad \exists i \in \mathrm{Dom}(\sigma) \quad a = \sigma(i)$

(ii) $\quad \exists \tau \in \beta(y) \quad \exists j \in \mathrm{Dom}(\tau) \quad a = \tau(j)$ □

On voit que cette définition généralise celle qui a été donné pour les faisceaux ordonnés moniques.

3.20. LEMME : Si D et D' sont des faisceaux ordonnés tels que D' est stable, alors pour toute fonction régulière $f : D \to D'$ monotone et possédant un minorant sur D, on a

$$f = \sqcup \{[e,e']_{\min(f)} : \exists \sigma \in \beta(f(e)) \quad \exists i \in \mathrm{Dom}(\sigma) \quad e' = \sigma(i)\}$$

où la borne supérieure est prise dans l'ensemble des fonctions monotones de D dans D'. □

Démonstration : on procède de même qu'au lemme précédent :

$$f'(x) = \sqcup \{[e,e']_{\min(f)} : \exists \sigma \in \beta(f(e)) \ \exists i \in \mathrm{Dom}(\sigma) \quad e' = \sigma(i)\}(x)$$

$$= \sqcup\{e' : \exists e \ \forall \sigma \in \beta(x) \ \exists i \in \mathrm{Dom}(\sigma) \quad e = \sigma(i) \quad \exists \tau \in \beta(f(e))$$
$$\exists j \in \mathrm{Dom}(\tau) \quad e' = \tau(j)\}$$

$$= \sqcup\{e' : \exists e \ldots \exists \tau \in \beta(f(e)) \ \exists j \in \mathrm{Dom}(\tau) \quad e' = \tau(j)\}$$

(stabilité de D' et $e \sqsubseteq x \Rightarrow f(e) \sqsubseteq f(x)$)

$$= \sqcup\{e' : \exists e \ldots \exists \nu \in \beta(f(x)) \ \exists j \in \mathrm{Dom}(\nu) \quad e' = \nu(j)\}$$

$$= \sqcup\{e' : \quad \exists \nu \in \beta(f(x)) \ \exists j \in \mathrm{Dom}(\nu) \quad e' = \nu(j)\}$$

$$= f(x) \quad .$$ □

D'où on tire :

3.21. PROPOSITION : Soit D, D' des faisceaux ordonnés, tq D est interpolable et D' est stable. Alors l'espace $[\![D \to D']\!]_{\min}$ des fonctions régulières monotones de D dans D' minorées sur D, ordonné extensionnellement est tel que :

il est muni d'une structure de fom fidèle par la fonction spectre $s : f \mapsto \{[e,e']_{min(f)} : \exists \sigma \in \mathcal{B}(f(e)) \; \exists i \in Dom(\sigma) \quad e' = \sigma(i)\}$ où $min(f)$ décrit l'ensemble des minorants de f. □

Démonstration : analogue à 3.14. □

III.2. - CONTINUITE DE L'ESPACE DES IMAGES

3.22. PROPOSITION : Soit D un fom interpolable, et D' un faisceau continu au sens de Scott, interpolable, tel que $\perp \in D'$. Alors l'espace $[D \to D']$ des fonctions régulières ordonné extensionnellement possède une structure de faisceau continu et contient un $\perp$. □

Démonstration : (Attention : dans la suite de cette démonstration, pour simplifier l'écriture, on note $[e,e']$ les échelons de Scott $[e,e']_\perp$)

D'après le Corollaire 3.15 (i) nous avons une structure de faisceau sur $[D \to D]$ par

$$\forall f \in [D \to D'] \qquad f = \sqcup\{[e,e']_\perp \; : e' \prec f(e)\}$$

avec $\forall f \quad min(f) = \perp$.

La question est de montrer que $[D \to D']$ possède une structure de faisceau continu, i.e. que sa relation $\ll_i$ way-below (bien en-dessous) est approximante. Puisque $[D \to D']$ possède déjà une structure de faisceau, il suffit de montrer que tout $[e,e']$ tq $e' \prec f(e)$ vérifie $[e,e'] \ll_i f$.

Hypothèse : soit $[e,e']$ tq $e' \prec f(e)$.

Nous voulons $[e,e'] \ll_i f$ c'est-à-dire :

$\forall$ idéal convergent $I \subseteq [D \to D'] \quad f \sqsubseteq \sqcup I \Rightarrow [e,e'] \in I$

$\Longleftrightarrow$

$\forall I \in Id_{cv}[D \to D']$

$$(\forall x \in D \quad f(x) \sqsubseteq \bigsqcup_{h \in I} h(x)) \Longrightarrow [e,e'] \in I$$

Notons $\prec$ l'approximation du faisceau continu sur D'. Nous avons :

$$\forall x \in D \qquad f(x) \sqsubseteq \bigsqcup_{h \in I} h(x) \Rightarrow$$

$$\bigsqcup_{h \in I} h(x) \sqsupseteq f(x) \succ [e,e'] (x) = \begin{cases} e' & \text{si} \quad e \prec x \\ \bot & \text{sinon.} \end{cases}$$

Nous diviserons le domaine des x en deux parts :

- si non $(e \prec x)$ $(e \notin s(x))$, nous ne pouvons rien dire à partir de la condition précédente puisque $\bot \prec$ n'importe quoi

- si $e \prec x$ $(e \in s(x))$ alors

$$[e,e'] (x) = e' \prec f(x) \sqsubseteq \bigsqcup_{h \in I} h(x)$$

donc ($\prec$ est way-below sur D') $\exists h(x)$ $e' \prec h(x)$ d'après Proposition 2.18. (i), car D' interpolable.

D'où on tire :

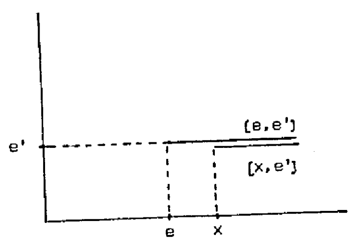
$$[x,e'] \in \{[a,a'] : a' \prec h(a)\} \Rightarrow$$

$$[x,e'] \prec h \Rightarrow [x,e'] \sqsubseteq h \Rightarrow [x,e'] \in I$$

pour tout $x \succ e$.

Maintenant nous affirmons :

$$[e,e'] = \bigsqcup \{[x,e'] : e \prec x\}$$



En fait $[e,e'] (u) = \begin{cases} e' & \text{si} \quad e \prec u \\ \bot & \text{sinon} \end{cases}$

D'autre part:

$$(\bigsqcup\{[x,e'] : e \prec x\}) (u) = \bigsqcup\{[x,e'] (u) : e \prec x\}$$

$$= \bigsqcup\{\text{si } x \prec u \text{ alors } e' \text{ sinon } \bot : e \prec x\} =$$

$$= \bigsqcup\{e' : e \prec x \text{ et } x \prec u\} = \text{(interpolabilité de } D)$$

$$= \bigsqcup\{e' : e \prec u\} = \begin{cases} e' & \text{si} \quad e \prec u \\ \bot & \text{sinon} \end{cases}$$

D'où la conclusion $[e,e'] = \sqcup\{[x,e'] : e \prec x\}$.

Donc, puisque $\forall x \succ e \quad [x,e'] \sqsubseteq h$ nous obtenons :

$$[e,e'] = \bigsqcup_{e \prec x} [x,e'] \sqsubseteq h \Rightarrow [e,e'] \in I$$

puisque I est un idéal, donc une section commençante.

Et ceci est vrai pour tout idéal I. Donc

$$\forall [e,e'] \qquad e' \prec f(e) \Rightarrow [e,e'] \ll f.$$

Ainsi

$$\{[e,e'] : e' \prec f(e)\} \subseteq \{[e,e'] : [e,e'] \ll f\} \subseteq$$
$$\{g \in [D \to D'] : g \ll f\} = \mathop{\downarrow\!\!\downarrow} f$$

Prenant les bornes supérieures, il vient :

$$f = \sqcup\{[e,e'] : e' \prec f(e)\} \sqsubseteq \sqcup(\mathop{\downarrow\!\!\downarrow} f) \sqsubseteq f$$

puisque $g \ll f \Rightarrow g \sqsubseteq f$. D'où $f = \sqcup \mathop{\downarrow\!\!\downarrow} f$.

Donc la fonction spectrale

$$f \mapsto \mathop{\downarrow\!\!\downarrow} f$$

définit un faisceau sur $[D \to D']$, et il est continu par définition.

Le $\perp$ de $[D \to D']$ est $\lambda x.\perp$. □

De même que la reconstruction par échelons de Scott de $[D \to D']_{min}$ se généralise des fom D aux faisceaux ordonnés interpolables D lorsque D' est stable, la proposition précédente se généralise de la même manière.

<u>DEFINITION</u> : <u>Interpolabilité globale</u>.

Soit $\langle \lambda, \beta \rangle$ un faisceau ordonné sur X.
On dira que ce faisceau est <u>globalement interpolable au point $e \in X$</u> ssi $\forall u \in X$ on a l'equivalence suivante

$$(\forall \tau \in \beta(u) \ \exists k \in Dom(\tau) \quad e = \tau(k)) \Longleftrightarrow$$

$$\exists x \in X \quad \begin{cases} (\forall \sigma \in \beta(x) \ \exists i \in Dom(\sigma) \quad e = \sigma(i)) \text{ et} \\ (\forall \tau \in \beta(u) \ \exists j \in Dom(\tau) \quad x = \tau(j)) \end{cases}$$

Il est globalement interpolable ssi il est <u>globalement interpolable</u> en tout point. □

L'interpolabilité globale est une autre façon de généraliser aux faisceaux ordonnés l'interpolabilité qui existe sur les fom.

<u>3.23. Fait</u> : Les faisceaux ordonnés (élémentaires) sont globalement interpolables en tout point rationnel.

<u>3.24. PROPOSITION</u> : Soit D un faisceau ordonné interpolable et globalement interpolable, D' un faisceau continu au sens de Scott, interpolable et possédant $\perp$. Alors l'espace $[\![D \to D']\!]$ des fonctions régulières monotones, ordonné extensionnellement possède une structure de faisceau continu et possède un $\perp$. □

<u>Démonstration</u> : on procède comme pour la proposition précédente.
Avec les mêmes notations il vient : pour tout $x \in D$ tq.

$$\forall \sigma \in \beta(x) \quad \exists i \in \mathrm{Dom}(\sigma) \quad e = \sigma(i)$$

on a un $h(x)$ tel que $e' \ll h(x)$ c'est-à-dire

$$[x,e'] \in \{[a,a'] \;:\; a' \ll h(a)\}$$

Maintenant du fait de l'interpolabilité globale de D on a :

$$[e,e'] = \sqcup \{[x,e'] \;:\; \forall \sigma \in \beta(x) \;\; \exists i \in \mathrm{Dom}(\sigma) \;\; e = \sigma(i)\}$$

Ceci vient de ce que : pour tout $u \in D$

$$\sqcup \{[x,e'] \;:\; \forall \sigma \in \beta(x) \;\; \exists i \in \mathrm{Dom}(\sigma) \;\; e = \sigma(i)\}(u) =$$

$$= \sqcup \{e' \;:\; \forall \tau \in \beta(u) \;\; \exists j \in \mathrm{Dom}(\tau) \;\; x = \tau(j) \;\text{ et }$$
$$\forall \sigma \in \beta(x) \;\; \exists i \in \mathrm{Dom}(\sigma) \;\; e = \tau(i)\}$$

= (interpolabilité globale)

$$= \sqcup \{e' \;:\; \forall \tau \in \beta(u) \;\; \exists \ell \in \mathrm{Dom}(\tau) \;\; e = \tau(\ell)\}$$

$$= \begin{cases} e' & \text{si } \forall \tau \in \beta(u) \;\; \exists \ell \in \mathrm{Dom}(\tau) \;\; e = \tau(\ell) \\ \perp & \text{sinon} \end{cases}$$

D'où la conclusion

$$[e,e'] = \sqcup \{[x,e'] \;:\; \forall \sigma \in \beta(x) \;\; \exists i \in \mathrm{Dom}(\sigma) \;\; e = \sigma(i)\}$$

Maintenant on continue le raisonnement de la même manière que lorsque D est un fom. d'où la proposition. □

III.3. - ECHELONS DE NOLIN

3.25. LEMME : Soit D, D' des faisceaux ordonnés, alors $\forall e \in D$, $e', m \in D'$ $e \sqsubseteq m$, l'échelon $\langle e,e'\rangle_m$ est une fonction régulière.

Démonstration : Comme

$$\langle e,e'\rangle_m : z \mapsto \begin{cases} e' & \text{si } z \sqsubseteq e \\ m & \text{sinon} \end{cases}$$

on a pour tout $z \in D$ $\forall \sigma \in \beta(z)$

$$\langle e,e'\rangle_m(z) = e' \Leftrightarrow z \sqsubseteq e \text{ (en supposant } m \neq e') \Leftrightarrow \bigsqcup \sigma \sqsubseteq e \Leftrightarrow$$

$$\forall u \in \sigma \quad u \sqsubseteq e \Leftrightarrow \forall u \in \sigma \quad \langle e,e'\rangle_m(u) = e$$

donc $\langle e,e'\rangle_m(z) = \bigsqcup(\langle e,e'\rangle_m(\sigma))$.

Le cas $m = e'$ découle du lemme 1.2 (i) .

D'où le lemme. □

On voit donc que pour les échelons de Nolin la structure "fine" du faisceau ordonné D n'intervient pas (seule la structure de poset joue) et la régularité des $\langle e,e'\rangle_m$ est pour ainsi dire "gratuite", et valable pour tous les faisceaux ordonnés et pas seulement pour les interpolables comme pour les $[e,e']_m$.

3.6. LEMME : Si D et D' sont des faisceaux ordonnés, alors pour toute fonction $f : D \to D'$ monotone majorée, pour tout majorant $max(f)$ de f, on a

$$f = \sqcap\{\langle e,e'\rangle_{max(f)} : e' \sqsupseteq f(e)\}$$

où la borne inférieure est prise dans l'ensemble des fonctions monotones de D dans D'.

(On pourrait dire que ce lemme établit une structure de faisceau sur l'espace $[D \to D']_{\sqsubseteq}$ des fonctions monotones de D dans D', les échelons de Nolin étant monotones).

Démonstration : On a $\forall x \in D$

$$(\sqcap\{\langle e,e'\rangle_{max(f)} : e' \sqsupseteq f(e)\})(x) =$$

$$\sqcap\{\langle e,e'\rangle_{max(f)}(x) : e' \sqsupseteq f(e)\} =$$

$$\sqcap\{\text{si } x \sqsubseteq e \text{ alors } e' \text{ sinon } max(f) : e' \sqsupseteq f(e)\} =$$

$$\sqcap\{e' : x \sqsubseteq e \text{ et } f(e) \sqsubseteq e'\} =$$

$$\sqcap\{e' : f(x) \sqsubseteq e'\} = f(x) \quad .$$

□

Ici encore, il apparait que les échelons de Nolin ont des propriétés plus simples que les échelons de Scott. Il n'y a donc pas symétrie entre ces deux notions, et elles ne sont pas réductibles l'une à l'autre.

3.27. COROLLAIRE : Soit D, D' deux faisceaux ordonnés, $[D \to D']_{max}$ l'espace des fonctions régulières de D dans D' qui sont majorées et monotones croissantes. Alors l'espace $[D \to D']_{max}$ est muni d'une structure de fom fidèle par la fonction spectre :

$$s : f \mapsto \{ \langle e, e' \rangle_{max(f)} : e' \sqsupseteq f(e) \}$$

où $max(f)$ décrit l'ensemble des majorants de f. $\square$

De même qu'au Corollaire 3.14, on aurait pu se contenter dans 3.27. d'une structure de faisceau sur $[D \to D']_{max}$, la fonction spectre étant remplacée par la fibre.

$$\beta(f) = \{ \{ \langle e, e' \rangle_{max(f)} : e' \sqsupseteq f(e) \} \; \forall x f(x) \sqsubseteq max(f) \}$$

chaque majorant permettant de "démarrer" une approximation et donnant un élément de la fibre.

Comme pour 3.14., le fom de 3.27. est évidemment ici élémentaire dans le cas général, et a pour noyau

$$\{ \langle e, e' \rangle_m : e \in D \;,\; e', m \in D' \quad e' \sqsubseteq m \}$$

c'est de même une connection de Galois (pour $\lambda = \sqcap$).

L'équivalent de la clause (ii) de 3.14. est sans intérêt ici, puisqu'il correspond à l'ordre auxiliaire $\sqsupseteq$.

Démonstration : évident. $\square$

On peut énoncer ici l'équivalent de 3.17. :

3.28. PROPOSITION : (Lemme de la base) :

Soit D un faisceau ordonné élémentaire, D' un poset, $f : D \to D'$ régulière majorée et monotone croissante, $max(f)$ un majorant de $f \neq \lambda x.\, max(f)$. Alors il existe une plus petite famille d'échelons

$$\{ \langle e_i \,,\, e'_i \rangle_{max(f)} : i \in I \} \quad \text{tq}$$

(i) $\forall i \quad s(e_i) \subseteq \{e_j : j \in I\}$

(ii) $i \neq j \Rightarrow e_i \neq e_j$

(iii) $e_i \subseteq e_j \Rightarrow e'_i \subseteq e'_j$

(iv) $\forall i \quad e_i$ est rationnel et $e'_i \neq max(f)$

(v) $f = \sqcap \{\langle e_i , e'_i \rangle_{max(f)} : i \in I\}$

Démonstration : analogue à 3.17. Prendre

$$\{e_i\}_{i \in I} = \{v \in Rt(D) : v \in Dom(f) \text{ et } f(v) \neq max(f)\}$$

$\forall i \quad e'_i = f(e_i)$. La famille

$$\{\langle e_i , e'_i \rangle_{max(f)} : i \in I\}$$

vérifie (i) à (iv). Pour tout rationnel $v \in D$

$$f(v) = (\sqcap\{\langle e_i , e'_i \rangle_{max(f)} : i \in I\})(v)$$

d'après lemme 3.26. La proposition découle de la proposition 3.5. □

On peut itérer la construction du Corollaire 3.27. sans aucun problème. La signification intuitive de ce processus, pour le programmeur, reste cependant à éclaircir si on ne se donne pas des conditions supplémentaires sur les espaces de départ. Il n'y a pas d'équivalent de la proposition 3.16. et de sa démonstration. Ceci est une des raisons semble-t-il pour lesquellles la solution adoptée par Nolin pour obtenir un faisceau de la convergence simple sur l'espace des fonctions est de se donner un o.e. trivial sur cet espace. On peut en effet regarder cet oe comme un faisceau de la convergence simple, si on suppose qu'on a de façon implicite un o.e. trivial (pour $\lambda = \sqcap$) sur l'espace D' des valeurs des fonctions, permettant de faire des approximations par le "haut". Mais du fait de la trivialité de cet o.e. ceci ne nous donne plus de façon de calculer sur les fonctions. (cf. Chapitre IV).

En résumé, comparant la méthode des échelons de Scott à celle des échelons de Nolin, nous pouvons dire que l'on a là des approches assez différentes au même problème : reconstruire l'espace des fonctions régulières.

L'une utilise des "micro-éléments" $[e,e']_m$ $e \in D$, $e',m \in D'$ et obtient les autres éléments de l'espace comme bornes supérieures.

Pour ce on a besoin en particulier de la régularité des $[e,e']_m$, i.e. de l'interpolabilité de D. L'autre utilise des "macro-éléments" $\langle e,e' \rangle_m$, $e \in D$, $e',m \in D'$ qui sont toujours réguliers, et reconstruits les autres éléments comme bornes inférieures. Contrairement à la première, la seconde méthode ne permet pas d'obtenir (presque) immédiatement un faisceau de la convergence simple sur l'espace des fonctions.

III.4. - ANNEXE : LES PREORDRES D'EGLI-MILNER ET DE SMYTH ET LE PROBLEME DES SYSTEMES APPROXIMANTS ASSOCIES AU PARALLELISME

L'argument de cette section est que le powerdomain au sens de Plotkin [1976], ou le powerdomain "faible" au sens de Smyth [1978], présentés par ces auteurs comme des cpo's de parties de cpo's, se ramènent en fait essentiellement à des structures de faisceaux, que nous allons exhiber (en utilisant 3.14), sur des espaces de fonctions régulières quotientés par une relation d'équivalence. Les arbres utilisés par Smyth pour simplifier la construction originale de Plokin ne sont rien d'autre qu'une façon de dessiner le graphe de ces fonctions régulières.

D'une manière plus précise, nous allons voir que certaines de ces fonctions régulières sont des systèmes approximants directement déduits des constructions liées aux échelons de Scott et que ces auteurs ont voulu lier avec les préordres d'Egli-Milner (ou de Smyth) de façon à pourvoir reconstruire l'espace des fonctions régulières tout entier. Ces auteurs ont montré un résultat équivalent à dire que cette construction s'insère bien dans la théorie des cpo modulo le passage à l'ordre quotient.

La notion de faisceau va nous permettre

1. de simplifier encore les constructions de Smyth et de Plotkin, ainsi que leurs démonstration,
2. d'indiquer le lien qui existe entre elles et le problème général de construction de faisceaux sur des objets fonctionnels.

3. de relier le powerdomain à la dichotomie existant entre les faisceaux les plus généraux et ceux dont les systèmes approximants sont des ensembles où chaque élément est indexé par lui-même.

Notations : Soit N un ensemble fini, N^{ω} l'ensemble des suites finies et infinies sur N muni de l'oe dont les rationnels sont les suites finies, $\overline{Rat} = N^{\omega} - Rt(N^{\omega})$, E un cpo discret et D un cpo quelconque.

Dans la suite on sera dans le cas de figure suivant : comment mettre une structure de faisceau sur $[X \to Y]$ avec $X = N^{\omega}$, $Y \in \{E, D\}$.

En utilisant un argument analogue à celui de Plotkin 1976 pp 458 (qui s'appuie essentiellement sur le fait que N est fini), on voit que $\forall f \in [N^{\omega} \to E]$, $(f(\overline{Rat})$ est fini ou $\perp \in f(\overline{Rat}))$.
De plus $f(\overline{Rat}) \neq \emptyset$ trivialement.

DEFINITION : Préordres sur $[N^{\omega} \to E]$:

(i) Préordre de Smyth : $\forall f$, $g \in [N^{\omega} \to E]$
$f \sqsubseteq_S g \iff g(\overline{Rat}) \subseteq \uparrow f(\overline{Rat})$

(ii) Préordre d'Egli-Milner : $\forall f, g \in [N^{\omega} \to E]$
$f \sqsubseteq_M g \iff g(\overline{Rat}) \subseteq \uparrow f(\overline{Rat})$ et
$f(\overline{Rat}) \subseteq \downarrow g(\overline{Rat})$ □

3.29. Fait : $\forall f, g \in [N^{\omega} \to E]$

(i) $f \sqsubseteq_S g \iff \perp \in f(\overline{Rat})$ ou $g(\overline{Rat}) - \{\perp\} \subseteq f(\overline{Rat})$

(ii) $f \sqsubseteq_M g \iff \perp \notin f(\overline{Rat})$ et $f(\overline{Rat}) = g(\overline{Rat})$
ou $\perp \in f(\overline{Rat})$ et $f(\overline{Rat}) \setminus \{\perp\} \subseteq g(\overline{Rat})$

(iii) si $E = 2 = \begin{array}{c} \top \\ | \\ \perp \end{array}$, alors

$f \sqsubseteq_S g \iff \perp \in f(\overline{Rat})$ ou $g(\overline{Rat}) \subseteq f(\overline{Rat})$

$f \sqsubseteq_M g \iff \{\top\} = f(\overline{Rat}) = g(\overline{Rat})$ ou
$(\perp \in f(\overline{Rat})$ et $f(\overline{Rat}) \setminus \{\perp\} \subseteq g(\overline{Rat}))$

(iv) $f \sqsubseteq g \Longrightarrow f \sqsubseteq_S g$ et $f \sqsubseteq_M g$ i.e. l'ordre extensionnel est plus fin que les préordres d'Egli-Milner et de Symth.

Démonstration : (i) - (iii) immédiats.

$$\text{(iv)} : \quad f \sqsubseteq g \Leftrightarrow \forall x \quad f(x) \sqsubseteq g(x) \Rightarrow \begin{array}{l} f(\overline{Rat}) \subseteq \downarrow g(\overline{Rat}) \\ g(\overline{Rat}) \subseteq \uparrow f(\overline{Rat}) \end{array}$$

i.e. $f \sqsubseteq_M g$ et $f \sqsubseteq_S g$. □

Soit $=_S$ (resp. $=_M$) la relation d'équivalence induite par $\sqsubseteq_S$ (resp. $\sqsubseteq_M$), i.e.

$f =_S g \Leftrightarrow f \sqsubseteq_S g$ et $g \sqsubseteq_S f$ (resp. etc...)

3.30. LEMME : $\forall f,g \in [N^\omega \to E]$

(i) $f =_M g \Longleftrightarrow f(\overline{Rat}) = g(\overline{Rat})$

(ii) $f =_S g \Longleftrightarrow f =_M g$ ou $\perp \in f(\overline{Rat}) \cap g(\overline{Rat})$ □

Démonstration :

(i) $f =_M g \Longleftrightarrow f(\overline{Rat}) \subseteq \downarrow g(\overline{Rat})$ et $g(Rat) \subseteq \uparrow f(\overline{Rat})$
et $g(\overline{Rat}) \subseteq \downarrow f(\overline{Rat})$ et $f(\overline{Rat}) \subseteq \uparrow g(\overline{Rat})$

$\Longleftrightarrow$ (a) $f(\overline{Rat}) \sqsubseteq \downarrow g(\overline{Rat}) \cap \uparrow g(\overline{Rat})$
(b) $g(\overline{Rat}) \sqsubseteq \downarrow f(\overline{Rat}) \cap \uparrow f(\overline{Rat})$

(a) $\Longleftrightarrow f(\overline{Rat}) \sqsubseteq \{u \in E : \exists x,y \in g(\overline{Rat}) : x \sqsubseteq u \sqsubseteq y\}$

Comme E est discret $x \sqsubseteq u \sqsubseteq y \Longleftrightarrow u \in \{x,y\}$,

alors (a) $\Longleftrightarrow f(\overline{Rat}) \subseteq g(\overline{Rat})$
(b) $\Longleftrightarrow g(\overline{Rat}) \subseteq f(\overline{Rat})$

donc $f =_M g \Longleftrightarrow f(\overline{Rat}) = g(\overline{Rat})$

(ii) de la même façon on a :

$f =_S g \iff g(\overline{Rat}) \subseteq \uparrow f(\overline{Rat})$ et $f(\overline{Rat}) \subseteq \uparrow g(\overline{Rat})$

$\iff$ (comme E est discret)

$$\begin{cases} \bot \in f(\overline{Rat}) & \text{ou } g(\overline{Rat}) \subseteq f(\overline{Rat}) \text{ et} \\ \bot \in g(\overline{Rat}) & \text{ou } f(\overline{Rat}) \subseteq g(\overline{Rat}) \end{cases}$$

ce qui donne quatre cas

(1) $\bot \in f(\overline{Rat}) \cap g(\overline{Rat})$

(2) $\bot \in f(\overline{Rat}) \subseteq g(\overline{Rat})$

(3) $\bot \in g(\overline{Rat}) \subseteq f(\overline{Rat})$

(4) $f(\overline{Rat}) = g(\overline{Rat})$

qui se réduisent à deux

$$\bot \in f(\overline{Rat}) \cap g(\overline{Rat}) \quad \text{ou} \quad f(\overline{Rat}) = g(\overline{Rat}).$$

d'où le lemme. □

Nous définissons maintenant les mêmes préordres sur $[N^\omega \to D]$ suivant une démarche parallèle à celle de Plotkin.

<u>DEFINITION : Préordres sur $[N^\omega \to D]$</u>

(i) <u>Préordre de Smyth</u> : $\forall f,g \in [N^\omega \to D]$

$f \sqsubseteq'_S g \iff \forall$ cpo discret $E\ \forall h : D \to E$ régulière $h(\circ f) \sqsubseteq_S (h \circ g)$

(ii) <u>Préordre d'Egli-Milner</u> : $\forall f, g \in [N^\omega \to D]$

$f \sqsubseteq'_M g \iff \forall$ cpo discret $E\ \forall h : D \to E$ régulière $(h \circ f) \sqsubseteq_M (h \circ g)$

□

<u>3.31. Fait</u> :

(o) les relations $\sqsubseteq'_S$ et $\sqsubseteq'_M$ sont réflexives et transitives (i.e. des préordres).

(i) si D est un cpo discret alors $f \sqsubseteq'_M g \Leftrightarrow f \sqsubseteq_M g$ et $f \sqsubseteq'_S g \Leftrightarrow f \sqsubseteq'_S g$

(ii) $\forall f,g \in (N^\omega \to D)$ $f \sqsubseteq g \Rightarrow f \sqsubseteq'_S g$ et $f \sqsubseteq'_M g$ (i.e., l'ordre extensionnel est plus fin que les preordres $\sqsubseteq'_S$ et $\sqsubseteq_M$.

Démonstration : (i) préordre de Smyth : on a trivialement $f \sqsubseteq'_S g \Rightarrow f \sqsubseteq_S g$. (Il suffit de prendre $D = E$, $h = id_E$). Réciproquement, soit $h \in [D \to E]$, a-t-on $(h \circ g)(\overline{Rat}) \subseteq \uparrow (h \circ f)(\overline{Rat})$.

Deux cas se présentent. Si $\perp \notin (h \circ f)(\overline{Rat})$ on a

E discret $\Rightarrow$ si $\perp \notin f(\overline{Rat})$, alors

$$f(\uparrow f(\overline{Rat})) = (h \circ f)(\overline{Rat}) = \uparrow (h \circ f)(\overline{Rat})$$

Or $\quad f \sqsubseteq_S g = g(\overline{Rat}) \quad \subseteq \quad \uparrow f(\overline{Rat}) \quad \Rightarrow$

$$(h \circ g)(\overline{Rat}) \subseteq h(\uparrow f(\overline{Rat})) = \uparrow (h \circ f)(\overline{Rat})$$

Si $\perp \in (h \circ f)(\overline{Rat})$ on a $\uparrow (h \circ f)(\overline{Rat}) = E$ d'où la conclusion trivialement. Donc $f \sqsubseteq_S g \Rightarrow f \sqsubseteq'_S g$, d'où l'équivalence.

□

Préordre d'Egli-Milner : de même $f \sqsubseteq_M g \Leftrightarrow f \sqsubseteq_S g$ et $f(\overline{Rat}) \subseteq \downarrow g(\overline{Rat}) \Rightarrow (h \circ g)(\overline{Rat}) \subseteq h(\downarrow g(\overline{Rat})) \subseteq \downarrow (h \circ g)(\overline{Rat})$ par monotonie, d'où la conclusion.

□

(ii) trivial, analogue à 3.29.

On peut donc confondre $\sqsubseteq'_S$ et $\sqsubseteq_S$ (resp. $\sqsubseteq'_M$ et $\sqsubseteq_M$) dans le cas des cpo discrets.

(h o g)

Notation : dans la suite nous noterons $\sqsubseteq_S$ le préordre $\sqsubseteq'_S$ (resp. $\sqsubseteq_M$ le préordre $\sqsubseteq'_M$).

(h o g)

□

3.32. PROPOSITION (lemme de représentation) :

Soit $f, g \in [N^\omega \to D]$, $2 = \begin{smallmatrix}\top \\ \perp\end{smallmatrix}$ l'espace de Sierpinski. Alors

(i) $\quad f \sqsubseteq_M g \iff \forall h : D \to 2$ continue

$\qquad (h \circ f) \sqsubseteq_M (h \circ g)$

(ii) $\quad f \sqsubseteq_S g \iff \forall h : D \to 2$ continue

$\qquad (h \circ f) \sqsubseteq_S (h \circ g)$ $\square$

<u>Démonstration</u> : l'implication $\Rightarrow$ est évidente dans les deux cas :

(i) $\Leftarrow$: soit $\forall h : D \to 2 \ (h \circ f) \sqsubseteq_M (h \circ g)$

soit $\gamma : D \to 2$ régulière, E discret

soit $\eta : E \to 2$

$x \mapsto T$ si $x \neq \perp \wedge x \in (\gamma \circ f)\mathrm{Rat}$

$\perp$ sinon

$$\begin{tikzcd} & N^{\omega} \arrow[d, "f{,}g"] & \\ & D \arrow[dl, "h"'] \arrow[dr, "\gamma"] & \\ 2 & & E \arrow[ll, "\eta"] \end{tikzcd}$$

η est continue

Appliquant l'hypothèse on a

$\eta \circ \gamma \circ f \sqsubseteq_M \eta \circ \gamma \circ g$, c'est-à-dire d'après 3.29.

$(\perp \notin (\eta \circ \gamma \circ f)\overline{\mathrm{Rat}}$ et $(\eta \circ \gamma \circ f)\overline{\mathrm{Rat}} = (\eta \circ \gamma \circ g)\overline{\mathrm{Rat}}$

ou $(\perp \in (\eta \circ \gamma \circ f)\overline{\mathrm{Rat}}$ et $(\eta \circ \gamma \circ f)\overline{\mathrm{Rat}} \setminus \{\perp\} \subseteq (\eta \circ \gamma \circ g)\overline{\mathrm{Rat}}$

1er cas : $\perp \notin (\eta \circ \gamma \circ f)\overline{\mathrm{Rat}} \Rightarrow$

$\perp \notin (\gamma \circ f)(\overline{\mathrm{Rat}})$ et $(\gamma \circ f)(\overline{\mathrm{Rat}}) = (\gamma \circ g)(\overline{\mathrm{Rat}})$

2ème cas : $\perp \in (\eta \circ \gamma \circ f)\overline{\mathrm{Rat}} \Rightarrow \perp \in (\gamma \circ f)\overline{\mathrm{Rat}}$

Considérons maintenant $\forall e \in (\gamma \circ f)(\overline{\mathrm{Rat}}) \setminus \{\perp\}$ l'échelon de Scott $\eta_e = [e, T]_{\perp}$, qui est continu de E dans 2 (E interpolable).

Appliquant l'hypothèse il vient :

$$\eta_e \circ \gamma \circ f \sqsubseteq_M \eta_e \circ \gamma \circ g$$

d'où

$\perp \in (\eta_e \circ \gamma \circ f)(\overline{\mathrm{Rat}})$ et

$(\eta_e \circ \gamma \circ f)(\overline{\mathrm{Rat}}) \setminus \{\perp\} \subseteq (\eta_e \circ \gamma \circ g)(\overline{\mathrm{Rat}})$

i.e. $e \in (\gamma \circ g)(\overline{\mathrm{Rat}})$, d'où

$(\gamma \circ f)(\overline{\mathrm{Rat}}) \setminus \{\perp\} \subseteq (\gamma \circ g)(\overline{\mathrm{Rat}})$

d'où en remettant les deux cas ensembles et en appliquant à nouveau 3.29,

$$f \sqsubseteq_M g$$

(ii) on procède de la même manière; gardant le même schéma et les mêmes variables, on a :

si $\eta : E \to 2$: $x \mapsto T$ si $x \neq \perp \wedge x \in (\gamma \circ f)\overline{Rat}$

$\perp$ sinon

η est continue $\Rightarrow$

$\eta \circ \gamma \circ f \sqsubseteq_S \eta \circ \gamma \circ g \Rightarrow$ (d'après (3.29)

$\perp \in (\eta \circ \gamma \circ f)(\overline{Rat})$ ou $(\eta \circ \gamma \circ g)\overline{Rat} \subseteq (\eta \circ \gamma \circ f)\overline{Rat}$

1er cas : $\perp \notin (\eta \circ \gamma \circ f)\overline{Rat} \Rightarrow$

$(\eta \circ \gamma \circ g)\overline{Rat} \subseteq (\eta \circ \gamma \circ f)\overline{Rat} \Rightarrow$

$(\eta \circ \gamma \circ g)\overline{Rat} = \{T\} \Rightarrow (\gamma \circ g)\overline{Rat} \subseteq (\gamma \circ f)\overline{Rat}$

2ème cas : $\perp \in (\eta \circ \gamma \circ f)\overline{Rat} \Rightarrow \perp \in (\gamma \circ f)\overline{Rat}$

d'où rassemblant les deux cas et appliquant 3.29.,

$$f \sqsubseteq_S g$$

□

N^ω est un fom élémentaire, les $f \in [N^\omega \to D]$ sont minorées par $\perp$, donc la proposition 3.18. s'applique et

$$\forall f \in [N^\omega \to D] \quad f = \sqcup \{[e, f(e)]_\perp \quad : e \in Rt(N^\omega)\}$$

Nous allons voir que cette structure de faisceau sur $[N^\omega \to D]$ ordonné extensionnellement "passe", ou plutôt "filtre" , à l'espace $[N^\omega \to D]$ préordonné par $\sqsubseteq_S$ ou $\sqsubseteq_M$, et la construction du powerdomain consiste fondamentalement en ce passage. On obtient ainsi "presque" une structure de ω-faisceau sur $[N^\omega \to D]$, à ceci près que la fonction limite est multivoque.

Nous avons d'abord un résultat de compactité (la clause (ii) est analogue au résultat de Smyth [SMY] pp 28) :

3.33. LEMME (Propriété de compacite)

Soit N un ensemble fini, D un cpo, l'ensemble $\mathcal{N}$

$\mathcal{N} = \{ \bigsqcup_{q \leq n} [e_q, e'_q] \; : n \in \mathbb{N}, \; e_q \in Rt(N^\omega)$ et

$$(\bigsqcup_{q \leq n} [e_q \, , e'_q]_\perp) \, (e_q) = e'_q\}$$

$(\bigsqcup_{q \leq n} [e_q, e'_q]_\perp)(e_q) = e'_q \}$

Alors $\forall f, g \in [N^\omega \to D]$

(i) $f \sqsubseteq_S g \iff (\forall a \in \mathcal{N} \quad a \sqsubseteq_S f \Rightarrow a \sqsubseteq_S g)$

(ii) $f \sqsubseteq_M g \iff (\forall a \in \mathcal{N} \quad a \sqsubseteq_M f \Rightarrow a \sqsubseteq_M g)$ □

<u>DEMONSTRATION</u> : (en appliquant le lemme de réprésentation)

(ii) est plus simple :

$\Leftarrow$ est obtenue par transitivité

$\Rightarrow$: soit $h \in [D \to 2]$, $a \in \mathcal{N}$

$a \sqsubseteq_M f \iff \{T\} = (h \circ a)\overline{Rat} = (h \circ f)\overline{Rat}$

ou $\{ \perp \in (h \circ a)\overline{Rat}$ et $(h \circ a)(\overline{Rat}) \setminus \{\perp\} \subseteq (h \circ f)\overline{Rat}\}$

$a \sqsubseteq_M g \iff \{T\} = (h \circ a)\overline{Rat} = (h \circ g)\overline{Rat}$

ou $\perp \in (h \circ a)Rat$ et $(h \circ a)(\overline{Rat}) \setminus \{\perp\} \subseteq (h \circ f)\overline{Rat}$

1/ $\perp \notin (h \circ a)\overline{Rat} \Rightarrow (h \circ a)\overline{Rat} = (h \circ f)\overline{Rat} = (h \circ g)\overline{Rat}$

2/ $\perp \in (h \circ a)\overline{Rat} \Rightarrow (h \circ a)\overline{Rat} \setminus \{\perp\} \subseteq (h \circ f)\overline{Rat}$

$\Rightarrow (h \circ a)\overline{Rat} \setminus \{\perp\} \subseteq (h \circ g)\overline{Rat}$

i.e. $T \in (h \circ f)\overline{Rat} \Rightarrow T \in (h \circ g)\overline{Rat}$

rassemblant les deux conclusions il vient

$$f \sqsubseteq_M g.$$

(i) soit $\forall a \in \mathcal{N} \quad a \sqsubseteq_S f \Rightarrow a \sqsubseteq_S g$. Il vient

$$a \sqsubseteq_S f \iff \begin{cases} \forall h : D \to 2 \text{ continue} \\ \perp \in (h \circ a)\overline{Rat} \text{ ou } (h \circ f)\overline{Rat} \subseteq (h \circ a)\overline{Rat} \end{cases}$$

$\Downarrow$

$a \sqsubseteq_S g \iff \forall h' : D \to 2$ continue

$(h' \circ a)\overline{Rat}$ ou $(h \circ g)\overline{Rat} \subseteq (h \circ a)\overline{Rat}$

Il faut montrer

$\forall h \in [D \to 2]\ \perp \in (h \circ f)\overline{Rat}$ ou $(h \circ g)\overline{Rat} \subseteq (h \circ f)\overline{Rat}$

Supposons que c'est faux, i.e.

$\exists h \in [D \to 2]\ \perp \notin (h \circ f)\overline{Rat}$ et $(h \circ g)\overline{Rat} \not\subseteq (h \circ f)\overline{Rat}$

i.e. $(h \circ f)\overline{Rat} = \{T\}$ et $(h \circ g)\overline{Rat} = \{\perp, T\}$

1er cas : $\exists a \in \mathcal{N}$ $\quad a \sqsubseteq_S f$ et $\perp \notin (h \circ a)\overline{Rat}$
alors $(h \circ g)\overline{Rat} = (h \circ f)\overline{Rat} = \{T\}$ contradiction

2ème cas : $\forall a \in \mathcal{N}$ $\quad a \sqsubseteq_S f \implies \perp \in (h \circ a)\overline{Rat}$
c'est-à-dire

$\forall a \in \mathcal{N}$ $(h \circ f)\overline{Rat} \sqsubseteq (h \circ a)\overline{Rat} \implies \perp \in (h \circ a)\overline{Rat}$
Comme $(h \circ f)\overline{Rat} = \{T\}$ $\exists u \in f(\overline{Rat})$ $h(u) = T$
Considérons la fonction $a = \sqcup\{[x,u] : x \in N\}$
Comme N est fini $a \in \mathcal{N}$. De plus

$$\begin{aligned} \forall y \in N^\omega \quad a(y) = u \in f(\overline{Rat}) &\implies \\ a(\overline{Rat}) \sqsubseteq f(\overline{Rat}) &\implies \\ (h \circ f)\overline{Rat} = \{T\} = (h \circ a)\overline{Rat} & \end{aligned}$$

ce qui impliquerait $\perp \in (h \circ a)\overline{Rat}$ d'après l'hypothèse
$(\forall a \in \mathcal{N}$ $\quad a \sqsubseteq_S f \implies \perp \in (h \circ a)\overline{Rat})$ ce qui est absurde.

Donc le 2ème cas est contradictoire comme le premier, d'où la partie $\Rightarrow$ de l'assertion (i) . La partie $\Leftarrow$ est obtenue par transitivité. □

Maintenant il est facile de voir que le préordre $\sqsubseteq_S$ (resp. $\sqsubseteq_M$) définit presque une structure de faisceau sur $[N^\omega \to D]$, dont le noyau serait $\mathcal{N}$, par

$$s : f \mapsto \{a \in \mathcal{N} \ : a \sqsubseteq_S f\}$$
(resp. $\quad s : f \mapsto \{a \in \mathcal{N} \ : a \sqsubseteq_M f\}$).

Plus précisément si on note par

$$\mathrm{lub}(\{a \in \mathcal{N} : \ a \sqsubseteq_S f\}) = \mathrm{lub}(s(f))$$

(resp. etc...) l'ensemble des bornes supérieures de $s(f)$, alors on a :

3.34. PROPOSITION :

(i) $\forall f \in [N^\omega \to D]$ $\exists h \in \mathrm{lub}(s(f))$ $f = h$

(ii) Le quotient de l'espace $[N^\omega \to D]$ par la relation d'équivalence $\sim$ induite par le préordre $\sqsubseteq_S$ (resp. $\sqsubseteq_M$) fournit un fom sur $[N^\omega \to D]/_\sim$. Ce fom est élémentaire et a pour ensemble de rationnels $\mathcal{N}/_\sim$.

<u>3.35. COROLLAIRE</u> : L'espace $[N^{\omega} \to D]/_{\sim}$ est à un isomorphisme de poset près, le powerdomain au sens de Plotkin si $\sim$ est $=_M$ (resp. le powerdomain "faible" au sens de Smyth si $\sim$ est $=_S$). •

<u>DEMONSTRATION DE LA PROPOSITION</u> : D'après 3.17. on a : $\forall f \in [N^{\omega} \to D]$
$f = \sqcup \{a \in \mathcal{N} : a \sqsubseteq f\} = \sqcup \{f|_{V_n} : V_n = \{u \in N^{\omega} : \text{longueur } (u) \leq n\}\}$
Posons $f_n = f|_{V_n} \in \mathcal{N}$. Alors $\forall n\ f_n \sqsubseteq_S f$ (resp. $f_n \sqsubseteq_M f$) d'après 3.31. Soit maintenant $g \in [N^{\omega} \to D]$ tq $\forall n\ f_n \sqsubseteq_S g$ (resp. $f_n \sqsubseteq_M g$). Montrons $f \sqsubseteq_S g$ (resp. $f \sqsubseteq_M g$).

D'après la propriété de compacité il suffit que

$$\forall a \in \mathcal{N} \qquad a \sqsubseteq_S f \Rightarrow a \sqsubseteq_S g$$

(resp. $a \sqsubseteq_M f \Rightarrow a \sqsubseteq_M g$)

Soit $a \in \mathcal{N}$ $a \sqsubseteq_S f$ il existe V_m tel que $a = \bigsqcup_{q \leq n} [e_q, e'_q] \Rightarrow \forall q\ e_q \in V_m$
i.e. $a \sqsubseteq f|_{V_m} = f_m \sqsubseteq_S g$ (resp. $\sqsubseteq_M g$)

$\Rightarrow a \sqsubseteq_S g$ (resp. $a \sqsubseteq_M g$)

D'où $f \sqsubseteq_S g (f \sqsubseteq_M g)$. Donc f est bien une borne supérieure des f_n. Comme on vient de voir que les $a \sqsubseteq_S f$ ($a \sqsubseteq_M f$) sont finis pour la famille $\mathcal{F} = \{\{f_n\}_{n \in \mathbb{N}}\}$ on a bien que f est une borne supérieure pour l'ensemble $\{a \in \mathcal{N} : a \sqsubseteq_S f\} = s(f)$ (resp. $\{a \in \mathcal{N} : a \sqsubseteq_M f\} = s(f)$) □

Le reste de la proposition est évident.

Le corollaire est immédiat si on note 3.30 -

On peut noter encore que toute suite $\{g_n\}_{n \in \mathbb{N}}$ d'éléments de $\mathcal{N}$ $\sqsubseteq_S$-ascendante (resp. $\sqsubseteq_M$-ascendante) a une borne supérieure dans $[N^{\omega} \to D]$ qui est définie par $g : N^{\omega} \to D$

$\mathrm{Dom}(g) \cap \overline{\mathrm{Rat}} = \{u \in (\bigcap_n \mathrm{Dom}(g_n)) \cap \overline{\mathrm{Rat}}$: la suite g_n est $\sqsubseteq$-croissante le long de u, i.e. $\{g_n(u)\}_{n \in \mathbb{N}}$ est croissante dans $D\}$

$\mathrm{Dom}(g) \cap \mathrm{Rt}(N^{\omega}) = s(\mathrm{Dom}(g) \cap \overline{\mathrm{Rat}})$

$$g : x \mapsto \bigsqcup_m g_m(x).$$

Le passage au quotient fournit une structure de cpo. On a donc une opération

$$P : D \to [N^{\omega} \to D]/_{\sim}$$

dite "powerdomain".

Smyth [SMY] note que la classe des cpo tels que toute paire d'éléments majorée admet une borne supérieure est close pour $\mathcal{P}$ lorsque $\sim$ est $=_S$. Plotkin introduit dans le même but, lorsque $\sim$ est $=_M$, la classe des objets SFP qui sont des ω-limites inductives (au même sens où D_∞ est une limite inductive) de cpo's finis.

En résumé nous avons pris comme objets les $f \in [N^\omega \to D]$ (qui dénotent des calculs parallèles), et nous les avons reliés entre eux par les préordres d'Egli-Milner et de Smyth dans le but avoué d'avoir ainsi un cpo. La clause (i) de la proposition 3.34. nous fournit la meilleure approximation possible de la structure de faisceau qui respecte la structure des objets. La clause (ii) de 3.34., en passant aux classes d'équivalence de ces objets, permet de retrouver un faisceau et par là des objets connus de la théorie des cpo's, mais elle oblige à des identifications obscures pour le programmeur que nous avons mentionnés dans les prolégomènes.

On ne peut manquer de noter la parenté qui existe entre l'opération qui consiste à "mélanger" certains $f, g \in [N \to D]$ (ou moyen de la relation d'équivalence $=_M$ ou $=_S$) en se référant à leurs images et celle qui consiste à dire (cf. chap. I. pp. 59) que si $\lambda = \Upsilon \circ \mathrm{Im}$ avec $\Upsilon \in \{\sqcup, \sqcap\}$, i.e. si les limites sont prises par borne inférieure (borne supérieure), alors pour autant que la notion de convergence/non-convergence soit concernée, il revient au même de confondre plusieurs systèmes approximants distincts $\sigma : j \mapsto X_j = \sigma(j)$ à condition qu'ils aient la même image $\mathrm{Im}(\sigma) = \{X_j : j \in J\} \subseteq X$. Les fonctions $f \in [N^\omega \to D]$ peuvent donc être considérées sous cet égard comme des systèmes approximants (ce qu'ils sont en réalité, puisque nous avons montré au Chapitre I de quelle manière ils expriment des suites de calcul, ou de réductions dans le λ-calcul, etc.), dont il s'agit de trouver le faisceau associé.

Ceci indique que la solution serait de sortir de la vision habituelle des cpo's comme domaines de calcul et / ou trouver une autre construction qui respecte mieux les objets informatiques. Certains auteurs (Smyth [PLS], Plotkin [1978], ...) préconisent l'emploi de <u>catégories</u>.

Il nous semble quant à nous que même dans ce cadre la notion de convergence joue un grand rôle (cf. la notion de faisceau sagittal au Chapitre IV), et que l'on y repose le même problème. Dans une direction différente la notion de système approximant (ou calcul) a été aussi remise en relief par Arnold et Nivat, qui utilisent une métrique pour relier ces systèmes approximants entre eux.

En fin de compte on peut dire que, fondamentalement, le préordre d'Egli-Milner (resp. de Smyth) est la notion qu'il faut rajouter aux faisceaux ordonnés et aux fonctions régulières pour obtenir le powerdomain au sens de Plotkin (resp. le powerdomain faible au sens de Smyth). Et ces préordres sont là pour relier les systèmes approximants entre eux de la même manière que la métrique d'Arnold et Nivat est là pour relier les calculs infinis non-déterministes entre eux par passage à la KP-convergence associée.

IV - SEMANTIQUE D'UN LANGAGE DE PROGRAMMATION AVEC DECLARATION DE TYPES ; RETRACTES

Nous avons vu par exemple que tous les cpo sont des faisceaux et les fonctions continues des fonctions régulières, par conséquent toute la sémantique utilisant ces objets peut être décrite dans le cadre des faisceaux. Nous avons maintenant suffisamment de matériel pour nous attaquer à un autre problème de la sémantique des programmes : les déclarations de type.

La notion de type a bénéficiée ces dernières années d'un regain d'intérêt et d'une floraison de recherches : travaux de Guttag [GUT] , langages CLU [LIS] , Alphard [WJL] , Russell [DEM] , FOL [WEY] ... après avoir été pendant une certaine période surtout l'objet de l'intérêt des praticiens. Etait-ce dû au manque d'un véritable modèle (au sens de la théorie des modèles, i.e. dénotationnel) comportant des types en tant qu'objets de calcul ? En tout cas la liaison entre la notion de rétracte (au sens de Scott) et la notion de type au sens usuel restait un domaine de recherches. Il est caractéristique que les modèles présentés par L. Nolin [NOL] et

J. Donahue [DON] , surtout orientés vers la programmation, ne comportaient pas de construction mathématique complète.

L'une des occurences initiales de la notion de type en informatique se situe au niveau du codage binaire des nombres : si à chaque nombre on associe un nombre fini de bits, alors si on veut manipuler des nombres suffisamment grands on est obligé de choisir des représentations différentes pour les nombres entiers ($\in \mathbb{N}$) et pour les nombres "réels" ($\in \mathbb{Q}-\mathbb{N}$), et donc une arithmétique différente pour chacune de ces classes de nombres, donc un type différent et ceci doit apparaître au niveau de la programmation.

Nous avons montré (cf. Prolégomènes) comment ces "fourches caudines" de l'informatique ont été perçues par L. Nolin comme un concept fondamental.

La notion de type est ignorée ou desservie dans plusieurs langages de programmation. Ainsi en LISP (cf. Allen 1978) les types sont assignés dynamiquement et les erreurs de types ne sont détectés que lorsqu'il est trop tard pour faire quoi que ce soit.

Nous donnerons ici au mot type la signification suivante : un type est un objet de calcul, dénotant une valeur, au même titre qu'un programme dénote une valeur (fonctionnelle) (Nolin englobe toutes ces valeurs sous le nom générique d'algorithmes), et un même élément peut avoir plusieurs types. Ainsi en FORTRAN, l'élément 2 a le type integer, mais aussi en toute rigueur tous les types $[1,n] \subseteq \mathbb{N}$, $n \in \mathbb{N}$ puisqu'on a une notion de tableau dans le langage et que les indices des éléments de ces tableaux doivent se trouver entre les bornes déclarées. L'élément 2 peut être considéré comme le type $\{2\}$, et $\{2\}$ a en même temps le type de tous les types qui le contiennent (i.e. un ensemble est approximé par tous les ensembles qui le contiennent).

IV. 1. LE LANGAGE DE PROGRAMMATION

Nous n'avons pas l'intention de fournir ici une description complète d'un langage de programmation, nous voulons simplement illustrer l'utilisation des faisceaux (ordonnés) pour traiter la notion de type et la différence de notre approche avec celle de Scott par exemple.

Nous définissons d'abord un langage $\mathcal{L}_t$ de types au sens de la logique combinatoire (cf. [HIN] pp. 66 sqq) :

(i) on a un certain nombre de types de base : $\alpha, \beta, \dots$ dont nous ne préciserons pas la syntaxe.

(ii) si α, β sont des types alors $F \alpha \beta$ est un type

(iii) si $\{\alpha_i\}_{i \in I}$, $\{\beta_i\}_{i \in I}$ sont des familles de types, alors $\bigwedge_{i \in I} F \alpha_i \beta_i$ est un type. □

Nous interpréterons les types de base $\alpha, \beta, \dots$ comme des éléments a,b,... de quelque faisceau. Les types $F \alpha \beta$ correspondront aux échelons de Nolin $\langle a,b \rangle$, où a interprète α et b interprète β.

Nous nous intéressons aux programmes avec déclaration de la forme suivante

$$\bigwedge_{i \in I} F \alpha_i \beta_i : G(x) \Leftarrow \tau[G](x) \qquad (1)$$

où $\tau[G]$ est un terme obtenu en composant un nombre fini de fois des fonctions de base, des prédicats de base et la variable fonctionnelle G.

La signification intuitive qu'il convient d'accorder au programme (1) en s'appuyant sur ce qu'on fait dans des langages comme Algol et Fortran, peut être esquissée de la façon suivante : si on a :

$$F \alpha \beta : G(X) \Leftarrow \tau[G](x) \qquad (2)$$

ceci signifie : $\forall$ donnée x, si x vérifie "x est du type α" et si le programme G s'arrête pour la donnée x, alors G(x) vérifie "G(x) est du type β".

Le lecteur ne peut manquer de noter l'analogie de cette assertion avec, par exemple, l'assertion de la logique de Hoare suivante

$$\alpha \{G\} \beta \tag{3}$$

qui signifie : "Pour tout état de mémoire donné x, si x vérifie la précondition α et si le programme G s'arrête pour l'état x, alors l'état final $G(X)$ vérifie la post-condition β".

L'expression (2), comme l'expression (3), décrivent une certaine <u>approximation de la fonction calculée par le programme</u> G : l'expression (3) de façon "prédicative" (i.e. au moyen des assertions α et β), l'expression (2) de façon "typique" (i.e. au moyen des types α et β). (On voit donc que sur le plan théorique, la programmation avec déclarations de type n'est pas fondamentalement différente de la programmation avec assertions).

Donc en fin de compte la construction d'une sémantique des programmes avec déclarations de types, comme celle pour les programmes avec assertions (i.e. manipulant une sémantique exiomatique au lieu d'une sémantique "typique") revient à construire* entre deux descriptions de la fonction calculée par le programme G : sa description algorithmique, donnée par le texte de G, et sa description plus ou moins approximative donnée par la déclaration de types (respectivement la précondition et la postcondition).

Cette construction est relativement aisée dans le second cas, que nous avons considéré dans Nait-Abdallah, 1976. Elle requiert dans le premier cas, que nous examinons ici, des instruments plus complexes et plus fins, peut-être du fait que la notion de type est beaucoup plus proche de la machine que la notion de prédicat.

Nous retrouvons une notion (généralisée) étudiée dans Scott : celle de <u>rétracte</u>, mais nous en faisons une utilisation différente.

Nous opérons comme dans la sémantique usuelle du point fixe STO, VUI , prenant les extensions régulières des fonctions (cf. Prop. 3.5.) au lieu de leurs extensions régulières. Ainsi pour le <u>conditionnel de Mc Carthy</u> :

$$\lambda p\, a\, b.\, (p \to a, b) : \begin{array}{l} (\text{vrai}, a, b) \mapsto a \\ (\text{faux}, a, b) \mapsto b \\ (\text{booléen}, a, b) \to a \sqcup b \\ (\bot, a, b) \mapsto \bot \\ (\top, a, b) \mapsto \top \end{array}$$

(*) un modele permettant de verifier le recollement

La 3ème ligne se justifie ainsi : savoir que p a une valeur booléenne, c'est plus que n'en savoir rien du tout et comme $p \in \{vrai, faux\}$ il y a autant de raisons de choisir vrai que faux, d'où le résultat non-déterministe $a \sqcup b$. Nous avons supposé implicitement ici que l'espace X est un sup-semilattice, mais nous précisons ce point par la suite.

<u>3.36. Fait</u> : Le conditionnel de Mc Carthy jouit des propriétés suivantes :

(i) $(vrai \to a, b) = a$

(ii) $(faux \to a, b) = b$

(iii) $(p \to (p \to a, b), c) = (p \to a, c)$
si et seulement si $b \sqsubseteq a \sqcup c$

(iv) $(p \to a, (p \to b, c)) = (p \to a, c)$
si et seulement si $b \sqsubseteq a \sqcup c$

(v) $((p \to q, r) \to a, b) = (p \to (q \to a, b), (r \to a, b))$
si et seulement si $a = a \sqcup b = \top$

(vi) si $f_1(x) = g(p(x) \to h_1(x), h_2(x))$
et $f_2(x) = (p(x) \to g(h_1(x)), g(h_2(x)))$

où les fonctions g, p, h_1, h_2 sont régulières, alors $f_2 \sqsubseteq f_1$. Si de plus on a $g(\perp) = \perp$, $g(\top) = \top$ et

$$g(h_1(x) \sqcup h_2(x)) = g(h_1(x)) \sqcup g(h_2(x))$$

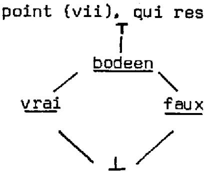
alors $f_1 = f_2$.

(vii) la fonction ψ
$x \mapsto (p(x) \to a(x), b(x))$

où a, b, p sont régulières, n'est généralement pas régulière. □

<u>DEMONSTRATION</u> : immédiat. Le seul fait qui mérite peut-être d'être explicité est le point (vii), qui ressort du contre-exemple suivant

Soit B =



Soit X un foe canonique, qui est un supsemillatice possédant $\top$ et $\perp$, contenant $\mathbb{N}$.

$p : X \to B$, $a : X \to X$, $b : X \to X$ régulières tq

$p(1) = \text{vrai}$ $\quad p(2) = \text{faux}$ $\quad p(\underline{12}) = \text{booléen}$

$a(1) = 0$ $\quad a(2) = 1$ $\quad a(\underline{12}) = a(1) \sqcup a(2) = \{0,1\}$

$b(1) = 3$ $\quad b(2) = 4$ $\quad b(\underline{12}) = b(1) \sqcup b(2) = \{3,4\}$

$$\psi(12) = (p(12) \to a(\underline{12}),\ b(\underline{12})) =$$
$$(\text{booleen} \to a(\underline{12}),\ b(\underline{12})) = a(\underline{12}) \sqcup b(\underline{12}) = \{0,1,3,4\}$$

$$\psi(1) \sqcup \psi(2) = (p(1) \to a(1),\ b(1)) \sqcup (p(2) \to a(2), b(2))$$
$$= (\text{vrai} \to a(1),\ b(1)) \sqcup (\text{faux} \to a(2),\ b(2)) = a(1) \sqcup b(2) = \{0,4\}$$

donc $\psi(\underline{12}) \neq \psi(1) \sqcup \psi(2)$ i.e. n'est pas régulière. □

Le seul point délicat est que, pour respecter l'intuition que l'on a des fonctions (cf. [SHA], Proleg. et supra §II), il conviendra de prendre ici non pas la composition usuelle des fonctions mais leur composition régulière. On associe ainsi une fonctionnelle

$$\tau : g \to \tau(g) \qquad \forall g \in [X \to X]$$

au terme fonctionnel $\tau[G]$

3.37. <u>PROPOSITION</u> : Si X est complet sous condition et les $[X^p \to X]$ munis de fom's élémentaires fidèles, alors la fonctionnelle τ associée au $\tau[G]$ est régulière sur le noyau de $[X \to X]$.

<u>DEMONSTRATION</u> : par induction sur la structure de $\tau[G]$.

(i) la fonctionnelle identité est régulière (1.2(i))

(ii) la fonctionnelle constante est régulière (1.2(ii))

(iii) si $\tau_1, \dots, \tau_n$ sont régulières sur $[X \to X]$ et $f \in [X^n \to X]$ régulière, alors

$$\tau : g \mapsto O(< \tau_1(g), \dots, \tau_n(g) >, f)$$

est régulière d'après la régularité de O et la fidélité de $[X \to X]$ (Proposition 3.8. (v)), puisque la fonction

$$< \ > : g \to < \tau_1(g), \dots, \tau_n(g)$$

est régulière : $\quad \forall \sigma \in \beta(g)$

$\sqcup < \ >(\sigma) = \sqcup < \tau_1(\sigma), \dots, \tau_n(\sigma) > =$
$< \sqcup \tau_1(\sigma), \dots, \sqcup \tau_n(\sigma) > = < \tau_1(g), \dots, \tau_n(g) > = < \ >(g)$.

(iv) si $\tau_1, \dots, \tau_n$ sont régulières, alors

$$\tau : g \to O(< \tau_1(g), \dots, \tau_n(g) >, g)$$

est régulière (même chose qu'en (iii)). □

Maintenant va intervenir la structure du faisceau X.

IV. 2. POINTS FIXES DE FONCTIONNELLES

A priori il y a plusieurs façons d'obtenir un point fixe pour la fonctionnelle τ. On peut par exemple généraliser le théorème de Kleene pour les cpo's plats. Si X est un fom élémentaire et si on ne considère que l'espace $[X \to X]_m$ des fonctions régulières possédant un minorant approximant, alors il existe sur $[X \to X]_m$ un fom de la convergence simple (Cor. 3.14) i.e. $f_i < f \iff \forall x \; f_i(x) < f(x)$.
Ce faisceau est fidèle, et comme τ est régulière sur $[X \to X]_m$ (Prop. 3.8. (vi))(1) :

$$\forall x \in X \quad \tau(\lim_i f_i)(X) = (\lim_i \tau(f_i))(x) = \lim_i \tau(f_i)(x)$$

Si on considère maintenant une suite

$$\forall i \in \mathbb{N} \quad \tau^i(f_o) = f_i \quad , \quad f_o \in [X \to X]_m$$

Comme $[X \to X]_m$ est stable par τ, on peut supposer la suite $\{f_i\}_{i \in \mathbb{N}}$ convergente sur quelque domaine $D \subseteq X$ (i.e. $\forall x \in D, \sqcup \{f_i(x) : i \in \mathbb{N}\}$ existe); et sur D :

$$\tau(\lim_i f_i) = \tau(\lim_i \tau^i(f_o)) = \lim_i \tau^{i+1}(f_o)$$

La limite $\lim_i f_i$ est solution de l'équation

$$\exists ? \, g \qquad g = \tau(g)$$

sur le domaine D si et seulement si

$$\lim_i \tau^i(f_o) = \lim_i \tau^{i+1}(f_o)$$

(1) à condition que X soit complet sous condition.

Comme τ est monotone (Lemme 3.1.), il suffit que

$$\lim_i \tau^{i+1}(f_o) \subseteq \lim_i \tau^i(f_o)$$

ce qui est évident puisque lim = $\sqcup$. Donc on peut énoncer :

<u>3.38. PROPOSITION</u> : Si X est un poset complet sous condition muni d'un fom élémentaire, $[X \to X]_m$ l'espace des fonctions régulières sur X possédant un minorant approximant, muni du fom de la convergence simple, alors toute suite $\{\tau^n(f_o)\}_{n \in \mathbb{N}}$ pour $f_o \in [X \to X]_m$, convergente sur un domaine $D \subseteq X$ (i.e. tq $\forall x \in D$ $\bigsqcup_n \tau^n(f_o(x)$ existe), est point fixe sur D de l'équation fonctionnelle $g = \tau(g)$. En particulier si $\perp \in X$, $\bigsqcup_n \tau^n(\lambda x.\perp)$ est la plus petite solution sur D de $g = \tau(g)$.

<u>DEMONSTRATION</u> : cf. supra.

Mais ceci n'est qu'une façon d'utiliser la régularité de τ (prop. 3.37), dont en fait la seule exigence est la fidélité des fom élémentaires dont sont munis les espaces fonctionnels. En particulier la proposition 3.37. nous paraît se relier, en un certain sens, (que nous n'avons pas la place de développer ici) à plusieurs résultats de Nivat sur les grammaires algébriques [NIV'77] , où cet auteur énonce pour une "fonctionnelle" analogue à τ plusieurs propriétés de régularité lorsque l'on considère les faisceaux associés aux topologies supérieures et inférieures de Scott.

Ainsi $[X \to X]$ peut être muni du faisceau associé à la topologie inférieure de Scott :

$$\forall f \in [X \to X] \quad \mathfrak{F}(f) = \{S \text{ chaîne descendante } \subseteq [X \to X] \text{ et } f = \sqcap S\}$$

Ce faisceau est fidèle (trivialement); s'il définit un fom élémentaire (i.e. algébrique), alors τ est régulière dessus; on peut donc énoncer :

3.39. PROPOSITION : Si X est un faisceau ordonné, $[X \to X]$ l'espace des fonctions régulières de X dans X muni du faisceau associé à la topologie supérieure de Scott.

$\beta : f \mapsto \{S \subseteq [X \to X] : S$ descendante et $f = \sqcap S\}$, tel que $[X \to X]$ est élémentaire, alors toute suite $\{\tau^n(f_0)\}_{n \in \mathbb{N}}$, pour $f_0 \in [X \to X]$, convergente sur un domaine $D \subseteq X$ (i.e. tq $\forall x \in D$ $\underset{n}{\sqcap}\, \tau^n(f_0)(X)$ existe), est point fixe sur D de l'équation fonctionnelle $g = \tau(g)$. En particulier si $T \in D$, $\underset{n}{\sqcap}\, \tau^n(\lambda x.T)$ est le plus grand point fixe de $\tau(g) = g$ sur X.

DEMONSTRATION : immédiat (analogue à 3.37).

(Nous avons énoncé un résultat analogue dans [NAI 78] en nous appuyant sur un argument inexact. Nous croyions en effet que sur un foe dont toute partie majorée possède une borne supérieure toute fonction régulière est $\sqcap$-continue. Ceci n'est pas vrai : il suffit de considérer $X = \mathcal{P}(\mathbb{N})$ ordonné par inclusion et $f : X \to X$ définie sur les rationnels par :

$f(\phi) = \phi$ et $\forall n \in \mathbb{N}$ $f(\{n\}) = \{1\}$. Alors

$\phi = f(\phi) = f(\underset{p}{\sqcap}\,[p,+\infty]) \neq \underset{p}{\sqcap}\, f([p,+\infty]) = \{1\}$)

Nous laissons au lecteur le soin de formuler d'autres "théorèmes du point fixe" en prenant d'autres faisceaux fonctionnels fidèles.

IV. 3. SEMANTIQUE DES PROGRAMMES

Si le programme ne contenait pas de déclarations de types, les énoncés du paragraphe précédent suffiraient à fournir une sémantique. Mais nous devons tenir compte du "filtrage" associé au niveau de l'ordinateur à la déclaration de type, et donc la fonctionnelle qui nous intéresse n'est pas τ mais sa transformée par ce "filtrage".

La compléxité du domaine de calcul X dépend de la complexité des déclarations de types que l'on s'autorise au niveau du langage. Nous avons vu que pour bien interpréter le conditionnel de Mc Carthy,

X devait en tout cas être un sup-semillatice. Supposons qu'on ne considère que des programmes de la forme

$$F\ \alpha\ \beta \quad : G(x) \Leftarrow \tau[G](x).$$

avec la signification (Algol, Fortran,...) : si x est du type α procéder au calcul et vérifier que $G(x)$ a le type β, sinon (délivrer soit la valeur indéfinie $\top$ soit la valeur β).
Alors on peut procéder comme suit :

<u>DEFINITION</u> : Un <u>domaine commode</u> X est un faisceau ordonné élémentaire tel que :

(i) X est un lattice possédant $\top$

(ii) $\forall a \in X,\ \forall S \subseteq X$ tel que $\sqcup S$ existe

$$a \sqcap (\sqcup\{s : s \in S\}) = \sqcup\{a \sqcap s : s \in S\} .$$

On note que si X est un treillis complet, la clause (ii) revient à dire que X est une <u>algèbre de Heyting complète</u> (cf.[SCS]).

Dans un domaine commode, la fonction

$$x \mapsto a \sqcap x$$

est régulière pour tout a. Si X est un domaine commode, et si τ est une fonctionnelle sur $[X \to X]$, $\forall a,b \in X$ on définit sa transformée $N_{a,b}(\tau)$ par :

$$N_{a,b} : \tau \mapsto \lambda f.\lambda x.\ \langle a, b \sqcap \tau(f)(x) \rangle_{\top}(x)$$

(on peut aussi définir, suivant notre convention :

$$N^{*}_{a,b} : \tau \mapsto \lambda f.\lambda x.\langle a, \tau(f)(x) \rangle_{b}\ (x))$$

où $\langle \ldots \rangle$ dénote l'échelon de Nolin.

Cette transformation traduit le "filtrage" de la fonctionnelle τ.

<u>3.40. PROPOSITION</u> : Si $[X \to X]$ est muni d'un faisceau fidèle, alors

(i) la transformée $N_{a,b}(\tau)$ (resp. $N^*_{a,b}(\tau)$) d'une fonctionnelle régulière est régulière.

(ii) restreinte à l'espace des fonctionnelles régulières muni d'un faisceau fidèle la transformation est régulière. □

<u>DEMONSTRATION</u> : (i) résulte du fait que l'espace des images des fonctions considérées possède la propriété de commodité (ii):

car soit $\tau' = N_{a,b}(\tau)$, $f \in [X \to X]$, $\sigma \in \beta(f)$, alors :

$\tau'(f) = \lambda x. < a,b \sqcap \tau(\sqcup\sigma)(x) >_T (x)$ = (τ régulière)

$\lambda x. < a,b \sqcap (\sqcup \{ \tau(h) : h \in \sigma \})(x) >_T (x)$ = (fidélité) =

$\lambda x. < a,b \sqcap (\sqcup \{ \tau(h)\ (x) : h \in \sigma \}) >_T (x)$ = (propriété de commodité de x)

$\lambda x. < a, \sqcup \{b \sqcap \tau(h)(x) : h \in \sigma \} >_T (x) =$

$\sqcup \{ \lambda x. < a,b \sqcap \tau(h)(x) >_T (x) : h \in \sigma \} = \sqcup \tau'(\sigma)$

d'où la régularité de $\tau' = N_{a,b}(\tau)$.

Même argument pour $N^*_{a,b}(\tau)$.

(ii) on a de même pour $N_{a,b}$:

$N_{a,b}(\tau) = \lambda f. \lambda x. < a,b \sqcap \tau(f)(x) >_T (x)$ = (si $\sigma \in \beta(\tau)$)

$= \lambda f. \lambda x. < a,b \sqcap ((\sqcup\sigma)f)(x) >_T (x) =$

$\lambda f. \lambda x. < a,b \sqcap (\sqcup \{ \theta(f) : \theta \in \sigma \})(x) >_T (x) =$

$\lambda f. \lambda x. < a,b \sqcap (\sqcup \{ \theta(f)(x) : \theta \in \sigma \}) >_T (x) =$

$\lambda f. \lambda x. < a, \sqcup \{b \sqcap \theta(f)(x) : \theta \in \sigma \} >_T (x) =$

$\sqcup \{ \lambda f. \lambda x. < a,b \sqcap \theta(f)(x) >_T (x) : \theta \in \sigma \} = \sqcup N_{a,b}(\sigma)$.

Même chose pour $N^*_{a,b}$. □

Si X est un domaine commode et τ régulière, $N_{a,b}(\tau)$ $(N^*_{a,b}(\tau))$ est aussi régulière; on peut donc appliquer 3.39 <u>si $[X \to X]$ est élémentaire</u>. Nous dirons alors suivant [NOL] que la fonction calculée par le programme :

$$F \alpha\ \beta\ : G(x) \Leftarrow \tau[G](x)$$

a interprétant α et b interpétant β, est par définition le plus grand point fixe de la fonctionnelle régulière $\tau' = N_{a,b}(\tau)$ (resp.

$\tau' = N^{*}_{a,b}(\tau)$). (Le lecteur peut se référer à Naït-Abdallah, 1978 pour plus de détails sur le calcul de ce point fixe). Ceci exprime de façon claire ce qui était entrevu de façon empirique dans Shamir et Wadge, 1977.

Si l'on considère maintenant des programmes comportant des déclarations plus complexes :

$$\bigwedge_{i \in I} F\alpha_i\, \beta_i \; : \; G(x) \Leftarrow \tau[G](x)$$

on voit qu'il est commode d'interpréter les types $F\alpha_i\beta_i$ comme les échelons de Nolin $<a_i,b_i>_T$ où α_i (resp. β_i) dénote a_i (resp. b_i). On donnera à la déclaration ci-dessus la signification intuitive suivante :

si x est du type α_i ,procéder au calcul et vérifier que $G(x)$ a le type β_i , et ce pour tout $i \in I$, sinon (i.e. $\nexists\, \alpha_i$ tq x est du type α_i) (délivrer soit la valeur T soit la valeur qui a tous les types β_i).

Si $u = \bigwedge_{i \in I} F\alpha_i\, \beta_i$, on définit la transformation

$$N_u(\tau) = \tau' \quad \text{avec} \quad \forall f$$

$$N_u(\tau)(f)(x) = \tau(f)(x) \sqcap (\textstyle\sqcap_i <a_i,b_i>_T(x)) \quad \text{si} \quad \exists\, a_i \quad x \sqsubseteq a_i$$

$$= T \quad \text{sinon}$$

et la fonction calculée sera le plus grand point fixe de $N_u(\tau)$ (resp. de $N^{*}_u(\tau)$ définie de façon analogue).

IV - 4. RETRACTES CORRESPONDANT AUX DECLARATIONS DE TYPES

DEFINITION : (i) Soit X un faisceau. Une rétraction sur X est une fonction $r: X \to X$ telle que r est régulière et $r \circ r = f$.

(ii) si r est une rétraction, le rétracte déterminé par r est l'image de r :

$$Im(r) = \{x \in X : x = r(x)\}.$$

□

3.41. Fait : Si r est une rétraction alors $ror = o(r,r)$. □

DEMONSTRATION : immédiat. □

Si r est une rétraction, r régulière $\Rightarrow \forall x \in X$,
$\forall \sigma \in \beta(x)$ on a : $r(x) = r(\sqcup\{s : s \in \sigma\}) = \sqcup\{r(s) : s \in \sigma\}$.
Donc les images des fibres donnent une structure de faisceau induite sur le rétracte déterminé par r.

3.42. LEMME : Soit X un faisceau, Y un domaine commode, $f \in [X \to X]$. Définissons $\forall a \in X$, $b \in Y$ les transformations :

$$N_{a,b}(\tau) = \lambda f.\lambda x.< a,b \sqcap \tau(f)(x) >_T(x)$$

$$N^*_{a,b}(\tau) = \lambda f.\ x.< a,\tau(f)(x) >_b(x)$$

alors (i) $N_{a,b}$ et $N^*_{a,b}$ sont idempotentes

(ii) si $[X \to Y]$ et $[(X \to Y] \to [X \to Y]]$ sont munis de faisceaux fidèles, alors $N_{a,b}$ et $N^*_{a,b}$ sont des rétractions.

DEMONSTRATION : (i) $\forall f \in [X \to Y]$, $\forall x \in X$ $(N_{a,b}(N_{a,b}(\tau)))(f,x) =$
$< a,b \sqcap (N_{a,b}(\tau))(f)(x) >_T (x) =$
$< a,b \sqcap < a,b \sqcap \tau(f)(x) >_T(x) >_T (x) =$ (si $x \sqsubseteq a$ alors $b \sqcap \tau(f)(x)$ sinon T) $= < a,b \sqcap \tau(f) >_T (x) = N_{a,b}(\tau)(f,x)$
i.e. $N_{a,b} \circ N_{a,b} = N_{a,b}$ - d'où l'idempotence de $N_{a,b}$.

Même raisonnement pour $N^*_{a,b}$

(ii) résulte de (i) et proposition 3.40. □

Pour voir le lien avec les travaux de Scott (cf. Scott, 1979, Swansea) on peut voir aussi que :

3.43. LEMME : $\forall a \in X$, $\forall b \in Y$ $\nexists r : X \to X$, $t : Y \to Y$, r et t rétractions telles que $\forall f \in [X \to Y]$ $N_{a,b}(f) = tofor$. □

<u>DEMONSTRATION</u> : soit $r : X \to X$ une rétraction, $\mathcal{U} = r(X)$ le rétracte déterminé par r.

$\forall u \in r(X) \quad u = r(v)\ ,\ \ v \in X$

$r(u) = r(r(v)) = r(v) = u$, donc r est l'identité sur

$\mathcal{U} = r(X)$.

Supposons qu'il existe deux rétractions r, t telles que $N_{a,b}(f) = tofor$. Alors

$\forall x \in r(X) = \mathcal{U} \quad t(f(x)) = \langle a, b \sqcap f(x) \rangle_T (x) =$ si $x \sqsubseteq a$ alors

$b \sqcap f(x)$ sinon T

sur $(\downarrow a) \cap \mathcal{U}$ on a $tof = \lambda x. b \sqcap f(x)$

sur $(X - \downarrow a) \cap \mathcal{U}$ $tof = \lambda x. T$

Donc sur $f((X \setminus \downarrow a) \cap \mathcal{U})$ $t = \lambda x. T$

sur $f(\mathcal{U} \cap \downarrow a)$ $t = \lambda y. b \sqcap y$

on voit que la définition de t dépend étroitement de la définition de f, et pas seulement de a et b. D'où le lemme.

On peut énoncer un fait analogue à 3.43. pour $N^*_{a,b}$:

$$g = N^*_{a,b}(f) = \lambda x. \langle a, f(x) \rangle_b \ .$$

Dans cette approche, l'échelon $\langle a,b \rangle$ représente la classe des fonctions qui prenant un argument de type a rendent un résultat de type a. ou si l'on préfère le type Fab de la logique combinatoire, et l'ordre que nous prenons est celui de $[X \to Y]$. Pour travailler sur les fonctionnelles nous leurs associons des transformations $N_{a,b}$. Ces objets sont clairement distincts des rétractes :

$(a \quad o\!\!\longrightarrow b) = \lambda f.\ b \circ f \circ a$

considérés par D. Scott [SCO 76] munis de l'ordre $\leqslant$:

$a \leqslant b \iff a \circ b = b \circ a = a$

ce qui conduit au résultat

$a \leqslant a'$ et $b \leqslant b' \Longrightarrow (a \quad o\!\!\longrightarrow b) \leqslant (a' \quad o\!\!\longrightarrow b')$

contradictoire avec les propriétés des échelons de Nolin. qui donnent (à l'indice près que l'on peut prendre = T) :

$a \sqsubseteq a'$ et $b' \sqsubseteq b \Longrightarrow \langle a,b \rangle \sqsubseteq \langle a',b' \rangle$

Nous aboutissons à une interprétation de la déclaration de type dans un en-tête de procédure qui n'est pas celle de Scott au moyen du rétracte $a \quad o\!\!\longrightarrow b$, mais qui nous paraît plus proche des objets manipulés par le programmeur dans des langages comme Algol ou Fortran.

CHAPITRE IV

FAISCEAUX SAGITTAUX ET LIMITES.

Nous abordons dans ce chapitre la méthode utilisée dans Hofmann 1979, Smyth-Plotkin 1978, pour calculer des espaces (sémantiques pour nous) par approximations successives et nous montrons comment cette méthode se relie très naturellement à la notion de faisceau que nous étudions ici. Il s'agit en fait de la méthode utilisée par D. Scott 1969 pour la construction de D_∞, reprise par la suite, dont nous montrons ici l'articulation avec la notion d'approximation que nous avons relevée pour $P\omega$, les espaces d'algorithmes de Nolin, etc... et qui correspond à une réalité informatique.

Nous esquissons une notion de "domaine universel" dans lequel résoudre des équations aux domaines (informatiques) : il s'agit de la catégorie des faisceaux munie de diverses notions d'approximation d'un faisceau par des diagrammes de faisceaux.

Nous montrons ensuite comment les méthodes des chapitres précédent peuvent s'appliquer pour construire des espaces "riches" au sens qu'ils vérifient non seulement des équations aux domaines mais expriment des notions très proches de la programmation usuelle : nous nous occuperons ici de la notion de <u>type</u>, déja abordée au chapitre précédent. On supposera que tous les faisceaux considérés sont éléments de quelque univers $\mathcal{U}$.(cf. infra)

I. <u>Les catégories de faisceaux</u> : <u>Fais</u>, <u>Foen</u>,...

Nous posons d'abord une définition [HOF 2] :

<u>Définition</u> : <u>fonction fortement régulière</u>.

Soient X,Y deux faisceaux. Une fonction $f : X \quad Y$ est <u>fortement régulière</u> ssi f est régulière et $\forall x \in X \ \forall \sigma \in \beta(x) \ (f \circ \sigma) \in \beta(f(x))$

4.1. <u>Fait</u> (i) La classe des faisceaux munie des applications fortement régulières forme une catégorie, appelée <u>Fais</u>.

(ii) <u>Fais</u> possède des produits et des coproduits

<u>démonstration</u> : (i) immédiat

(iii) il suffit de montrer que les flèches associées sont fortement régulières. Ce qui découle du lemme 1.3 et des définitions du chapitre I.

4.2 <u>Lemme</u> : La catégorie <u>Fais</u> possède des équaliseurs et des co-équaliseurs.

<u>démonstration</u> : (i) Equaliseurs :

Soient X, Y deux faisceaux et

$$X \underset{g}{\overset{f}{\rightrightarrows}} Y \quad \text{avec } f,g \text{ fortement régulières}$$

Soit $Z^* = \{x \in X : f(x) = g(x)\}$. Prenons pour Z le plus grand sous-ensemble E de Z^* tel que la trace sur Z du faisceau de X définit un faisceau sur E et l'injection $i : E \to X$ commute avec les limites i.e. $\forall \sigma \in \mathcal{F}(E)$ on a

$$(i \circ \lim_E)(\sigma) = (\lim_X \circ i)(\sigma)$$

Alors l'injection $e : Z \to X$ est fortement régulière et

$Z \xrightarrow{e} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ fournit un équaliseur de f,g. Car soit $W \xrightarrow{h} X \underset{g}{\overset{f}{\rightrightarrows}} Y$ tel que $f \circ h = g \circ h$. On a $h(W) \subseteq X$.

Comme h est fortement régulière h(W) est un sous-faisceau de X et $h(W) \subseteq Z^* \Rightarrow h(W) \subseteq Z$ en tant que sous-faisceau.

D'où une flèche unique $k : W \to Z$ tq le diagramme

$$\begin{tikzcd} W \arrow[r, "k"] \arrow[rr, bend right, "h"'] & Z \arrow[r, "e"] & X \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Y \end{tikzcd} \quad \text{commute}$$

(ii) coéqualiseurs : Dans les mêmes conditions soit

$$X \underset{g}{\overset{f}{\rightrightarrows}} Y$$

Soit $\sim$ la relation d'équivalence la plus grossière sur Y tq $\forall a \in X$ $f(a) \sim g(a)$. Posons $Z = Y/\sim$. La projection $e : Y \to Y/\sim = Z$ est un équaliseur de f,g dans Ens. Il suffit maintenant de munir Z du faisceau quotient car $\sim$ est régulière :

$$
\begin{tikzcd}
 & \sigma \in \beta(x) \arrow[dl, "f"'] \arrow[dr, "g"] & \\
(f \circ \sigma) \in \beta(f(x)) & & (g \circ \sigma) \in \beta(g(x))
\end{tikzcd}
$$

Par définition on aura $(f \circ \sigma) \sim (g \circ \sigma)$ (cf. chap. I.); prenant la limite $\lim(f \circ \sigma) = f(x)$

$$\lim(g \circ \sigma) = g(x)$$

et $f(x) \sim g(x)$ par définition de $\sim$.

On sait, d'après les travaux de Freyd [FRE] que les équaliseurs et produits (resp. les coéqualiseurs et coproduits) suffisent à calculer les limites (resp les colimites) dans les catégories. Donc Fais est à la fois complète et co-complète ([MCL] pp 105 sqq.)

Mais la catégorie Fais est trop vaste pour des applications pratiques. Nous définissons quelques catégories plus petites :

1. Foen : a pour objets les posets complets sous condition munis de f.o.m. élémentaires, et pour flèches les fonctions simples (monotones croissantes) composées de façon régulière.
2. Ups : la sous catégorie pleine de Fais donne les objets sont des posets munis du faisceau associé à la topologie de Scott.
3. Cont* : la sous-catégorie pleine de Ups dont les objets sont des faisceau continus.
4. Alg* : la sous-catégorie pleine des faisceaux algébriques de Cont*

Les catégories Ups, Cont* et Alg* sont isomorphes, respectivement à DDI, DCont* et DAlg*, qui sont définies de la même façon pour l'ordre inverse.

4.3 Théorème : Foen, Ups, Cont*, DAlg* sont fermées par produits et coproduits. Elles possèdent des équaliseurs et des coéqualiseurs

Démonstration :

Même démarche que pour Fais, en notant que le passage d'un faisceau à son quotient, dans la démonstration de l'existence des coéqualiseurs, respecte chacune de ces catégories. Nous détaillons l'argumentation pour Foen.

Rappel : Si X est ordonné muni d'un faisceau $<\lambda, \beta>$, une relation d'équivalence $\sim$ sur X est régulière ssi :

$$\forall\, \sigma_1, \sigma_2 \in (X) \quad \sigma_1 \sim \sigma_2 \Rightarrow \lambda(\sigma_1) = \lambda(\sigma_2)$$

En se référant à la définition générale, le faisceau quotient sur $X/\sim$ est donné par

$$\lambda^* : [\sigma] \mapsto [\lambda(\sigma)] \quad \forall\, \sigma \in \mathfrak{F}(x)/\sim$$

$$\beta^* : [\sigma] \in \beta^*([x]) \Leftrightarrow [\sigma] = [\tau] \quad \text{où}$$

$$\tau \in \beta(u), \quad u \in [x]$$

(c'est-à-dire $\beta^*([x]) = \beta([x])/\sim$ en prenant le prolongement canonique de β et $\sim$ aux parties).

Ordre sur l'espace quotient $X/\sim$:

Ici $\lambda = \sqcup$; on peut induire cet ordre de $\lambda^* = \sqcup^*$ donné ci-dessus.

Soit $[x]$ défini par $y \in [x] \Leftrightarrow x \sim y$

posons $\quad [u] \leq [v] \Leftrightarrow \sqcup(\{[u], [u]\}) = [v]$

$$\Leftrightarrow \quad [v \sqcup u] = [v] \Leftrightarrow (v \sqcup u \sim v)$$

La relation $\leq$ est un ordre partiel sur $X/\sim$ car :

- réflexive : $[u] \leq [u] \Leftrightarrow u \sqcup u = u \sim u$
- transitive : $[u] \leq [v]$ et $[v] \leq [w]$

impliquent : $u \sqcup v \sim v$

$\qquad\qquad v \sqcup w \sim w$

La régularité de $\sim$ permet d'écrire successivement :

$$u \sqcup v \sqcup w \sim v \sqcup w \sim w$$

$$u \sqcup (v \sqcup w) \sim u \sqcup w$$

d'où, par associativité de $\sqcup$ et transitivité de $\sim$, on tire $u \sqcup w \sim w$ i.e. $[u] \leq [w]$

- antisymétrique :

$$[u] \leq [v] \text{ et } [v] \leq [u] \Leftrightarrow$$

$$v \sqcup u = v \text{ et } v \sqcup u = u \Leftrightarrow$$

$$[u] = [v]$$

On sait qu'un fom est défini par

$$X \underset{s}{\overset{\sqcup}{\leftrightarrows}} P(X)$$

Le <u>fom quotient</u> est donné par la fonction spectre :

$$s^* : [x] \to \bigcup s\ ([x])/\sim$$

Soit $a \in X$ rationnel $\Leftrightarrow$ $a \in s(a)$ $\Rightarrow$
$a \in \bigcup s([a])$ $\Rightarrow$ $a \in \bigcup s\ ([a])/\sim\ = s^*([a])$
$\Rightarrow$ $[a]$ est <u>rationnel</u> pour le fom quotient.

Donc si un fom est élémentaire (ses spectres ne contiennent que des rationnels) alors son quotient est élémentaire.

<u>fin du rappel sur les faisceaux quotients</u>.

Il faut maintenant revenir aux coéqualiseurs dans <u>Foen</u> ; l'argument est le même que pour <u>Fais</u> :

$$Z \underset{g}{\overset{f}{\rightrightarrows}} X \to Y = X/\sim$$

où $\sim$ est défini par $\forall a \in Z\ \ f(a) \sim g(a)$

On a vu que $\sim$ est régulière. Soit le <u>préordre</u> sur X
$u \underset{\sim}{\sqsubseteq} v \Leftrightarrow u \sqcup v \sim v$ (alors l'ordre $\leq$ ci-dessus est induit par le préordre $\underset{\sim}{\sqsubseteq}$ en passant au quotient par la relation d'équivalence $u \approx v \Leftrightarrow u \underset{\sim}{\sqsubseteq} v$ et $v \underset{\sim}{\sqsubseteq} u$).

Soit $S \subseteq X$ tel que $\exists\, m \in X\ \forall a \in S\ \ a \sqcup m \sim m$
(i.e. $\forall a \in S\ \ a \underset{\sim}{\sqsubseteq} m$ ou $[a] \leq [m]$)

Comme par définition de $\sim$:

$$\begin{tikzcd}[column sep=small] f(u) = m \sqcup a & \sim & m = g(u) \\ & u \arrow[ul, "f"] \arrow[ur, "g"'] & \end{tikzcd}$$

$\exists\ u \in Z\ \ \forall a \in S\ \ f(u) = a \sqcup m$
$\Rightarrow$ $a \sqsubseteq f(u)$ $\forall a \in S$ $\Rightarrow$ comme X est complet sans condition, S a une borne supérieure pour $\sqsubseteq$ dans X et cette borne passe par $\underset{\sim}{\sqsubseteq}$, car $s \sqsubseteq t \Rightarrow s \underset{\sim}{\sqsubseteq} t$ donc $[s] \leq [t]$.

Donc $Y = X/\sim$ est complet sous condition. D'où on déduit que <u>Foen</u> a des coéqualisseurs. □

<u>Exponentiation de faisceaux et foncteurs.</u>

Appelons provisoirement <u>DCont</u> le sous-catégorie pleine de <u>Fais</u> dont les objets sont les faisceaux continus au sens de Scott pour l'ordre inverse, interpolables, contenant T. Et soit <u>Interp</u> une catégorie qui a pour objets tous les faisceaux ordonnés globalement interpolables $\in |\underline{\text{Fais}}|$.

La proposition 3.24 nous donne une fonction

$$
\begin{array}{rl}
f : & |\underline{\text{Interp}}| \times |\underline{\text{DCont}}| \to |\underline{\text{DCont}}| \\
 & (X, \qquad D) \to [X \longrightarrow D]
\end{array}
$$

dès que le faisceau continu de $[X \to D]$ est interpolable. L'interpolabilité de $[X \to D]$ est réalisée par exemple quand les spectres de D sont des semilattices (cf. Prop. 3.16).

Donc pour que ça marche, il suffit de completer la définition de <u>DCont</u> en imposant aux spectres des objets d'être des semilattices.

D'où alors une vraie fonction

$$G : |\underline{\text{Interp}}| \times |\underline{\text{DCont}}| \to |\underline{\text{DCont}}|$$

On veut prolonger ceci en un foncteur, analogue à celui qu'on a pour la catégorie <u>UPS</u> des trellis complets munis des fonctions Scott-continues :$[. \to .]$: $\underline{\text{UPS}} \times \underline{\text{UPS}} \to \underline{\text{UPS}}$

$$
\begin{array}{l}
\text{avec } [\,. \to .\,]\ (X,Y) = [X \to Y] \\
\text{et } \ f : S_1 \to S_2,\ g : T_1 \to T_2 \\
[\,f \to g\,] : [S_2 \to T_1] \to [S_1 \to T_2] \\
\qquad\qquad h \to ghf
\end{array}
$$

$$
\begin{array}{lll}
\text{Soit } & f : S_1 \to S_2 & \text{dans } \underline{\text{Interp}} \\
 & g : T_1 \to T_2 & \text{dans } \underline{\text{DCont}}
\end{array}
$$

$$
\begin{array}{ccc}
S_2 & \xrightarrow{h} & T_1 \\
{\scriptstyle f}\uparrow & & \downarrow{\scriptstyle g} \\
S_1 & & T_2 \\
\underline{\text{Interp}} & & \underline{\text{Dcont}}
\end{array}
\qquad
\begin{array}{l}
G(f,g) : G(S_2,T_1) \to G(S_1,T_2) \\
\qquad\qquad h \mapsto ghf
\end{array}
$$

1. Il faut vérifier que ghf est dans $G(S_1, T_2)$. Il suffit pour cela de se placer dans les conditions de Prop 3.9 .

2. Il faut vérifier que $G(f,g)$ est un <u>DCont</u>-morphisme.

Si $h \in G(S_2,T_1)$, H = spectre de $h (h = \downarrow\!\sqcap H)$

$G(f,g)\ (\downarrow\!\sqcap H)(u) = g(\downarrow\!\sqcap H)\ f\ (u) =$

$\sqcap\ \{g(\downarrow\!\sqcap H)(v) : v \in s(f(u))\} =$

$\sqcap\{g(\downarrow\!\sqcap\{k(f(v)) : k \in H\}) : v \in s(f(u))\} =$

(tout <u>DCont</u>-morphisme est continu pour la topologie de Scott inverse)

$= \sqcap\{\downarrow\!\sqcap\{g\ k\ f\ (v) : k \in H\} : v \in s(f(u))\} =$

(les infs sont calculés point par point dans $G(S_1,T_2)$)

$= \sqcap\{\downarrow\!\sqcap\ \{g\ k\ f : k \in H\})(v) : v \in s(f(u))\}$

$= \sqcap\ \{\downarrow\!\sqcap G(f,g)(H)(v) : v \in s(f(u))\}$

$= (\downarrow\!\sqcap\ G(f,g)(H))\ (u)$

Donc $G(f,g)$ est bien un DCont-morphisme (la décomposition le long des noyaux ne fait pas problème).

d'où on tire:

4.4. <u>Proposition</u> : Soit <u>Foen</u> la catégorie définie comme plus haut, et <u>DCont</u> la sous-catégorie pleine de <u>Fais</u> dont les objets sont les faisceaux ordonnés X tels que

(i) X est muni d'un T

(ii) X est continu au sens de Scott pour l'ordre inverse

(iii) Les spectres de X sont des semilattices

Alors

$$G = \underline{\text{Foen}} \times \underline{\text{DCont}} \to \underline{\text{DCont}}$$

est un foncteur.

□

Si $A \in |\underline{\text{Interp}}|$, $B \in |\underline{\text{DDI}}|$, l'espace $[A \to B]$ ordonné extensionnellement peut être muni de la topologie de Scott pour l'ordre inverse. Le faisceau associé à cette topologie en fait un objet de <u>DDI</u>. Ce faisceau est fidèle. D'où

$$\begin{array}{rccc} h : & |\underline{\text{Interp}}| \times |\underline{\text{DDI}}| & \to & |\underline{\text{DDI}}| \\ & A,\quad B & \to & [A \to B] \end{array}$$

(en fait ceci marche si $A \in |\underline{\text{Fais}}|$).

On veut prolonger h en un foncteur. On opère de la même manière.

$f : S_1 \to S_2$ dans <u>Interp</u>

$g : T_1 \to T_2$ dans <u>DDI</u>

$H(f,g) : H(S_2,T_1) \to H(S_1,T_2)$

$$h \mapsto g\,h\,f$$

1) H est bien défini dès qu'on se restreint dans <u>Interp</u> aux faisceaux ordonnées qui sont dans <u>Foen</u> (on prend alors la composition régulière des fonctions simples).

2) Il faut vérifier que $H(f,g)$ est un <u>DDI</u>-morphisme. Soit S une chaine descendante dans $H(S_2,T_1)$ de limite h

$$H(f,g)(h)(u) = H(f,g)(\sqcap S)(u) =$$

(modulo la composition régulière qui ne pose pas de problème) =

$g(\sqcap S)(f)(u) = g(\sqcap\{s(f(u)) : s \in S\}) =$ (g est un <u>DDI</u>-morphisme)

$\sqcap\{g(s(f(u))) : s \in S\} =$ (les infs sont calculés point par point dans $[S_1 \to T_2]$ et les $g\,s\,f$ sont dans $[S_1 \to T_2] \in$ <u>DDI</u>)

$= (\sqcap\{g\,s\,f : s \in S\})\,(u) = (\sqcap H(f,g)(S))(u)$; donc $H(g,f)$ est bien un <u>DDI</u>-morphisme.

D'où la Proposition

4.5. <u>Proposition</u> :

Soit <u>Foen</u> la catégorie définie comme en 4.4, et <u>DDI</u> la sous-catégorie pleine de <u>Fais</u> dont les objets sont des posets munis du faisceau S pour l'ordre inverse. Alors

$$H : \underline{\text{Foen}} \times \underline{\text{DDI}} \to \underline{\text{DDI}}$$

est un foncteur. □

Si $X \in |\underline{\text{DCont}}|$ $s,y \in X$ notons $x \succeq y$ le fait que x approxime y (i.e. $x \in s(y)$).

4.6 <u>Proposition</u> : Soit $X \in$ <u>Fais</u>, $Y \in$ <u>DDI</u> avec $T \in Y$, soit

$[\![X \to Y]\!] = \{f \in [\![X \to Y]\!] : f$ monotone croissante$\}$

$N = \{\langle e,e'\rangle_T : e \in X,\ e' \in Y\}$.

Alors le faisceau élémentaire sur $[\![X \to Y]\!]$ défini par $s : f \to s(f) = (\uparrow f) \cap N$ est plus fin que le faisceau de $[\![X \to Y]\!]$ considéré comme un objet de <u>DDI</u> (noté $[\![X \to Y]\!]_{DDI}$).

De plus si $X \in$ Interp, $[X \to Y]_{DDI} = [\![X \to Y]\!]_{DDI} = H(X,Y)$

□

Démonstration : immédiat par 4.5 et 3.14

4.7 Proposition. Soit $X \in$ Foen, $Y \in$ DCont et $\forall e \in X$, $e' \in N(Y)$ soit $\langle e, e'\rangle^*_T : x \in X \to \begin{cases} e' & \text{si } e \in s(x) \\ T & \text{sinon} \end{cases}$

$\mathcal{N}^* = \{\langle e,e'\rangle^*_T : e \in X,\ e' \in Y\}$

Alors le faisceau sur $[X \to Y]$ défini par

$$s^* : f \to s^*(f) = \{\langle e,e'\rangle^*_T : e' \geq f(e)\}$$

est plus fin que le faisceau de $[X \to Y]$ considéré comme un objet de DCont. De plus

□

$$[X \to Y]_{\underline{DCont}} = G(X,Y)$$

Démonstration : le fait que s^* est un faisceau découle de 3.15 (i). Si $[X \to Y]$ est muni de la topologie de Scott pour $\supseteq$ et $\mathcal{P}([X \to Y])$ de la topologie de Scott, alors s^* est topologique. Le reste découle de 4.5 et 2.18 (ii). □

Ces deux faits assez faciles à établir éclaireront l'utilisation que nous ferons plus loin des échelons de Nolin en liaison avec la signification intuitive des algorithmes F X Y de cet auteur.

II. Limites et colimites : la notion d'approximation dans les catégories de faisceaux

Notre présentation utilise des outils empruntés à Manin 1977, pp 96 sqq., [SCS] pp 206 sqq, [MCL], et Schubert 1972.

Soit $\mathcal{U}$ un univers (de von Neumann, cf. Manin). On supposera $\forall x \in$ Fais $|x| \in \mathcal{U}$.

Une petite catégorie est une catégorie dont la classe des objets et la classe des morphismes sont des éléments de $\mathcal{U}$.

Soit C une catégorie arbitraire. Un diagramme dans C est simplement un foncteur $\sigma : I \to C$ où I est une petite catégorie.

Soit X une petite catégorie. On définit alors l'ensemble (ou la catégorie discrète) des diagrammes de X F(X) comme étant

$F(X) = \{\sigma : I \to X \mid I$ est une petite catégorie et σ un foncteur de I dans X$\}$

<u>Exemples de diagrammes</u> :

(i) Soit ω la petite catégorie

$0 \to 1 \to 2 \to 3 \to \dots$

$D \in$ <u>Fais</u> un faisceau et soit σ le diagramme

$$\forall i \qquad \sigma(0) = D, \quad \sigma(i) = [\sigma(i-1) \to \sigma(i-1)]$$

On a alors le diagramme

$$D = D_0 \xrightarrow{i_0} D_1 \xrightarrow{i_1} D_2 \xrightarrow{i_2} D_3 \xrightarrow{i_3} \dots$$

où $D_i = \sigma(i)$ $\forall i \in \omega$.

(ii) Soit ω^{op} la catégorie duale de ω :

$0 \leftarrow 1 \leftarrow 2 \leftarrow 3 \leftarrow \dots$

On définit de même un diagramme

$$D = D_o \xleftarrow{j_o} D_1 \xleftarrow{j_1} D_2 \leftarrow \dots$$

Un <u>diagramme projectif</u> (1) dans une catégorie X est un foncteur $\sigma : I^{op} \to X$ dont le domaine est un poset dirigé I tel que $\forall i \in I$ $\sigma(1_i) = 1_{\sigma(i)}$ et $i \le j \le k \Rightarrow \sigma(i \leftarrow k) = \sigma(i \leftarrow j)\ \sigma(j \leftarrow k)$.

Dualement un <u>diagramme inductif</u> (2) dans une catégorie X est un foncteur $\sigma : I \to X$ tel que I est dirigé et $\forall i \in I$ $\sigma(1_i) = 1_{\sigma(i)}$,

$i \le j \le k \Rightarrow \sigma(i \to k) = \sigma(i \to j)\ \sigma(j \to k)$

(On rappelle que dans un poset $a \to b$ dénote $a \sqsubseteq b$).

<u>Exemple d'application</u> : le "powerdomain"

Reprenons les notations du chapitre III § III-4. Si N est un ensemble (fini), alors $(N^{\omega})^{op}$ est dirigé et toute fonction régulière $f : N^{\omega} \to D$ où D est un cpo définit un foncteur sur les catégories associées $\sigma_f : N^{\omega} \to D$ tq

$$\forall i \in N^{\omega} \quad \sigma_f(1_i) = 1_{\sigma_f(i)} = 1_{f(i)}$$

$$i \le j \le k \Rightarrow \sigma_f(i \to k) = \sigma_f(i \to j)\ \sigma_f(j \to k)$$

puisque f est monotone.

---

(1) ou <u>inverse</u>

(2) ou <u>direct</u>

Donc toute fonction régulière $f \in [N^{\omega} \to D]$ définit un diagramme projectif dans la catégorie associées à D. Réciproquement si $\sigma : N^{\omega} \to D$ est un diagramme projectif dans D, alors il définit de manière unique une fonction monotone $g \; : Rat(N^{\omega}) \to D$ donc une fonction régulière $\bar{g} \; : N^{\omega} \to D$. (Noter que cette bijection n'utilise que partiellement le fait que D est un cpo). $\square$

Si L est un objet d'une petite catégorie X, le foncteur $|L| : I \to X$ est par définition le foncteur constant de valeur L sur les objets, et 1_L sur les flèches.

Soit $\sigma \in F(X)$ un diagramme sur X. Un <u>cône de sommet</u> L sur σ est une transformation naturelle $g : |L| \to \sigma$:

$$\begin{tikzcd} & I \arrow[dl, "|L|"'] \arrow[dr, "\sigma"] & \\ X \arrow[rr, bend right, "g"'] & & X \end{tikzcd} \qquad \text{i.e.}$$

$$\begin{tikzcd} i \arrow[d, "a"'] & |L|\,(i) = L \arrow[d, "id_L"'] \arrow[rr, "g_i"] & & \sigma\,(i) \arrow[d, "\sigma\,(a)"] \\ j & |L|\,(j) = L \arrow[rr, "g_j"'] & & \sigma\,(j) \end{tikzcd} \quad \text{tq}$$

le diagramme commute, i.e.

$$\begin{tikzcd} & L \arrow[dl, "g_i"'] \arrow[dr, "g_j"] & \\ \sigma\,(i) \arrow[rr, "\sigma\,(a)"'] & & \sigma\,(j) \end{tikzcd}$$

D'une façon plus explicite ceci signifie que pour chaque objet i de I on a un X-morphisme $g_i : L \to \sigma(i)$ tel que pour toute flèche $a : i \to j$ le diagramme ci-dessus commute.

Un cone $g : |L| \to \sigma$ sur un diagramme σ est appelé un <u>cône limite</u> ssi pour tout cône $h : |H| \to \sigma$ sur le même diagramme, il existe un unique X-morphisme $h^* : H \to L$ tel que $g|h^*| = h$ où $|h^*| : |H| \to |L|$ est la transformation naturelle constante avec $|h^*|(i) = h^* : H \to L$ pour tous objets i de I. C'est-à-dire que pour tout $i \in I$ le diagramme

$$\begin{tikzcd} & H \arrow[d, "h^*"] \arrow[dl, bend right, "h_i"'] \\ \sigma(i) & L \arrow[l, "g_i"] \end{tikzcd} \qquad \text{commute}$$

Le sommet L d'un cône limite d'un diagramme σ est appelé <u>la limite de</u> σ, et noté <u>$\lim \sigma$</u>. Les flèches $g_i : \lim \sigma \to \sigma(i)$ sont appelées <u>flèches limites</u>.

De la même manière on peut définir : un <u>co-cône de sommet</u> L <u>sous un diagramme</u> σ est une transformation naturelle $g : \sigma \to |L|$

$$\begin{tikzcd} i \arrow[d, "a"'] & \sigma(i) \arrow[r, "g_i"] \arrow[d, "\sigma(a)"'] & L & (i) = L \arrow[d, "id"] \\ j & \sigma(j) \arrow[r, "g_i"'] & L & (j) = L \end{tikzcd}$$

tq le diagramme commute

$$\begin{tikzcd} \sigma(i) \arrow[rr, "\sigma(a)"] \arrow[dr, "g_i"'] & & \sigma(j) \arrow[dl, "g_j"] \\ & L & \end{tikzcd}$$

On obtient de la même manière des colimites en renversant les flèches.

On a donc ici une notion de reconstruction des objets, de même qu'on avait une notion de reconstruction des points dans les faisceaux du chapitre I. Cette notion est la fonction (partielle) qui associe à tout diagramme σ de X sa limite (ou sa colimite) dans X lorsqu'elle existe. On a donc deux fonctions partielles

$$\lim, \operatorname{colim} : F(X) \to X$$

qui sont en fait des <u>foncteurs</u> si X est complète (cf [MCL], pp 110, exercice 3).

Le "foncteur" lim : $F(X) \to X$ qui envoie les diagrammes sur leurs limites possède la <u>propriété de Fubini forte</u> (i.e. $\lim\limits_{i,j} w_{ij}$ existe si et seulement si $\lim\limits_{i}(\lim\limits_{j} w_{ij})$ existe et dans le cas où l'un existe on a l'égalité) (cf. Schubert 1972, Proposition 7.6.2 pp 55).

Si on dualise ceci i.e. si on remplace les cônes par les co-cônes, limites par colimites, on a encore la même chose : la <u>propriété de Fubini forte</u> vient de Schubert 1972, Proposition 8.6.2 pp 67, par exemple, (i.e. les colimites commutent avec les colimites).

Notons qu'en général les limites ne commutent pas avec les colimites : ainsi dans les catégories qui sont des posets (avec $a \subseteq b \Leftrightarrow a \to b$ est une flèche de la catégorie), les limites sont des bornes inférieures et les colimites des bornes supérieures et leur commutativité signifie une complète distributivité très forte du poset (cf Birkhoff 1961, pp 146 sqq, [SCS] pp 59, et infra)

On voit donc que l'analogie avec des notions présentées au chapitre I est très forte.

<u>Notation</u> : Si x et y sont des objets de quelque catégorie on notera $x \cong y$ le fait que ces objets sont <u>isomorphes</u> (on ne peut pas les distinguer à l'intérieur de la catégorie)

D'où la définition :

<u>Définition</u> : <u>Faisceau sagittal</u> (ou <u>large</u>)

Soit X une petite catégorie. Un <u>faisceau sagittal</u> (ou <u>large)</u> sur X est un couple $\langle \lambda, \beta \rangle$ tel que, si $Ob(X)$ dénote les objets de X,

(i) $\lambda : F(X) \to Ob(X)$ est une fonction partielle, $\lambda \in \{ \lim, \mathrm{colim} \}$

(ii) $\beta : Ob(X) \to \mathcal{P}(F(X))$

(a) $\forall x \in Ob(X) \quad \forall \sigma \in \beta(x) \quad x = \lambda(\sigma)$

(b) propriété de Fubini : pour tout diagramme $\sigma : I \times J \to X$ sur X, si $\forall j \quad \sigma_j : I \times \{j\} \to X$, $x_j = \lambda(\sigma_j)$ et si le diagramme $\tau : j \to x_j$ a pour limite y, alors le diagramme σ a pour limite y.

Un faisceau sagittal sur une catégorie X étant donné, on a de la même façon une notion de <u>fibre</u> $\beta(x)$ d'un objet $x \in X$, <u>noyau du faisceau</u>, et de <u>système approximant d'un objet</u> de la catégorie.

<u>Remarque 1</u> : Si X est un ensemble, J un ensemble d'indices, si on considère chaque ensemble comme une catégorie discrète, alors toute famille $\sigma : J \to X$, $j \to \sigma(j) = x_j$ est un diagramme dans la catégorie X. Comme la notion d'ensemble peut être vue comme la notion de catégorie la plus pauvre, les familles sont donc des diagrammes extrêmement simples. On voit donc que les systèmes approximants manipulés ici (les diagrammes), sont des généralisations très simples de ceux que nous avions aux chapitres précédents (les familles). L'intérêt de cette généralisation pour nous est qu'elle montre (comme on va le voir) une ligne de force des modèles de sémantique des langages de programmation et du λ-calcul : la notion de convergence dans ces modèles et les formes variées sous lesquelles elle apparaît.

<u>Remarque 2</u> : <u>La catégorie des faisceaux Fais possède une structure de faisceau sagittal</u> (trivialement) soit pour $\lambda = \lim$, soit pour $\lambda = \mathrm{colim}$. (On note la parenté avec les faisceaux ordonnés où on avait $\lambda = \bigsqcup$ ou $\lambda = \bigsqcap$). La catégorie <u>Fais</u> constitue en quelque sorte un faisceau sagittal universel, i.e. dans lequel on peut "injecter" par un foncteur fidèle tous les faisceaux sagittaux dont les objets sont des faisceaux.

<u>Foncteurs réguliers</u>. <u>Définition</u> :

Soit X (resp. Y) une petite catégorie munie d'un faisceau sagittal $<\lambda, \beta>$ (resp. $<\lambda', \beta'>$). Un foncteur $F : X \to Y$ est <u>régulier en $x \in |X|$</u> ssi

$\forall \sigma \in \beta(x) \quad F(x) = \lambda'(F(\sigma))$

(où le signe $=$ dénote un isomorphisme)

Un foncteur $F : X \to Y$ est <u>régulier</u> ssi il est régulier pour tout objet $x \in X$. □

Un foncteur $F : X \to Y$ régulier est donc un foncteur qui préserve toutes les limites au sens du faisceau sagittal de X.

<u>Exemples</u> :

(i) (cf. [MCL], pp 112, théorème 1). Si <u>Set</u> est la catégorie dont les objets sont les petits ensembles, et les flèches toutes les fonctions entre eux, C une petite catégorie, tout hom-foncteur $C(a,.) : C \to \underline{Set}$ preserve toutes les limites

(ii) ([SCS] pp 219, théorème 3.19) Soit $\underline{INF}\uparrow$ la catégorie dont les objets sont les treillis complets et les flèches toutes les fonctions préservant les bornes sup. des parties dirigées et toutes les bornes inférieures, munie du faisceau sagittal $\langle \lim, \beta \rangle$ dont les systèmes approximants sont les diagrammes projectifs. Alors le foncteur "espace des fonctions"

$$\text{Funct} : \underline{INF}\uparrow \to \underline{INF}\uparrow$$

qui envoie chaque objet D sur $[D \to D]$ est régulier.

(iii) Les foncteurs <u>pro-continus</u> de [SCS] pp 224, ne sont rien d'autre que des foncteurs régulier entre deux catégories pro-complètes (= où tout diagramme projectif a une limite). □

Les faisceaux sagittaux et les foncteurs réguliers ont été utilisés par Hofmann [HOF] et Smyth et Plotkin [PLS] pour calculer les points fixes de certaines classes de foncteurs et leurs approximations comme limites ou colimites.

En ce qui concerne leur articulation avec les autres faisceaux, on peut reprendre ici la même démarche qu'au Chapitre I pour les fonctions régulières et s'apercevoir qu'elle s'applique sans grand changement. (Ce fait a été noté par Smyth et Plotkin pour les colimites dans les ω-catégories).

Ainsi :

4.8 <u>Lemme</u> : Soient X, Y deux faisceaux sagittaux. Alors

(i) le foncteur identité est régulier

(ii) Tout foncteur constant est régulier

(iii) Si $\{F_i : i \in I\}$ est un diagramme de foncteurs réguliers en $x \in X$ et tels que les limites $\lim\limits_i F_i(x)$ et $\forall\sigma \in \beta(x)$ $\lim\limits_\sigma(\lim\limits_i F_i(\sigma))$ existent dans Y, alors le "foncteur partiel"

$$G : y \mapsto \lim_i F_i(y)$$

est régulier en x. □

<u>Démonstration</u> : analogue à 1.2 mais avec des diagrammes plus compliqués.

4.9. <u>Lemme</u> : régularité des foncteurs projections. Soient X,Y deux faisceaux sagittaux et $X \times Y$ leur produit. Alors les deux foncteurs projection $P_1 : X \times Y \to X$ et $P_2 : X \times Y \to Y$ sont réguliers.

4.10. <u>Lemme</u> : Soient X,Y,Z des faisceaux sagittaux. Alors un foncteur $F : X \times Y \quad Z$ est régulier sur $X \times Y$ ssi il est régulier pour chacune de ses "variables". □

etc...

On peut relever que en considérant les ensembles partiellement ordonnés X comme des catégories C par $Ob(C) = X$, $hom(a,b) = \{a \to b\}$ si $a \sqsubseteq b$, tout ce qui a été fait aux chapitres précedents pour les faisceaux ordonnés peut être repris dans le cadre des faisceaux sagittaux. On peut en particulier pour décrire la déclaration de type dans un en-tête de procédure (cf. chap III § IV)

<u>type b</u> <u>procédure</u> g(x) <u>type a</u> x
<u>début</u>
etc
<u>fin</u>

définir à la place de l'échelon de Nolin <a,b> un <u>foncteur de Nolin</u> $F_{a,b}$

$$F_{a,b} : X \to X \qquad\qquad F : \begin{array}{l} X \times X \to X \\ (a,b) \to F_{a,b} \end{array}$$

contravariant sur a et covariant sur b. Le cumul des types (i.e. leur intersection au sens de Nolin) étant obtenu en prenant des limites (au sens des catégories) d'objets $F_{a,b}$.

Nota : Dans la suite nous ne considérerons que des petites catégories.

III. La méthode de Scott de calcul des limites

4.11 Proposition [SCS] : Soient X,Y des posets dont toute partie majorée possède une borne supérieure, et possédant $\perp$. Alors $\forall f : X \to Y$

(i) f est continue pour la topologie inférieure et est un morphisme de semilattice $\Leftrightarrow$ f préserve les bornes inférieures.

(ii) f est continue au sens de Lawson et est un morphisme de semi-treillis $\Leftrightarrow$ f preserve les bornes inférieures et est continue au sens de Scott. □

Noter que la Lawson-continuité ne suffit pas en général à impliquer le membre droit de (ii).

Nous allons d'abord traduire pour les cpo's des propriétés établies dans [SCS] pour les treillis complets.

Nous définissons trois catégories dont les objets sont les cpo's complets sous conditions (= toute partie majorée a une borne sup), et les flèches :

pour LWS = les morphismes de semilattice continus au sens de Lawson.
pour CPO = les fonctions continues au sens de Scott.
pour $SUPP^o$ = les fonctions f préservent des sup. arbitraires (donc Scott-continues) et tq $\forall\ \mathcal{U}$ ouvert dans Dom(f) $\uparrow f(\mathcal{U})$ est ouvert dans Im(f).

On rappelle que dans une connection de Galois $S \underset{d}{\overset{g}{\rightleftarrows}} T$ entre posets, g est l'adjointe supérieure et d l'adjointe inférieure ; g préserve tous les infs et d préserve tous les sups. L'une quelconque de ces deux fonctions détermine l'autre de façon unique, d'où les deux foncteurs adjoints :

$$\hat{}: g \to \hat{g} = d$$
$$\vee: d \to \check{d} = g$$

4.12 Proposition : Les catégories LWS et $SUPP^o$ sont duales sous les foncteurs adjoints $\vee$ et $\hat{}$.

dém : [SCS] pp 180 et théorème 1.10 pp 182.

4.13 <u>Proposition</u> : Soit $D : J^{op} \to LWS$ un système projectif dans LWS, $g_j : \lim D \to D(j)$ son cône limite. Alors le cône colimite $\hat{g}_j : D(j) \to \lim D$ est déterminé pour tous $i,j \in J$ par les formules :

(i) $g_j \hat{g}_i = \sup\{ g_{jk} \hat{g}_{ik} : i,j \leq k \text{ dans } J\}$

(ii) $\sup \hat{g}_j g_j = 1_{\lim D}$

(iii) Si de plus...
$$\begin{tikzcd} D(i) \arrow[rr, "\hat{g}_{ij}"] \arrow[dr, "d_i"'] & & D(j) \arrow[dl, "d_j"] \\ & S & \end{tikzcd}$$

est un cône dans <u>CPO</u> sous le système direct $\hat{D}$, alors il existe une unique fonction $d : \lim D \to S$ dans CPO telle que $d_i = d\hat{g}_i$ pour tout $i \in J$, qui est donnée par la formule

$$d : x \to \sup\{d_j g_j(x) : j \in J\}$$ □

<u>démonstration</u> : retranscription de celle de [SCS] pp 209-212 pour les treillis complets, en considérent que toutes les familles considérées sont monotones donc convergent dans des cpo's.

4.14 <u>Proposition</u> : Soit $D : J^{op} \to LWS$ un système inverse dans LWS. Soit $g_j : L \to D(j)$ un cône sur D. Alors les assertions suivantes sont équivalentes :

(i) $g_j : L \to D(j)$ est un cône limite dans LWS

(ii) $\hat{g}_j : \hat{D}(j) \to L$ est un cône colimite dans <u>CPO</u>

<u>démonstration</u> : découle de 4.12 et 4.13. cf. aussi [SCS] pp 213. □

La méthode de Scott pour construire des modèles du λ-calcul peut alors être énoncé de la façon suivante

4.15 <u>Proposition</u> : Soit X un cpo muni du faisceau associé à la topologie de Scott, et le diagramme

$$X_o \underset{j_o}{\overset{i_o}{\rightleftarrows}} X_1 \underset{j_1}{\overset{i_1}{\rightleftarrows}} X_2 \underset{j_2}{\overset{i_2}{\rightleftarrows}} \cdots X_n \underset{j_n}{\overset{i_n}{\rightleftarrows}} \cdots$$

où $X_o = X$, $X_{n+1} = [X_n \to X_n]$ au sens des cpos

$$i_o(x) = \lambda t.x$$
$$j_o(y) = y(\bot)$$
$$j_{n+1}(y) = j_n \circ y \circ i_n$$
$$i_{n+1}(x) = i_n \circ x \circ j_n$$

Alors la limite projective X_∞

$$X_\infty = \varprojlim X_n = \{ (x_n)_{n \in N} : x_n = j_{n+1}(x_{n+1}) \}$$

est telle que $X_\infty = [X_\infty \to X_\infty]$ dans la catégorie des cpos, avec pour flèches les fonctions continues au sens de Scott. □

Démonstration :

(i) Si $x \in X_\infty$ définir $\forall y \in X_\infty \quad x(y) = \bigsqcup_n x_{n+1}(y_n)$.

Alors $y \to x(y)$ est clairement règulière à cause de la régularité des x_{n+1}, de $y \to y_n$ et du lemme 1.2. Donc on peut injecter X_∞ dans $[X_\infty \to X_\infty]$.

(ii) Réciproquement si $f \in [X_\infty \to X_\infty]$ définir $[f] \in X_\infty$ par $[f] = \bigsqcup_n (\lambda y \in X_n . (f(y))_n)$, et $[f](x) = \bigsqcup_n f_{n+1}(x_n) =$

$$\bigsqcup_n ([f].x_n)_n = \bigsqcup_n ((\bigsqcup_m (\lambda y \in X_m . (f(y))_m).x_n)_n = \text{(Fubini)}$$

$$\bigsqcup_{n,m} (\lambda y \in X_n . (f(y))_m . x_n)_n = \bigsqcup_n (\lambda y \in X_n . (f(y))_n . x_n) = \bigsqcup_n (f(x_n))_n =$$

$$\text{(Fubini)} \bigsqcup_{p,q} (f(y_p))_q = \bigsqcup_p f(y_p) = f(x).$$

d'où la proposition. (la bijection est régulière). □

On peut voir dans l'énoncé précédent que $X_n \underset{j_n}{\overset{i_n}{\rightleftarrows}} X_{n+1}$ est une connection de Galois avec $i_n = \hat{j}_n$. On peut donc appliquer la proposition 4.14.

L'énoncé précédent peut aussi s'exprimer dans le cadre des catégories de la manière suivante : soit CPO la sous-catégorie pleine de Ups dont les objets sont des cpo. Alors CPO possède des produits et des équaliseurs, donc est complète. De plus le foncteur d'oubli

$$\text{CPO} \to \text{Set}$$

qui oublie la structure algébrique des objets préserve et crée des produits et équaliseurs, donc les limites (cf [MCL]).Donc X_∞ est (une) limite projective dans CPO.

Dans la suite on appellera <u>diagramme de Scott</u> le diagramme défini en 4.14. X_o sera la <u>racine</u> de ce diagramme.

Avant d'appliquer ceci à nos faisceaux, nous voulons enregistrer un fait important qui nous indique dans certains cas, que l'information interne des faisceaux (ou si l'on préfère leur processus d'approximation interne) circule bien au niveau du faisceau sagittal. C'est une propriété du diagramme de Scott.

<u>Lemme de circulation</u> :

4.16 <u>Lemme</u> : Soit X un fom où $\perp \in X$ et supposons que $\forall n\ X_{n+1} = [X_n \to X_n]$ est muni du faisceau de la convergence simple. Considérons le diagramme de Scott défini au moyen de la suite X_n. (On supposera que $fog = o(g,f)$ pour les fonctions). Alors pour tout $(x_n)_{n \in \mathbb{N}} \in \varprojlim X_n$ $\forall n \in \mathbb{N}$, on a

(i) $a < x_n \Rightarrow i_n(a) < i_n(x_n)$

i.e. $i_n(s(x_n)) \subseteq s(i_n(x_n)) \subseteq s(x_{n+1})$

(ii) $b < x_{n+1} \Rightarrow j_n(b) < j_n(x_{n+1})$

i.e. $j_n(s(x_{n+1})) \subseteq s(j_n(x_{n+1}))$

(iii) $s(x_n) = j_n(s(x_{n+1}))$ et $i_n(s(x_n)) = s(i_n(x_n))$

<u>Démonstration</u> : Ceci peut se montrer par induction

(i) $a < x_n \Rightarrow i_n(x) < i_n(x_n)$

$n = 0$ $a < x_o \Rightarrow i_o(x) = \lambda t.a < \lambda t.x_o = i_o(x_o)$

(faisceau de la convergence simple)

Soit vrai pour n ; montrons le pour $n+1$:

$a < x_{n+1}$ $i_{n+1}(a) = i_n o a\, o j_n$.

$i_{n+1}(x_{n+1}) = i_n\, o x_{n+1} o j_n$

$a\, o j_n < x_{n+1}\, o j_n$ (faisceau fidèle) d'où

$i_n o\, a\, o j_n < i_n\, o\, x_{n+1}\, o j_n$ en utilisant l'hypothèse d'induction.

(ii) $b < x_{n+1} \Rightarrow j_n(b) < j_n(x_{n+1})$

$n = 0$: $b < x_1 \Rightarrow j_o(b) = b(\perp) < j_o(x_1) = x_1(\perp)$ (fidélité) soit vrai pour n :

$b < x_{n+1} \Rightarrow j_n(b) < j_n(x_{n+1})$

? même chose pour $n + 1$:

$$b < x_{n+2} \overset{?}{\Rightarrow} j_{n+1}(b) < j_{n+1}(x_{n+2})$$

$$j_{n+1}(b) = j_n ob \circ i_n$$

$$j_{n+1}(x_{n+2}) = j_n \circ x_{n+2} \circ i_n$$

$b < x_{n+2} \Rightarrow \forall t \in X_{n+1} \quad b(t) < x_{n+2}(t)$

si $t = i_n(s)$ alors

$$\forall s \in X_n \; b(\underbrace{i_n(s)}_{\in X_{n+1}}) < x_{n+2}(\underbrace{i_n(s)}_{\in X_{n+1}})$$

on descend tout par j_n en vertu de l'hypothèse d'induction :

$\forall s \in X_n \; (j_n \text{ ob} \circ i_n)(s) < (j_n \circ x_{n+2} \circ i_n)(s)$

d'où $j_n \circ b \circ i_n < j_n \circ x_{n+2} \circ i_n$ puisque le faisceau est de la convergence simple.

i.e. $j_{n+1}(b) < j_{n+1}(x_{n+2}) = x_{n+1}$ d'où (ii)

(iii) et (iv) se déduisent de (i) et (ii) $\square$

<u>Remarque</u> : <u>Doubles connections de Galois</u>

On peut noter ici qu'un diagramme de Scott étant donné, l'application $j_o : X_1 \to X_o$ est de la forme $\delta : [S \to T] \to T$,

$\delta(f) = f(\bot)$. Comme

$\delta(\sqcup f_j) = (\sqcup f_j)(\bot) = \sqcup(f_j(\bot)) = \sqcup\delta(f_j)$

la fonction δ conserve des sup. arbitraires $\Rightarrow \delta$ est l'adjointe inférieure d'une <u>connection de Galois</u> dont l'adjointe supérieure est

$$\gamma(t) = \max \; \delta^{-1}(\downarrow t) \quad = \max\{f : f(\bot) \sqsubseteq t\}$$

c'est-à-dire

$$\gamma(t)(s) = \begin{cases} T & \text{si } s \neq \bot \\ t & \text{sinon (i.e. } s = \bot \text{ i.e. } s \sqsubseteq \bot) \end{cases}$$

i.e. $\gamma(t) = <\bot, t>_T$

L'adjointe supérieure de j_o (dans le cas d'un poset avec $\top$ et $\perp$) est donc l'application $t \to <\perp, t>_T$ qui envoie tout élément $t \in X_1$ sur l'<u>échelon de Nolin</u> $<\perp, t>_T$

Plus généralement $\forall_n$ j_{n+1} conserve des sups arbitraires et a pour adjointe supérieure, si X_n est un treillis complet

$$\gamma_{n+1}(t) = \max\{ f : j_n \circ f \circ i_n \sqsubseteq t \}.$$

Nous appellerons ceci des <u>doubles connections de Galois</u>. □

4.17. <u>Théorème</u> : Soit D un cpo tel que deux éléments quelconques majorés possèdent une borne supérieure, et tel qu'il est muni d'une structure de faisceau continu. Alors le diagramme de Scott de racine D a une limite projective D_∞ qui possède une structure de faisceau continu et de cpo complet sous condition. De plus

$$D_\infty = [D_\infty \to D_\infty]$$

<u>Démonstration</u> (en utilisant uniquement des notions de théorie des faisceaux).

Noter que dans un cpo deux éléments majorés quelconques ont une borne supérieure ssi il est complet sous condition à cause de la convergence des parties dirigées.

Nous savons que si D et D' sont des faisceaux continus interpolables, $\perp \in D'$ alors $[D \to D']$ possède une structure de faisceau continu par Proposition 3.22. Nous voulons faire marcher ceci à des niveaux de fonctionnalité supérieurs, i.e. avoir une propriété d'interpolabilité sur $[D \to D']$. Ceci peut être fait en utilisant Proposition 2.15 qui indique que tout faisceau continu sur un semi-treillis complet est interpolable. Maintenant si D, D' sont des cpo's complets sous-condition, alors $[D \to D']$ est un cpo complet sous condition. Les cpo's complets sous condition forment une sous-classe des semi-treillis complets. Donc si D,D' sont des cpo's, complets sous condition, faisceaux continus, $\perp \in D'$, alors $[D \to D']$ vérifie les mêmes hypothèses et contient $\perp$.

D_∞ est-il complet sous condition ? Oui car $\forall n \in N$ $j_n(a \sqcup b) = j_n(a) \sqcup j_n(b)$ puisque

$$j_o(a \sqcup b) = (a \sqcup b)(\perp) = a(\perp) \sqcup b(\perp) = j_o(a) \sqcup j_o(b)$$

$$j_{n+1}(a \sqcup b) = j_n \circ (a \sqcup b) \circ i_n = j_n \circ (a \circ i_n \sqcup b \circ i_n) =$$

$$j_n \circ a \circ i_n \sqcup j_n \circ b \circ i_n = j_{n+1}(a) \sqcup j_{n+1}(b)$$

(ceci peut aussi s'exprimer en disant que j_n est l'adjointe inférieure d'une connection de Galois, comme nous l'avons noté plus haut, donc prèserve les bornes supérieures).

Donc D_∞ est un cpo complet sous condition qui possède une structure de faisceau continu déduite de celle de $\prod_{i=0}^{\infty} D_i$ (lemme 4.16)

Maintenant puisque toute fonction régulière sur un faisceau continu, semi-treillis complet, est continue, on se ramène à 4.15. □

Ce résultat généralise la construction de Scott sur les treillis continus. Il est mentionné sans démonstration par H. Egli [EGL].

Il reste que le théorème précédent peut sembler peu satisfaisant du point de vue de la théorie de la programmation. La raison en est que tout élément de D_∞ est en même temps une fonction de D_∞ dans D_∞, et l'origine en est la nature purement combinatoire du λ-calcul de Church. Cette propriété n'est évidemment pas vraie dans les programmes où l'on utilise des objets plus primitifs que les autres, qui sont les données : nombres, valeurs de vérité, etc... et ces éléments n'ont pas la même "puissance fonctionnelle" que les éléments de D_∞, capables d'"ingérer" n'importe quel élément pour fournir une valeur. Un modèle contenant de tels "atomes", E_∞, a été construit par C.P. Wadsworth [WAD], mais ce modèle est trop petit lorsque le langage de programmation contient aussi des déclarations de types, comme dans l'exemple pseudo-Algol suivant dont nous avons déjà discuté dans les Prolégomènes.

```
Booléen x ;
y : = 0
si  x  alors  y : = fact(y) sinon  y: = y * 3 fsi
```

Tout programmeur écrivant un compilateur conversationnel pour un tel langage doit traiter les déclarations (comme booléen x) comme de simples affectations, donc considérer des ensembles (tels {vrai, faux}) comme des données, ou en d'autres termes considérer les types de données comme des objets (donc des objets de l'espace qui donne la signification de ses programmes et données). Nous avons indiqué dans les Prolégomènes les contributions de Nolin et Shamir et Wadge dans ce sens (cf aussi Weyhrauch [WEY] sur le système FOL, et Donahue [DON] axé sur l'implémentation d'un langage avec types: Russell).

Comme les algorithmes de Nolin et les fonctions "tight" de Shamir et Wadge sont simplement des fonctions régulières sur quelque faisceau (cf chap. I), nous voulons maintenant considérer ce problème dans le cadre général que nous avons esquissé. Notre but est de construire ce qu'on pourrait appeler un <u>modèle avec types,</u> qui généralise le <u>modèle avec atomes</u> de Wadsworth.

Avant d'entrer dans les détails techniques de notre construction, inspirée de celle de Nolin (cf. Proleg.) , nous en donnons une idée générale approximative, informatique et mathématique :

1) la base de notre construction sera la thèse 1 (cf. Proleg.27)

2) <u>la notion de bifaisceau</u> : nous voulons manipuler simultanément deux notions d'approximation
(i) d'une part celle associée à la vérification des types de bases (cf. Prolégomènes) et qui reconstruit, en partant d'objets <u>rationnels</u> (ou "<u>atomiques</u>" chez Nolin) tous les autres par borne supérieure.

(ii) d'autre part celle induite par la thèse 2 (pp 28): aux niveaux fonctionnels supérieurs, les fonctions sont approximées comme intersections de types, donc pour nous d'<u>échelons de Nolin</u>.
D'où la notion de bifaisceau.

3) <u>Le (bi-) foncteur Nolrat</u> : La thèse 2 implique d'emblée une structure de faisceau particulière sur l'espace des premières fonctions, exprimée en termes de programmation par :

"les déclarations de types dans un en-tête de procédure donnent une première approximation de la fonction calculée par cette procédure".

La même chose exprimés de façon mathématique fournit un foncteur Nolrat

Nolrat : $(X,Y) \rightarrow \mathrm{Nolrat}(X,Y) = [X \rightarrow Y]$

où X est un faisceau ordonné élémentaire normalisé dont la fonction spectre définit un fom, Y un bifaisceau possédant T ("tout" au sens de Nolin), dont le faisceau supérieur est algébrique.

4. Le schéma d'approximation de Wadsworth

Si $D = D_o$ est le bifaisceau fourni au départ, nous considétons le schéma de suite suivant (Wadsworth 1972) :

$$A_o = D$$
$$A_1 = D + [A_o \to A_o]$$
$$A_{n+2} = D + [A_{n+1} \to A_{n+1}] \quad \forall n \in N$$

Pour construire la limite de ces A_n, nous injectons chaque A_n dans A_{n+1} en envoyant la partie D de A_n dans la partie D de A_{n+1}, et la partie fonctionnelle de A_n dans la partie fonctionnelle de A_{n+1} (ce qui revient à résoudre par approximations successives l'équation de Nolin $A = A_B + A_F$).

5. Nous examinons la limite de ce schéma d'approximation dans la catégorie des ensembles (dénotée Set). Nous munissons cette limite d'une structure de "faisceau" qui correspond aux besoins de l'informatique et examinons ses propriétés. Nous obtenons une bijection

$$A_\infty \leftrightarrow D + [A_\infty \to A_\infty]$$

6. Tout ce qui précède nous amène à préciser la catégorie dans laquelle se trouvent nos objects, et à réexaminer la limite du schéma de 4. (qui devient alors un diagramme au sens où nous l'avons défini plus haut) dans cette nouvelle catégorie. □

Nous voulons maintenant nous attaquer à la construction de ce domaine avec types : A_∞ . Mais auparavant nous avons besoin de préciser et développer certains résultats ayant trait à l'exponentiation des faisceaux et à leur somme. L'aspect purement "sagittal" est insuffisant si nous voulons des espaces suffisamment riches.

4.18 Lemme (3.14 bis) : Soit D' un fom contenant $\perp$, $D = D_o + D_1$ un coproduit de deux fom interpolables D_o et D_1 tels que $\psi(D') = \sqcup$, $\psi(D_o) = \sqcup$, $\psi(D_1) = \sqcap$. Alors pour toute fonction $f : D \to D'$, on a :

$$f = f_o \amalg f_1 \quad \text{avec}$$

$$f_o = \sqcup\{[e,e']_{min(f)} : e' < f(e),\ e \in D_o\}$$

$f_1 = \sqcup\{[e,e']^*{}_{\min(f)} : e' < f(e),\ e \in D_1\}$

$[e,e']^*{}_{\min(f)} : z \to e'$ si $z \leq e$
$\qquad\qquad\qquad \bot$ sinon

en notant $z \leq e$ le fait que e approxime z (par le "haut"). □

Démonstration

On a $f_o = f_{|D_o}$ par le même raisonnement qu'en 3.14.

D'autre part $\psi(D_1) = \sqcap$ et $x \in D_1 \Rightarrow x = \sqcap\ s(x)$ donc
$f(x) = \sqcup f(s(x)) = \sqcup\{f(u) :\ u \geq x\}$

Notez que $x \sqsubseteq y \Rightarrow s(y) \subseteq s(x)$ car $y = \sqcap s(y) \sqsupseteq \sqcap s(x) \Rightarrow f(y) \sqsubseteq f(x)$ par monotonie

On a pour $x \in D_1$

$f_1(x) = \sqcup\ \{[e,e']^*{}_{\min(f)}(x) : e' < f(e)\}$
$\quad = \sqcup\ \{e' : \exists e\ \ x \leq e \quad e' < f(e)\}\ \sqcup \min(f)$

(puisque $x \leq e \Rightarrow x \sqsubseteq e \quad e' < f(e) \sqsubseteq f(x) \Rightarrow e' < f(x)$ par monotonie des spectres)

$= \sqcup\{e' : e' < f(e)\} \sqcup \min(f) = f(x)$

Donc $f_1 = f_{|D_1}$.

Donc $f = f_0 \perp\!\!\!\perp f_1$ i.e. f est obtenue par recollement de f_0 et f_1. On note $[D \to D'] = [D_o \to D'] * [D_1 \to D']$ □

Identifications canoniques :

Si D_o, D_1, D' sont comme dans le lemme précédent, il y a une façon canonique d'identifier $[D_o \to D']$ et $[D_1 \to D']$ à des parties de $[(D_o + D_1) \to D']$:
il suffit d'en faire des échelons généralisées:

$\forall f \in [D_o \to D'] \quad f \to \{\!\!\{ D_o, f \}\!\!\}$ où

$\{\!\!\{ D_o, f \}\!\!\} : x \to f(x)$ si $x \in D_o$
$\qquad\qquad\qquad \bot$ sinon

même chose pour $[D_1 \to D']$.

Avec ces identifications, le lemme précédent s'écrit maintenant :

$$f = f_o \sqcup f_1 \qquad \square$$

Ceci permet d'exprimer $[(D_o + D_1) \to D']$ en fonction de $[D_o \to D']$ et $[D_1 \to D']$.

4.19 <u>Remarque</u> : Si X, A, B sont des ensembles $f : X \to A + B$ détermine deux fonctions

$g_A = f|_{\{x \in X : f(x) \in A\}} : X \to A$

$g_B = f|_{\{x \in X : f(x) \in B\}} : X \to B$

Si X,A,B sont des faisceaux et f régulière alors g_A et g_B sont régulières car $\forall x \in X \quad f(x) \in A \Rightarrow \forall \sigma \in \beta(x)\ f \circ \sigma \in \mathcal{F}(A)$ du fait que $\lim_{A+B} = \lim_A \amalg \lim_B$ par définition du coproduit (chap. I). Donc on peut injecter $[X \to A+B]$ dans $[X \to A] \times [X \to B]$ et tout faisceau sur $[X \to A] \times [X \to B]$ détermine un faisceau sur $[X \to A + B]$, qui fait de $[X \to A + B]$ un sous-faisceau de $[X \to A] \times [X \to B]$ $\square$

<u>La Notion de bifaisceau et le foncteur Nolrat</u>

Pour mettre en oeuvre notre construction nous avons besoin d'une notion de <u>bifaisceau</u>.

<u>Définition</u> : <u>La catégorie BF des bifaisceaux</u>.

Un <u>objet</u> de la catégorie <u>BF</u> des <u>bifaisceaux</u> est un ensemble partiellement ordonné $X \in \mathcal{U}$ possédant T muni d'une double structure de faisceau :

(i) une structure de <u>faisceau inférieur</u> $<\lambda, \beta>$ avec $\lambda = \sqcup \circ \mathrm{Im}$ telle que X muni de $<\lambda, \beta>$ est un objet de <u>Interp</u>

(ii) une structure de <u>faisceau supérieur</u> $<\lambda^1, \beta^1>$ avec $\lambda^1 = \sqcap \circ \mathrm{Im}$, telle que X muni de $<\lambda^1, \beta^1>$ est un objet de <u>D Cont</u>

Un <u>morphisme</u> $f : X \to Y$ de la catégorie <u>BFc</u> est une fonction régulière du faisceau inférieur de X dans le faisceau inférieur de Y. (Les problèmes de composition sont passés sous silence).

<u>Remarque</u> : On présente ici les bifaisceaux comme des objets ayant une double structure de faisceau. Si on met de côté les morphismes, cette présentation n'empèche pas de les voir comme de

vrais faisceaux : il suffit de mélanger les deux fibres de chaque point. Ceci permet d'étendre d'emblée les notions de <u>produit</u> et <u>coproduits</u> aux faisceaux et bifaisceaux.

On définit le foncteur d'oubli τ suivant

$$\tau : BF \to \text{Interp}$$

comme étant le foncteur oubliant la structure de faisceau supérieur des bifaisceaux. On notera si $X \in \underline{BF}$ $\tau(D) = \underline{D_{inf}} = \underline{D}$

De la même manière on peut examiner comment utiliser les échelons de Nolin pour construire un foncteur : Soit D un bifaisceau, $\Delta = [D_{inf} \to D_{inf}]$ l'espace des fonctions régulières de D dans D pour le faisceau inférieur(1). Nous voulons faire de Δ un objet de D Cont. Supposons $T \in D$; $f \in \Delta \Rightarrow f = \sqcap \{\langle e,e'\rangle_T : e' \sqsupseteq f(e)\} = \sqcap s(f)$

$$s(\sqcap_i f_i) = \sqcap \{\langle e,e'\rangle : e' \sqsupseteq (\sqcap_i f_i)(e)\}$$

Renversant l'ordre :

$$s(\sqcup_i f_i) = \sqcup \{\langle e,e'\rangle : e' \sqsubseteq (\sqcup_i f_i)(e)\}$$

en gardant les mêmes notations pour les échelons.

Si $e \in D$ est rationnel $(\sqcup_i f_i)(e) = \sqcup_i (f_i(e))$ d'après 3.2 dès que D_{inf} est un foe normalisé dont la fonction spectre définit un fom. Alors $s(\sqcup_i f_i) = \sqcup \{\langle e,e'\rangle : e' \sqsubseteq \sqcup_i f_i(e)\}$

On veut $s : \Delta \to P(\Delta)$ Scott-continue i.e. $e' \sqsubseteq \sqcup_i f_i(e) \Leftrightarrow \exists i \; e' \sqsubseteq f_i(e)$ dès que f_i est dirigée, ce qui est impliqué par "e' est compact au sens de Scott".

Donc si e rationnel et e' compact au sens de Scott, f_i dirigée

$$s(\sqcup_i f_i) = \sqcup\{\langle e,e'\rangle : e' \sqsubseteq (\sqcup_i f_i)(e)\} = \text{(e rationnel)}$$

$$\sqcup\{\langle e,e'\rangle : e' \sqsubseteq \sqcup_i (f_i(e))\} = \text{(e' compact)}$$

$$\sqcup\{\langle e,e'\rangle : \exists i \; e' \sqsubseteq f_i(e)\} =$$

$$= \sqcup_i (\sqcup\{\langle e,e'\rangle : e' \sqsubseteq f_i(e)\}) = \sqcup_i s(f_i)$$

(1) Il s'agit donc des BF-morphismes.

$\Rightarrow$ s Scott-continue en $f = \bigsqcup_i f_i$ $\Rightarrow$ il existe une structure de fom qui fait de $[D_{\underline{inf}} \to D_{\underline{inf}}]$ un objet de <u>DCont</u> (chap II).

Il reste deux points à préciser:

(i) Dans une décomposition en échelons de Nolin, on peut se restreindre aux $<e,e'>$ où e est rationnel (Prop. 3.28), si la fonction régulière est monotone croissante.

(ii) Une image d'un point e, $f(e)$, étant donnée, on peut la reconstruire comme borne d'éléments compacts si le faisceau supérieur du bifaisceau d'arrivée est algébrique (chap. II)

D'où on tire :

Si <u>Foe</u> est une catégorie dont les objets sont des foe normalisés dont la fonction spectre définit un fom

<u>BFt</u> la sous-catégorie pleine de <u>BF</u> dont les objets ont pour faisceau inférieur une structure de <u>Foe</u>, et faisceau supérieur un faisceau algébrique, et sont munis d'un T,

alors la discussion qui precede fournit une fonction

$$\eta : |\underline{BFt}| \times |\underline{BFt}| \to |\underline{DCont}|$$

ou mieux :

$$\hat{\eta} : |\underline{Foe}| \times |\underline{BFt}| \to |\underline{DAlg}|$$
$$(X,Y) \to [X \to Y]$$

<u>Remarque</u> : Un point essentiel de cette discussion est que le deuxième argument de η est dans <u>BFt</u> et n'est confiné ni dans <u>DAlg</u> ni dans <u>Foe</u> : on doit utiliser en effet deux fonctions limites : la limite du faisceau supérieur pour reconstruire l'espace et la limite du faisceau inférieur pour calculer sur les fonctions $f : X \to Y$

Nous voulons maintenant étendre η en un foncteur

$$N : \underline{Foe} \times \underline{BFt} \to \underline{DAlg}$$

comme nous l'avons fait pour G et H. Ceci se fait sans difficulté suivant la même démarche. Le seul problème est de composer les morphismes de <u>Foe</u> : nous prenons pour morphismes les fonctions régulières simples munies de la composition régulière. On retrouve donc la catégorie <u>Foen</u>.

D'où on tire :

4.20 Proposition

Soit BFt la sous-catégorie pleine de BF dont les objets ont pour faisceau inférieur une structure de Foen, et faisceau supérieur un faisceau algébrique, et sont munis d'un T.

Alors

$$N : \text{Foen} \times \text{BFt} \to \text{DAlg}$$

est un foncteur.

Notation : Nous noterons Nolrat le foncteur N(pour "faisceau associé aux échelons de Nolin rationnels").

4.21 Théorème : La catégorie BFt est fermée par produit et coproduit. Elle possède des équaliseurs mais généralement pas de coéqualiseurs.

Démonstration : On applique le théorème 4.3 pour Foen et on procède comme pour Fais.

Définition : Classe des bifaisceaux commodes.

La classe C des bifaisceaux commodes est constituée par les bifaisceaux X BF tq que

(i) le faisceau supérieur est dans DAlg

(ii) le faisceau inférieur est dans Foen

(iii) $T \in X$.

Il s'agit donc de la classe des objets de la catégorie BFt

On n'impose aucune relation entre le faisceau supérieur et le faisceau inférieur.

Définition fonctions d'oubli et d'injection.

On définit les fonctions suivantes

(i) Inj : $|\text{DAlg}| \to C$ qui rajoute à chaque objet de DAlg un faisceau inférieur trivial (toujours dans Foen) $(x \to \{x\})$

(ii) Sup : $C \to |\text{DAlg}|$ qui oublie le faisceau inférieur des bifaisceaux.

(iii) Inf : $C \to |\text{Foen}|$ qui oublie le faisceau supérieur des bifaisceaux.

Si X,Y sont des objets de quelque catégorie A, notons $\hom_A(X,Y)$ l'espace des A-morphismes de X dans Y.

Si $X, Y \in C$ sont des bifaisceaux commodes posons :

$[\,.\to.\,](X,Y) = \{f : X \to Y \mid f$ est régulière pour chacun des faisceaux de $X\}$ =

$hom_{\underline{DAlg}}(Sup(X),\ Sup(Y)) \cap hom_{\underline{Foen}}(Inf(X),\ Inf(X),\ Inf(Y))$

Alors on peut énoncer.

4.22 Théorème. $[\,.\to.\,]$ (X,Y) peut être muni d'une structure de bifaisceau commode

Démonstration

(i) $[\,.\to.\,](X,Y)$ peut être considéré comme un objet de DAlg :

En effet si $X \in$ DAlg, $Y \in$ DAlg, alors $hom_{\underline{DAlg}}(X,Y)$ est dans DAlg au moyen du foncteur G, et pour cela on utilise les échelons de Scott. Maintenant si on restreint $hom_{\underline{DAlg}}(X,Y)$ aux Foen-morphismes, il convient de vérifier qu'il reste assez d'échelons de Scott pour qu'on puisse appliquer la même procédure. En fait nous allons vérifier que tous les échelons de Scott de DAlg sont des Foen-morphismes, et si on perd des fonctions en passant de $hom_{\underline{DAlg}}(Sup(X),\ Sup(Y))$ à $[\,.\to.\,](X,Y)$ ce ne sont pas eux. Car :

(a) les échelons de Scott sont des Foen-morphismes (pour l'ordre inverse). Cela est peut-être plus facile à voir si on renverse tous les ordres. si $e' \neq \bot$, on a

$[e,e']_\bot(\sqcap\sigma) = e'$ ssi $e \ll \sqcap\sigma$

(où $\ll$ dénote la way-below) ssi $\forall s \in \sigma\ e \ll s$ (les spectres de la way-below sont des sections commençantes) ssi

$\forall s \in \sigma\ [e,e']_\bot(s) = e'$ ssi

$$\sqcap[e,e']_\bot(\sigma) = e'$$

et ceci pour tout système approximant du faisceau de Foen

(b) la reconstruction des fonctions régulières au moyen des échelons de Scott marche (point par point)

(c) donc on peut appliquer la même technique qu'en 3.24 pour $[\,.\to.\,](X,Y)$

D'où une structure de faisceau algébrique ($\in$ DAlg) sur $[.\to.]\,(X,Y)$, d'où une structure Inj $([.\to.]\ (X,Y)) \in C$. $\square$

Notation : Dans la suite on notera $[.\to.]$ cette application de $C \times C$ dans C :

$$X,Y \to [.\to.]\ (X,Y)$$

Dans le même ordre d'idées on peut remarquer que :

4.23 Lemme : Soit X un bifaisceau tel que le faisceau supérieur de X est dans DAlg et le faisceau inférieur dans Foen. Alors si on définit $\forall e, e' \in X$

$\langle e,e' \rangle_T$ est un échelon de Nolin algébrique $\Leftrightarrow$ e est compact pour le faisceau dans DAlg ,

chaque échelon de Nolin algébrique est un échelon de Scott pour l'ordre inverse sur le faisceau supérieur de X. $\square$

Démonstration :

$$\langle e, e' \rangle_T(x) = \begin{cases} e' & \text{si } x \sqsubseteq e \\ T & \text{sinon} \end{cases}$$

Pour l'ordre habituel, sur un fom,

$$[e,e']_\perp\ (x) = \begin{cases} e' & \text{si } e \prec x \\ \perp & \text{sinon} \end{cases}$$

où $e \prec x$ dénote "e est un approximant de x par le bas".

Renversant les ordres :

$$[e,e']^*_T(x) = \begin{cases} e' & \text{si } e \text{ approxime } x \text{ par le haut} \\ T & \text{sinon} \end{cases}$$

Si on prend e compact pour le faisceau dans DAlg, on a $x \sqsubseteq e \Leftrightarrow e$ approxime x par le haut, d'où l'identité entre $\langle e,e' \rangle_T$ et $[e,e']^*_T$ $\square$

Les échelons de Nolin sont toujours réguliers pour le faisceau inférieur (cf. chap. III) ; s'ils sont algébriques alors ils sont réguliers aussi pour le faisceau supérieur.

Si on se place dans C, ceci nous amène à le préciser de la façon suivante.

$A_o = D \in C$

$A_1 = D + [A_o \to A_o]$

où $[A_o \to A_o] = \mathrm{Inj}(\mathrm{hom}_{\mathrm{Foen}}(\mathrm{Inf}(D), \mathrm{Inf}(D)), \mathrm{Nolrat})$

où le faisceau injecté dans $\underline{C}$ est fourni par le foncteur Nolrat.

Puis par récurrence

$A_{n+2} = D + [A_{n+1} \to A_{n+1}] = D + \Delta_{n+1}$ où $\Delta_{n+1} = [A_{n+1} \to A_{n+1}] =$

$[D + \Delta_n \to D + \Delta_n] =$

$[D \to D + \Delta_n] \times [\Delta_n \to D + \Delta_n] =$

$([D \to D] * [D \to \Delta_n]) \times ([\Delta_n \to D] * [\Delta_n \to \Delta_n])$

On prend toujours

$[D \to D] = \mathrm{Inj}(\mathrm{hom}_{\mathrm{Foen}}(\mathrm{Inf}(D), \mathrm{Inf}(D)), \mathrm{Nolrat})$

$[D \to \Delta_n] = M(D, \Delta_n) = [. \to .]\ (D, \Delta_n)$

muni du faisceau fourni par théorème 4.22

$[\Delta_n \to D] = M(\Delta_n, D)$ même chose qu'au cas précédent

$[\Delta_n \to \Delta_n] = [. \to .]\ (\Delta_n, \Delta_n) =$
$= \mathrm{Inj}(\mathrm{hom}_{\mathrm{DAlg}}(\Delta_n, \Delta_n), G)$.

Ce que nous pouvons résumer ainsi

$A_o = D$

$A_1 = D + \mathrm{Inj}(\mathrm{hom}_{\underline{\mathrm{Foen}}}(\mathrm{Inf}(D), \mathrm{Inf}(D)), \mathrm{Nolrat})$

$A_{n+2} = D + M^*(A_{n+1}, A_{n+1})\ \forall n \in N$

avec

$M^*(A_{n+1}, A_{n+1}) =$

$\{(\mathrm{Inj}(\mathrm{hom}_{\underline{\mathrm{Foen}}}(\mathrm{Inf}(D), \mathrm{Inf}(D)), \mathrm{Nolrat}) * M(D, \Delta_n)) \times$

$\times [M(\Delta_n, D) * \mathrm{Inj}([. \to .]\ (\Delta_n, \Delta_n), G)]\}$

Il reste à montrer que tous ces objets sont dans C. On se donne aussi

$\forall n\ \ i_n : A_n \to A_{n+1},\ j_n : A_{n+1} \to A_n$

$$i_o(x) = x$$

$$i_{n+1}(x) = x \quad \text{si} \quad x \in D$$
$$\qquad i_n \circ x \circ j_n \quad \text{si} \quad x \in [A_n \to A_n] = \Delta_n$$

$$j_{n+1}(y) = y \quad \text{si} \quad y \in D$$
$$\qquad j_n \circ y \circ i_n \quad \text{si} \quad y \in [A_{n+1} \to A_{n+1}] = \Delta_{n+1}$$

$$j_o(y) = y \quad \text{si} \quad y \in D$$
$$\qquad T \quad \text{sinon}$$

Le rôle de l'élément indéfini est joué par l'élément T (indéterminé).

4.24 <u>Lemme</u> : tous les objets A_n du schéma ci-dessus sont des bifaisceaux commodes.

<u>Démonstration</u> : Il suffit de vérifier que l'étoile dans $[D \to D] * [D \to \Delta_n]$ et dans $[\Delta_n \to D]*[\Delta_n \quad A_n]$ nous fournit bien des objets de $\underline{C}$. Pour cela on utilise les échelons de Scott comme en 4.18 (pour l'ordre inverse) et on précède comme au théorème précédent.

On vérifie aussi que les flèches du schéma ci-dessus sont bien définies :

(a) $i_o \in [\cdot \to \cdot] \quad (A_o, A_1)$

(b) plus généralement les i_n et j_n ont bien le domaine et codomaine désiré. Vérifions le pour i_{n+1}.

$$x \in [A_n \to A_n] \quad ? \quad i_{n+1}(x) \in [A_{n+1} \to A_{n+1}]$$

$y \in A_{n+1} = D + [A_n \to A_n]$. Calculer $(i_{n+1}(x))(y)$

<u>1er cas</u>: $y \quad D$

$$(i_{n+1}(x))(y) = (i_n \circ x_o\, j_n)(y) = (i_n \circ x)(y) =$$

On a x régulier en $y \in D \subset A_n$

i_n conserve la régularité (conserve les sups)

$\Rightarrow i_n \circ x$ régulier en $y \Rightarrow i_n \circ x \circ j_n$ régulier en y.

<u>2ème Cas</u> : $y \in [A_n \to A_n]$

$$(i_{n+1}(x)(y)) = (i_n \circ x \circ j_n)(y) = (i_n \circ x \circ j_n)(\sqcap s(y))$$

$= i_n \circ x\ (\sqcap j_n(s(y)))$ (j_n conserve les infs)

$= i_n(\sqcap x(j_n(s\ y))))$ (x DAlg-morphisme)

$= \sqcap i_n(x(j_n(s(y)))) =$

$= \sqcap(i_n \circ x \circ j_n)(s(y)) = \sqcap(i_{n+1}(x))(s(y))$

Ceci pour le faisceau supérieur.

Il n'y a rien à vérifier pour le faisceau inférieur de $[A_n \to A_n]$ (sa Foen-structure) puisque tous les éléments sont rationnels (lemme 3.2).

On refait le même raisonnement pour j_n. Donc tous les objets du schéma d'approximation précédent sont parfaitement définis. (Cette démonstration s'appuie lourdement sur le fait que les i_n et j_n sont des doubles connections de Galois).

Considérons la limite du diagramme précédent dans la catégorie Set des ensembles :

$$A_\infty = \{(x_n)_{n \in N} : x_n = j_n(x_{n+1})\ \forall n \in N\}$$

On note que pour toute suite $(x_n)_{n \in N} \in A_\infty$, pour tout n entier :

$x_{n+1} \in$ partie D de $A_{n+1} \Rightarrow x_n \quad x_{n+1}$

$x_{n+2} \in$ partie fonctionnelle de $A_{n+2} \Rightarrow$

$$x_{n+1} = j_n \circ x_{n+2} \circ i_n$$

$x_1 \in$ partie fonctionnelle de $A_1 \Rightarrow x_o = T$

Donc toujours dans Set cette limite (projective) peut être écrite (cf. [MCL]) :

$$A_\infty = \mathrm{Cone}(*, F_o) + \mathrm{Cone}(*, F')$$

(puisque l'union est le coproduit dans Set)

où F_o est le diagramme réduit à D

F' est le diagramme

$$\{T\} \leftarrow [A_o \to A_o] \underset{j_1}{\overset{i_1}{\rightleftarrows}} [A_1 \to A_1] \underset{j_2}{\overset{i_2}{\rightleftarrows}} [A_2 \to A_2] \underset{j_3}{\overset{i_3}{\rightleftarrows}} \cdots$$

Comme Cone $(*, F_o) = D$, il vient

$A_\infty = D + \mathrm{Cone}(*, F') = D + \varprojlim \Delta_K$ où

$\forall k \in N \quad \Delta_o = T$

$$\Delta_1 = [A_o \to A_o] = [D \to D]$$
$$\Delta_{k+1} = [A_k \to A_k]$$

et les flèches définies comme ci-dessus.

Il s'agit maintenant de relever cet objet dans la classe des bifaisceaux commodes, toujours selon la technique de Mac Lane, pp 106-107. La structure de bifaisceau sur A_∞ sera définie en prenant le coproduit (somme) d'un bifaisceau sur la partie D, dérivé de la structure de D, et d'un bifaisceau sur la partie $\underleftarrow{\lim}\ \Delta_k$ obtenu en remarquant que puisqu'on peut mettre tous les Δ_k dans <u>DAlg</u>, le diagramme F' est dans <u>DAlg</u> qui est complète (Théorème 4.3), donc $\underleftarrow{\lim}\ \Delta_k$ peut être calculée dans <u>DAlg</u>, ce qui induit une structure de <u>DAlg</u> sur cette partie de A_∞, qu'on va injecter dans BF par Inj.

Plus précisément

faisceau $(D + \Delta_\infty)$ = Inj(Inf(D)) ⨿ faisceau (Δ_∞)

avec :

. Inj(Inf(D)) veut dire que le faisceau supérieur est le faisceau trivial (tout élément est rationnel)

. faisceau (Δ_∞) est composé d'une <u>DAlg</u> structure (celle de Δ_∞) qui constitue le faisceau supérieur et d'un faisceau inférieur qui est l'o.e. trivial (Tout élément est rationnel).

On peut donc injecter A_∞ dans la classe des bifaisceaux commodes.

D'où la structure de bifaisceau commode de A_∞

$$A_\infty = D + \underleftarrow{\lim}\ \Delta_k$$

4.25 <u>Lemme</u> = $\forall x \in \underleftarrow{\lim}\ \Delta_k$, x détermine une fonction (partielle) régulière de A_∞ dans A_∞.

<u>Démonstration</u> : si $x \in \underleftarrow{\lim}\ \Delta_k$, posons

$$[x] : y \to \prod_n x_{n+1}(y_n) \text{ si } y \in (\underleftarrow{\lim}\ \Delta_k) \cup \mathrm{Rat}(D)$$
$$\bigsqcup \{[x](u) : u \in s(y)\} \text{ sinon}$$

(Ici Rat(D) dénote l'ensemble des rationnels pour la <u>Foen</u>-structure de D, i.e. le faisceau inférieur de D).

(i) $\forall a \in D$ $[x]$ est régulier en a par définition

$$[x]\ (a) = \bigsqcup\{[x]\ (u) : u \in s(a)\}$$

(ii) $\forall a \in \varprojlim \Delta_k$ $\forall \sigma \in \beta(a)$ il vient

$\prod\ [x]\ (\sigma) = \prod_n(\prod x_{n+1}(\sigma_n)) =$ (Fubini)

$\prod_n(\prod x_{n+1}(\sigma_n)) = (x_{n+1}$ régulière si $n \neq 0$ et $\sigma_o = \{T\})$

$= \prod_n(x_{n+1}(\prod\sigma_n)) = \prod_n x_{n+1}(a_n) = [x]\ (a)$ (circulation).

d'où le lemme □

On peut donc injecter $\varprojlim \Delta_k$ dans l'espace des fonctions régulières de A_∞ dans A_∞.

Réciproquement :

4.26 <u>Lemme</u> : $\forall f : A_\infty \to A_\infty$ il existe $]f[\in \varprojlim \Delta_k$ défini par

$]f[_o = T$

$]f[_{n+1} = \lambda y \in A_n.(f(y))_n$ □

<u>Démonstration</u> : ? $]f[\in \varprojlim \Delta_k$

c'est-à-dire a-t-on $]f[_n = j_n(]f[_{n+1})$?

a) $j_o(]f[_1) = T$ car $]f[_1 \in \Delta_1$

b) $j_{n+1}(]f[_{n+2}) = j_n \circ (]f[_{n+2}) \circ i_n =$

$= j_n \circ \lambda y \in A_{n+1}(f(y))_n \circ i_n =$

(puisque $A_n \xrightarrow{i_n} A_{n+1} \xrightarrow{]f[_{n+2}} A_{n+1} \xrightarrow{j_n} A_n$)

$= \lambda y \in A_n.\ j_n((f(i_n(y)))_{n+1}) =$

$= \lambda y \in A_n.\ j_n((f(y))_{n+1}) = \lambda y \in A_n.(f(y))_n$

$=]f[_{n+1}.$ q e d □

4.27. <u>Lemme</u> $\forall f : A_\infty \to A_\infty$ régulière on a $[]\ f\ [] = f$, i.e. les deux fonctions ont même domaine de définition et sont égales.

Démonstration : on applique le lemme 4.25.

Il est aisé de voir que puisque $x \in \lim \Delta_k$, $[x] : y \to \prod_n x_{n+1}(y_n)$;

il vient si $y \in \varprojlim \Delta_k$ $[]f[] : y \to \prod_n f_{n+1}(y_n) =$

$$\prod_n (\lambda y \in A_n . (f(y))_n)(y_n) =$$

$$\prod_n (f(y_n))_n = \prod_{n,m} (f(y_m))_n = \prod_m f(y_m) = f(y)$$

Maintenant soit $a \in D$. Il vient, si a est rationnel,

$$[]f[]\ (a) = \prod_n f_{n+1}(a_n) =$$

$$\prod_n (\lambda y \in A_n . (f(y))_n)(a_n) =$$

$$\prod_n (f(a))_n = f(a)$$

puisque $a = (a,a,\dots) \in A_\infty$

Maintenant si a est irrationnel, il vient :

$$[]f[]\ (a) = \bigsqcup \{[]f[]\ (u) : u \in s(a)\} = \bigsqcup \{f(u) : u \in s(a)\}$$

$$= f(a) \qquad \square$$

□

Donc on a pu injecter $[A_\infty \to A_\infty]$ dans $\varprojlim \Delta_k \subset D + \varprojlim \Delta_k = A_\infty$

On peut donc énoncer :

4.28 Théorème : Soit D un bifaisceau commode. Alors le schéma d'approximation de Wadsworth fournit dans Set une limite projective A_∞ qui est telle que

(i) $A_\infty = D + \varprojlim \Delta_k$, la limite des Δ_k pouvant être calculée dans DAlg, ce qui munit A_∞ d'une structure de faisceau

(ii) Si $[A_\infty \to A_\infty]_p$ est l'espace des fonctions régulières partielles de A dans A, alors il existe une bijection $\varprojlim \Delta_k \leftrightarrow [A_\infty \to A_\infty]_p$.

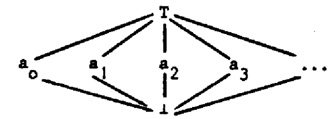
En particulier A_∞ contient l'espace de toutes les fonctions régulières de A_∞ dans A_∞. □

4.29 Proposition : le modèle E_∞ de Wadsworth est un cas particulier de domaine A_∞

:ales.

<u>Démonstration</u> : Nous avons $E_\infty = D + [E_\infty \to E_\infty]$ avec D étant un treillis complet plat

$D =$



D est trivialement un bifaisceau commode. Il suffit de noter que la structure de treillis continu de $[D \to D]$ est isomorphe à celle de Nolrat (D,D) (Les échelons de Scott et de Nolin sur D sont les

$$[e,e']_\perp(x) = \begin{cases} e' & \text{si } x \in \{e,T\} \\ \perp & \text{sinon} \end{cases}$$

$$\langle e,e' \rangle_T \ (x) = \begin{cases} e & \text{si } x \in \{e,\perp\} \\ T & \text{sinon} \end{cases} \ )$$

Et cet isomorphisme se propage tout au long de la construction. Donc E_∞ est isomorphe à quelque domaine A_∞. □

Nous avons ainsi donné une solution à notre problème de généraliser le modèle E_∞ de façon à y inclure des atomes plus complexes. Il convient d'apporter deux remarques importantes à notre construction

<u>Remarque 1</u> : Le domaine A_∞ n'a pas été obtenu comme limite dans une catégorie (on s'est simplement servi d'une classe C des bifaisceaux commodes, et telle quelle C peut difficilement fournir une catégorie). On s'est donc servi de l'idée de faisceau sagittal plutôt que de la notion elle-même. Vue sous cet angle, la théorie des faisceaux (sur un ensemble) nous parait de nature à initialiser une théorie analytique des équations aux domaines, analogue à ce qu'on fait pour les équations différentielles en analyse.

<u>Remarque 2</u> : Notre première approximation de <u>Foen</u>, au début de ce chapitre, était <u>Interp</u>. Nous aurions pu tout aussi bien prendre <u>Alg</u> à la place de <u>Foen</u>. Nous aurions ainsi obtenu des bifaisceaux "bialgébriques" et C nous aurait de suite fourni une catégorie en prenant pour morphismes les fonctions à la fois régulières pour le faisceau inférieur et le faisceau supérieur. La même construction peut alors être mise en oeuvre, en remplaçant partout <u>Foen</u> par <u>Alg</u>. De tels faisceaux bialgébriques, lorsque le support

est un treillis complet, sont considérés par Hofmann, Mislove et Stralka 1974 dans leur étude sur la dualité de Pontryaguine ; voir aussi Hofmann 1979.

Une raison pour ne pas se restreindre au cas où le faisceau inférieur est un objet de Alg, est que, dans le cas des treillis complets, les spectres qui nous intéressent ne sont pas dirigés. Par exemple, dans un langage de programmation, on peut se donner uniquement les types $0,1,2,3,\ldots$ et N ; on doit alors nettement distinguer entre les rationnels $(0,1,2,\ldots)$ et les non-rationnels $(N = \bigcup_n \{n\})$. On a la même chose avec les "valeurs" de vérité booléen, vrai et faux.

Loi de composition sur A_∞ :

On peut définir sur A_∞ une loi de composition qui correspond au calcul sur les types dans le λ-calcul typé ([CUR]. [HIN]) ou encore à la loi de composition sur les algorithmes de Nolin.

Définition : $\forall x,y \quad A$ on définit

$x \,.\, y = \sqcap\{z \in A_\infty : \langle y, z\rangle \sqsupseteq x\}$

Cette loi est a priori partielle mais on vérifie de suite que $x.y =$ si $x \in D$ alors T sinon $[x]\ (y)$ fsi où $[x]$ est défini comme dans le lemme 4.25.

Exemple: soit $a \in A_\infty$

(i) $a \in D$

$a.a = \sqcap\{z \in A_\infty : \langle a,z\rangle \sqsupseteq a\} = \sqcap\, \emptyset = T$

(ii) $a \in \Delta_\infty \Rightarrow a.a = [a]\ (a)$

$[a]\ (a) = \sqcap\{z \in A_\infty : \langle a,z\rangle \sqsupseteq a\}$ □

Interprétation du λ-calcul dans A_∞ : Procédures universelles :

On note que sur D tout échelon de Nolin est régulier pour le faisceau inférieur, et que sur $\Delta_\infty = \varprojlim \Delta_k$ tout échelon de Nolin algébrique est régulier (puisque Δ_∞ est algébrique). Ceci nous amène à poser :

Soit $K(\Delta_\infty) =$ le noyau de Δ_∞ pour le faisceau supérieur (= l'ensemble des éléments compacts), et soit $E = K(\Delta_\infty) \cup D$.

On peut alors définir une interprétation du langage Λ du λ-calcul dans A_∞ par une fonction :

$\psi : \Lambda \to A$

<u>1er cas</u> : $\psi(N) \in D$ alors par lemme 1.2(iii) on a :

$\psi((\lambda x.M)N) =$

$(\sqcap\{< a, \psi_{x \leftarrow a}(M) >_T : a \in K(\Delta_\infty)\})(\psi(N)) =$

$= \sqcap\{< a, \psi_{x \leftarrow a}(M) >_T(\psi(N)) : a \in K(\Delta_\infty)\}$

$= \sqcap\{\psi_{x \leftarrow a}(M) : \psi(N) \sqsubseteq a, a \in K(\Delta_\infty)\}$

$=$ (la fonction $\lambda a \in \Delta_\infty . \psi_{x \leftarrow a}(M)$ est continue)

$= \psi_{x \leftarrow \psi(N)}(M) = \psi([^N/x] M)$

<u>2ème cas</u> : $\psi(N) \in D$. Alors

$\psi((\lambda x.M)N) =$

$(\sqcap\{< a, \psi_{x \leftarrow a}(M) >_T : a \in D\})(\psi(N)) =$

$=$ (si $\psi(N)$ est rationnel pour le faisceau inférieur de D)

$\sqcap\{< a, \psi_{x \leftarrow a}(M) >_T(\psi(N)) : a \in D\}$

$= \psi_{x \leftarrow \psi(N)}(M) = \psi([^N/x] M)$ □

Ceci indique que dans notre modèle la β-réduction n'est satisfaite que dans certaines circonstances. Comme la substitution et la réduction sont les seuls mécanismes de calcul dont on dispose dans les langages de programmation, et que le λ-calcul peut être vu comme un langage de programmation idéalisé il convient de repérer dans quelles circonstances les calculs que l'on fait dans le langage au moyen de ces mécanismes, ont une signification dans le modèle que l'on s'efforce de mécaniser. Et de ne définir des mécanismes de calcul comme légitimes que dans la mesure où on les a "remontés" du modèle de cette façon.

En résumé : au lieu de définir d'abord un langage et son calcul, pour en chercher ensuite un modèle, on construit d'abord une structure mathématique qui réflète bien les phénomènes informatiques que l'on a en vue, pour construire ensuite une théorie de cette structure, qui fournira l'implémentation du langage. Cette façon de voir a été indiquée par L. Nolin, et nous parait beaucoup plus riche d'applications pratiques que celle qui consiste à ramener la sémantique des programmes à quelque homomorphisme dans une catégorie. La notion de faisceau devrait servir d'instrument privilégié dans cette direction de recherches.

Remarque

Les bifaisceaux commodes comme catégorie ; les BFc-objets

Pour faire de la classe $\underline{C}$ des bifaisceaux commodes une catégorie, il faudrait que la composition régulière (dans Foen) de fonctions simples régulières pour le faisceau supérieur, fournisse des fonctions régulières pour le faisceau supérieur. On pourrait alors poser $hom(X,Y) = [. \to .] (X,Y)$.

Soit $x \in X$, $s_{<<}(x)$ = spectre de x dans DAlg, $\forall j \in s_{<<}(x)$ $j = \sqcup\{u_{j,k} : k \in S(j)\}$ (la famille des $u_{j,k}$ est un système approximant de j pour la Foen-structure). Posons pour simplifier $J = s_{<<}(x)$. Alors si h, g sont régulières $h : X \to Y$, $g : Y \to Z$, il viendrait

$$o(h,g)\ (x) = o(f,g)\ (\sqcap\, s_{<<}(x)) =$$

$$o(h,g)\ (\sqcap_{j \in J}\ (\sqcup\{u_{j,k} : k \in S(j)\})) =$$

(si complète distributivité, [SCS] pp 59, ou Birkhoff 1961 pp 146)

$$o(h,g)\ (\sqcup_{f \in M}\ (\sqcap_{j \in J}\ u_{j,f(j)})) = (\text{ si } \{\sqcap_{j \in J} u_{j,f(j)} : f \in M\}$$

est un Foen-système approximant de x)

$$= \sqcup_{f \in M} (o(h,g)(\sqcap_{j \in J} u_{j,f(j)})) =$$

$$\sqcup_{f \in M} g(h(\sqcap_{j \in J} u_{j,f(j)}) = (\text{si } \{u_{j,f(j)}\ j \in J\}$$

est filtrante) $\sqcup_{f \in M} \sqcap_{j \in J} g(h(u_{j,f(j)})) =$

(complète distributivité)

$$\sqcap_{j \in J} \sqcup\{g(h(u_{j,k})) : k \in S(j)\} =$$

$$\sqcap_{j \in J} \sqcup\{o(h,g)\ (u_{j,k}) : k \in S(j)\} =$$

$$\sqcap_{j \in J} o(h,g)(j) = \sqcap o(h,g)(s_{<<}(x))$$

D'où la régularité de $o(f,g)$ pour $\underline{\text{DAlg}}$.

Donc en résumé il faudrait les conditions

(i) X est complétement distributif

(ii) $\{ \prod_{j \in J} u_{j,f(j)} : f \in M \}$ est un système approximant de $x \in X$ pour le faisceau inférieur.

(iii) $\forall f \in M$ $\{u_{j,f(j)}\}_{j \in J}$ est filtrante. Noter que les conditions (ii) et (iii) introduisent une relation entre les faisceaux inférieur et supérieur de X.

Muni de ceci on peut définir (sous réserve d'existence) :

Définition : Catégorie des bifaisceaux commodes

On appellera(it) catégorie des bifaisceaux commodes, BFc, la catégorie ainsi définie :

objets = les bifaisceaux X tels que

(i) le faisceau supérieur est dans DAlg

(ii) le faisceau inférieur est dans Foen

(iii) X est un ensemble ordonné complètement distributif, i.e. pour toute famille $\{x_{j,k} : j \in J,\ k \in S(j)\}$ dans X, on a

$$\prod_{j \in J} \Big(\bigsqcup_{k \in S(j)} x_{j,k} \Big) = \bigsqcup_{f \in M} \Big(\prod_{j \in J} x_{j,f(j)} \Big)$$

où M est l'ensemble des fonctions f définies sur J à valeur $f(j) \in S(j)$, au sens suivant : chaque fois que l'une des deux expressions est définie, l'autre l'est aussi et on a l'égalité

(iv) les deux faisceaux sont reliés par la propriété suivante : $\forall x \in X$ si($\forall j \in s_{<<}(x)$ $\{x_{j,k} : k \in S(j)\}$ est un système approximant de j pour le faisceau inférieur) alors $\{ \prod_{j \in J} x_{j,f(j)} : f \in M \}$ est un système approximant de x pour le faisceau inférieur (M défini en (iii)) et $\forall f \in M$ $\{x_{j,f(J)}\}_{j \in J}$ filtrante.

morphismes = les fonctions $f : X \to Y$ qui sont à la fois régulières pour les deux faisceaux, munies de la composition régulière. □

On notera que tout BFc-objet (= objet de BFc) est un bifaisceau commode, et si X, Y $\in$ BFc

$$\hom_{BFc}(X,Y) = \hom_{DAlg}(X,Y) \cap \hom_{Foen}(X,Y)$$
$$= [.\to.]\,(X,Y) \text{ (dans } \underline{\text{Set}})$$

Définissons si $f \in \hom_{BFc}(S_1,S_2)$ et $g \in \hom_{BFc}(T_1,T_2)$,

$$[.\to.]\ (f,g) : [.\to.]\ (S_2,T_1) \to [.\to.]\ (S_1,T_2)$$
$$h \to g\,h\,f$$

4.31 <u>Théorème</u> : $[.\to.]$: <u>BFc</u> x <u>BFc</u> $\to$ <u>B F c</u> est un foncteur.

<u>Démonstration</u> :

(i) les objets : Nous avons vu (4.22) que $\hom_{BFc}(X,Y)$ est un bifaisceau commode si X,Y sont des bifaisceaux commodes. Sa structure dans Foen est obtenue par Inj. Donc $\forall x \in \hom_{BFc}(X,Y)$ $\forall j \in s_{<<}(x)$ $\{x_{j,k} : k \in S(j)\}$ est un Foen-système approximant de $j \in s_{<<}(x) \Leftrightarrow$

$$\{x_{j,k} : k \in S(j)\} = \{j\}$$

Et $\{\prod_{j \in J} x_{j,f(j)} : f \in M\}$ est un Foen-système approximant de x puisque, si $J = s_{<<}(x)$,

$\{\prod_{j\in J} x_{j,f(j)} : f \in M\} = \prod_{j \in J} x_{j,j} = \{\prod_{j\in J} j\} = \{x\}$ et

$\forall f \in M$ $\{x_{j,f(j)} : j \in J\}$ est filtrant.

Pour la complète distribitivité on a :

$\forall$ famille $\{x_{j,k} : j \in J,\ k \in S(j)\}$ dans $\hom_{BFc}(X,Y)$ on a $\forall u \in X$

$(\prod_{j\in J} (\coprod_{k\in S(j)} x_{j,k}))(u) =$ (borne dans <u>DAlg</u>)

$= \prod_{j\in J} (\coprod_{k\in S(j)} x_{j,k})(u) =$ (borne dans <u>Foen</u>)

$= \prod_{j\in J} \coprod_{k\in S(j)} x_{j,k}(u) =$ (distributivité de Y)

$= \coprod_{f\in M} (\prod_{j\in J} x_{j,f(j)}(u)) =$ (en sens inverse)

$= \coprod_{f\in M} \prod_{j\in J} x_{j,f(j)})(u)$ - D'où l'égalité

$$\prod_{j\in J} \coprod_{k\in S(j)} x_{jk} = \coprod_{f\in M} \prod_{j\in J} x_{j,f(j)}$$

Donc $\hom_{BFc}(X,Y)$ peut être muni d'une structure de BFc-objet.

(ii) Il s'agit maintenant d'examiner comment se transforment les flèches

Soit $f : S_1 \to S_2$, $f : T_1 \to T_2$ dans <u>BFc</u>

$$\begin{tikzcd} S_2 \arrow[r, "h"] & T_1 \arrow[d, "g"] \\ S_1 \arrow[u, "f"] & T_2 \end{tikzcd} \qquad [\cdot \to \cdot]\,(f,g) : h \to g\,h\,f$$

$\underline{BFc}$ $\quad$ $\underline{BFc}$

Il faut vérifier que $[\cdot \to \cdot]\,(f,g)$ est un BFc-morphisme. On sait que c'est un DAlg-morphisme d'après les propriétés du foncteur G^* : <u>DAlg</u> x <u>DAlg</u> → <u>DAlg</u>. Le fait que ça soit aussi un BFc-morphisme se réduit à la monotonie croissante par construction du foncteur $[\cdot \to \cdot]$ sur les objets (le faisceau inférieur est fourni par Inj : <u>DAlg</u> → <u>BFc</u>) d'où le théorème . □

4.32 <u>Lemme</u> : <u>BFc</u> a des produits, des coproduits et des équaliseurs . □

On vérifie aisément que dans <u>BFc</u>, le schéma d'approximation de Wadsworth définit un diagramme, dont on peut calculer la limite A^*_∞ en transposant ce que l'on a fait dans C, et on a l'isomorphisme dans <u>BFc</u>

$$A^*_\infty \simeq D + [A^*_\infty \to A^*_\infty]$$

en suivant pas à pas la même démarche.

<u>Une construction simplifiée du modèle A_∞</u>

De même que pour D_∞, on peut simplifier la construction de A_∞ en appauvrissant sa structure de faisceau supérieur au lieu de <u>DAlg</u> on prend <u>DDI</u>. Ceci amène à remplacer les bifaisceaux commodes par les domaines commodes dont on est amené à préciser la définition (dans le cas présent) comme suit

Définition : domaine commode :

Un domaine commode est un bifaisceau X dont le faisceau supérieur est dans DDI, le faisceau inférieur dans Foen et possédant un T.

On remplace l'application

Nolrat : $|\underline{\text{Foen}}| \times |\underline{\text{BFn}}| \to |\underline{\text{DAlg}}|$

en munissant l'espace des fonctions régulières pour le faisceau inférieur $\hom_{\underline{\text{Foen}}}(\text{Inf}(D), \text{Inf}(D))$ du faisceau S pour l'ordre inverse, et en l'injectant dans la classe des domaines commodes en le munissant du f.o.e. trivial (i.e. tous les points sont rationnels), toutes autres choses restant égales par ailleurs. Cette opération qui remplace

$\text{Inj}(\hom_{\underline{\text{Foen}}}(\text{Inf}(D),\text{Inf}(D)), \text{Nolrat})$ sera notée

$\text{Inj}(\hom_{\underline{\text{Foen}}}(\text{Inf}(D), \text{Inf}(D)), \text{Sinv})$

(Sinv pour "faisceau S pour l'ordre inverse")

Reprenant le schéma de Wadsworth, si D est un domaine commode, on a

$A_o = D$

$A_1 = D + [A_o \to A_o]$

où $[A_o \to A_o] = \text{Inj}(\hom_{\underline{\text{Foen}}}(\text{Inf}(D), \text{Inf}(D)), \text{Sinv})$

$A_{n+2} = D + [A_{n+1} \to A_{n+1}] = D + \Delta_{n+1}$ avec

$[A_{n+1} \to A_{n+1}] = [D + \Delta_n \to D + \Delta_n] =$

$[D \to D + \Delta_n] \times [\Delta_n \to D + \Delta_n] =$

$([D \to D] * [D \to \Delta_n]) \times ([\Delta_n \to D] * [\Delta_n \to \Delta_n])$

De la même manière on prendra toujours

$[D \to D] = \text{Inj}(\hom_{\underline{\text{Foen}}}(\text{Inf}(D), \text{Inf}(D)), \text{Sinv})$

$[D \to \Delta_n] =$ Inj(" $\hom_{\underline{\text{DDI}}}(D, \Delta_n)$ muni du faisceau S pour l'ordre inverse") =

$= \text{Inj}(\hom_{\underline{\text{DDI}}}(\text{Sup}(D), \text{Sup}(\Delta_n)), \text{Sinv})$

$[\Delta_n \to D] = \text{Inj}(\hom_{\underline{\text{DDI}}}(\text{Sup}(\Delta_n), \text{Sup}(D)), \text{Sinv})$

$[\Delta_n \to \Delta_n] = \text{Inj}(\hom_{\underline{\text{DDI}}}(\text{Sup}(\Delta_n), \text{Sup}(\Delta_n)), \text{Sinv})$

On se donne les mêmes flèches que dans le cas où le faisceau supérieur est dans <u>DAlg</u> au lieu de <u>DDI</u>. L'étoile joue ici un rôle moins important que dans le cas algébrique : on peut dire qu'elle nous fournit toujours un espace de points qu'on peut munir d'un faisceau S pour l'ordre inverse. Le reste de la méthode s'applique sans problème, et on peut énoncer

4.33 <u>Théorème</u> : Soit D un domaine commode. Alors le schéma d'approximation de Wadsworth fournit dans <u>Set</u> une limite projective A qui est telle que

(i) $A_\infty = D + \varprojlim \Delta_k$, la limite des Δ_k étant calculée dans <u>DDI</u>, ce qui munit A_∞ d'une structure de faisceau

(ii) Si $[A_\infty \to A_\infty]_P$ est l'espace des fonctions régulières partielles de A_∞ dans A_∞, alors il existe une bijection $\varprojlim \Delta_k \leftrightarrow [A_\infty \to A_\infty]_P$. En particulier A_∞ contient l'espace de toutes les fonctions régulières de A_∞ dans A_∞

Ceci donne une justification théorique à la démarche de Nolin dans sa théorie des algorithmes.

BIBLIOGRAPHIE

[ABD] S.K. Abdali : A λ-calculus model of programming languages, computer languages, vol. 1. n°4 (1975), pp. 287-320

[ADJ] J. Goguen, J. Thatcher, E. Wagner, J. Wright : Initial algebra semantics and continuous algebras, JACM 24 (1977), 68-95

J. Goguen, J. Thatcher, E. Wagner, J. Wright : A uniform approach to inductive posets and inductive closure, MFCS 77, LNCS n° 53, Springer (1977), pp. 192-212

[AHO] A. Aho, J. Ullman : Principles of Compiler Design, Addison-Wesley (1978)

[ALG] P. Naur et al. : Revised Report on the Algorithmic Language Algol 60, Computer Journal 5 (1963), pp. 349-367

[ALL] J. Allen : Anatomy of LISP, Mc Graw. Hill, (1978)

[BRU] G. Bruns : Darstellungen und Erweiterungen geordneter Mengen I und II, j. f.d. reine u. angew. Mathematik, vol 209 (1961), pp. 167-220 et vol 210 (1962) pp. 1-23

[BOH] C. Böhm : The CUCH as a formal and description language, "Formal language description languages for Computer Programming", Proc. IFIP Working Conference, North-Holland, (1966) pp. 173-197

[BIR] G. Birkhoff : Lattice theory, AMS Publ., Providence (3ème éd. 1967)

[CHU] A. Church : The calculi of lambda-conversion, Ann. of Math. Studies 6, Princeton, 1941

A. Church : A set of axioms for the foundations of logic, Ann. of Math. Vol 33 (1932) pp. 346-366 et vol 34 (1933) pp. 839-864

[CUR] H.B. Curry, R. Feys : Combinatory Logic, Vol 1. North-Holland (1958)

[COU] B. Courcell, M. Nivat : The algebraic semantics of recursive program schemes, rapport IRIA 300 (1978)

[COU] B. Courcelle, M. Nivat : Algebraic families of interprétations, 17 th FOCS, Houston (1976) pp. 137-146

[DEM] A. Demers, J. Donabue : Report on the Programming Language Russell, Dep. of CS, Cornell University rapport TR 79-37 (1979)

[DON] J. Donahne : One the semantics of "data type" Cornell Univ. rapport TR 77-311 (1977)

[EGL] H. Egli : A mathematical model for non-deterministic computation, ETH Zurich (1975)

[FRE] P. Freyd : Abelian categories : an introduction to the theory of functors, Harper and Row, New York (1964)

[GOG] J. Goguen : Some ideas in algebraic semantics, Proc 3rd IBM Symp. on MFCS Math. Log. and CS, Kansai (Japon), (1978)

[GUE] I. Guessarian : Schémas de programmes récursifs polyadiques : Equivalences sémantiques et classes d'interprétations, Thèse Etat Paris VII, (1975)

[GIN] S. Ginsburg : The mathematical theory of context free languages. Mc Graw Hill, New York 1966

[GUT] J. Guttag : Abstract data types and the development of data structures, CACM 20 : 6 (1977)

[HIN] J. Hindley, B. Lercher, J.P. Seldin : Introduction to combinatory logic, London Math. Soc. Lect. Notes Series n°7, CUP, Cambridge (1972)

[HOF] K.H. Hofmann : Scott's construction from a limit and functorial point of view, 3rd Workshop on Continuous Lattices, Riverside, Cal. (1979)

[HOF2] K.H. Hofmann : Communication personnelle (1979)

K.H. Hofmann : Completely distributive algebraic Lattices, SCS-memo, November 1979

K. Hofmann, M. Mislove, A.R. Stralka :
The Pontryagin duality of 0-dimensional semilattices and its applications, Springer LNM n° 396, 1974

[IAN] I. Ianov : The logical schemes of algorithms, Problems of cybernetics, Pergamon Press (1960), pp. 82-140

[KLE] S. Kleene : Introduction to metamathematics, Van Nostrand (1952)

[KLR] S. Kleene, J. Rosser : The inconsistency of certain formal logics, Ann. of Math. vol. 36 (1935) pp. 630-636

[KUR] K. Kuratowski : Topology, Academic Press (1966)

[LAG] J.L. Lagrange : Suite du supplément à la théorie des fonctions dérivées Continuation du théorème de Taylor et de la manière de terminer les séries infinies. Leçons rédigées par Le Gentil, Ecole Polytechnique Paris (1798)

[LAN] J.P. Landin : A correspondance between Algol 60 and Church's Lambda Notation, CACM 8, pp. 89-101, 158-165 (1965)

[LEV] J.J. Lévy : Réductions correctes et optimales dans le lambda-calcul, Thèse Etat Paris VII, (1978)

[LIS] B. Liskov, A. Snyder, R. Atkinson, C. Scaffert :
Abstraction mechanisms in CLU, CACM 20 : 8 (1977)

[MAN] Y.I. Manin : A course in mathematical logic, Springer, New York (1977)

[MCL] S. Mac Lane : Categories for the working mathematician, Springer (1971)

[MAS] Z. Manna, A. Shamir : The convergence of functions to fixed points of recursive definitions. Stanford AI Lab report (1977)

[MRR] G. Markovsky, B. Rosen : Bases for chain complete posets, IBM J. Res. Dev. 20 (1976), pp. 138-147

[MEN] E. Mendelsohn : Introduction to mathematical logic, Princeton, Van Nostrand (1964)

[MUM] D. Mumford : Introduction to algebraic geometry (Preliminary version of first 3 chapters).

[NAI] M.A. Nait-Abdallah : Remarques sur les fondements de la sémantique axiomatiques, Thèse 3ème cycle Paris VII (1976)

M.A. Naït-Abdallah : Elementary Orders and programming Languages semantics 2nd Workshop on Continuous Lattices, Darmstadt (1978)

M.A. Naït-Abdallah : Ordres élémentaires, 1er Colloque AFCET-SMF, Palaiseau (1978), vol 2 pp. 115-123

M.A. Naït-Abdallah : Types and approximating calculi in programming languages semantics, 3rd Workshop on Continuous Lattices, Riverside, Cal. (1979)

M.A. Naït-Abdallah : Some remarks on Nolin's algorithm theory, Lambda-Conference, Swansea (1979)

[NIV] M. Nivat : Langages algébriques sur le magma libre et sémantique des schémas de programmes, Antomata, Languages and Programming (1972), pp. 293-307

M. Nivat : On the interpretation of recursive polyadic program schemes, Symp. math. 15 (1975) pp. 255-281

M. Nivat : Interprétation universelle d'un schéma de programme récursif, Rivista di Informatica 7 (1977) pp.9-16

M. Nivat : Mots infinis engendrés par une grammaire algébrique, RAIRO Info. théo. 11 (1977) pp. 311-327

M. Nivat, A. Arnold : Calculs infinis, interprétations métriques et plus grands points fixes, 1er Colloque AFCET-SMF, Palaiseau (1978), vol 1 pp. 191-208

M. Nivat, A. Arnold : The metric space of infinite trees, rapport IRIA n°300 (1978)

[NOL] L. Nolin : Systèmes algorithmiques, systèmes fonctionnels, Automata, Languages and Prograamming (1972), pp. 309-318

L. Nolin : Algorithmes universels, RAIRO rouge 2 (1974), pp. 5-18

L. Nolin : Théorie des algorithmes, notes de cours de DEA, Université Paris VII

L. Nolin : Les modèles informatiques du λ-calcul, in λ-calculus and computer science theory, Springer LNCS 19 (1975).

L. Nolin : Pour la théorie des algorithmes, d'après A. Naït Abdallah, in Lamdda-calcul et Sémantique formelle des langages de programmation, LITP (1978) pp. 267-275.

[PAI] M. Painlevé : CRAS pp 1156-1157, Paris (1909).

[PAP] S. Papert : Which distributive lattices are lattices of closed sets ? Proc. Cambr. Phil. Soc. vol. 55 (1959) pp. 172-176.

[PLO] G. Plotkin : A powerdomain construction, SIAM J. Comp., vol.5 (1976), pp. 452-487.

G. Plotkin : T^{ω} as a universal domain, JCSS 17 (1978) pp. 209-236.

G. Plotkin : The category of complete partial orders, a tool for making meanings, Lecture Notes, Pise, 1978.

[PLS] G. Plotkin, M. Smyth : The category-theoretic solution of recursive domain equations, DAI report 60, Edinburgh (1978).

[RAN] G.N. Raney : Completely distributive complete lattices, Proc. AMS vol. 3 (1952), pp. 667-680.

G.N. Raney : A subdirect-union representation for completely distributive complete lattices, Proc. AMS vol. 4 (1953) pp. 518-522.

[SCH] M. Schönfinkel : Uber die Bausteine der mathematischen Logik, Math. Annalen vol. 92 (1924), pp. 305-316.

H. Schubert: Categories, Springer, Berlin (1972).

[SCO] D. Scott : Outline of a mathematical theory of computation, 4th Princeton Conference on Information Sciences (1970).

D. Scott : Continuous lattices, in Toposes. Algebraic geometry and Logic, Springer LNM 274 (1972), pp. 97-136.

D. Scott : Models for various type-free calculi, in Logic, Methodology and Philosophy of Sciences IV (1973), pp. 157-187.

D. Scott : Data types as lattices, SIAM J. Comp. 5 (1976), pp. 522-587.

D. Scott : Some philosophical issues concerning theories of combinators, in λ-calculus and computer science theory, Springer LNCS 19 (1975), pp. 346-366.

D. Scott : Retractes as objects of computation; Lambda-Conference, Swansea (1979).

[SCS] G. Gierz, J.D. Lawson, K.D. Hofmann, M. Mislove, K.Keimel, D.S Scott : A compendium of Continuous Lattices, to appear in Springer LNM (1980).

[SHA] A. Shamir, W. Wadge : Data types as objects, in Springer LNCS 52 (1977) pp. 465-479.

[SMY] M.B. Smyth : Powerdomains, JCSS 16, (1978), pp. 23-26.

[STO] J.E. Stoy : Denotational semantics : the Scott-Stratchey approach to programming language theory, MIT Press (1977).

[SUC] B. Sucher : Contribution à la construction des types d'objets par approximation successive, thèse 3ème cycle Lille (1979).

[VUI] J. Vuillemin : Syntaxe, sémantique et axiomatique d'un langague de programmation simple, Thèse Etat Paris VII (1974)

[WAD] C.P. Wadsworth : Semantics and pragmatics of the λ -calculus PhD. thesis, Oxford (1971).

[WEY] R. Weyhrauch , R. Filman : An FOL Primer, Stanford AI Lab report n° 288, (1976).

R. Weyhrauch : Prolegomena to a theory of mechanized formal reasoning, Stanford AI Lab report n°315 (1978).

[WIE] E. Wiedmer : Exakte Rechnen mit reellen Zahlen und anderen unendlichen Objekten, Diss. ETH Zürich (1977).

[WUL] W. Wulf, R. London, M. Shaw : Abstraction and verification in Alphard : Introduction to language and methodology, report USC/ISI 75-45 (1976).

INDEX et NOTATIONS

NOTATIONS