DIAGRAMMES

PIERRE ARGERON

Sémantique catégorique des types : comprendre le système F

Diagrammes, tome 19 (1988), exp. nº 2, p. PA1-PA40

<http://www.numdam.org/item?id=DIA_1988__19__A2_0>

SEMANTIQUE CATEGORIQUE DES TYPES: COMPRENDRE LE SYSTEME F

Pierre Ageron

ABSTRACT

This paper introduces new ideas in categorical theory of types. It is divided into two parts (two chapters of a forthcoming "thèse de doctorat"). In the first part, we are concerned with the question "what is a type in a category ?". The usual "type = functor" point of view is criticized and replaced by a more general and more expressive framework in the spirit of Ehresmann's sketches, thus providing a unification between algebraic theory of types and logical theory of types. In the second part, we apply these ideas to give a new category-theoretical interpretation of Girard's system F of second order λ-calculus. Among other things, we discuss equational presentations, variances analysis and uniformity.

REMERCIEMENTS

Je remercie mon directeur de thèse J.Y. Girard de bien vouloir accepter ce texte comme témoignage de mon admiration pour ses travaux.

A.M.S. SUB CLASS. : 18 A 15, 03 G 30, 18 D 35, 03 B 15

Je veux exprimer toute ma reconnaissance à C. Lair pour sa disponibilité aux étudiants et l'aide considérable qu'il m'a apportée.

. Enfin, je remercie C. Even et C. Retoré pour leurs encouragements et remarques constructives.

PARTIE I:

QU'EST-CE QU'UN TYPE DANS UNE CATEGORIE?

Ce qui suit est une discussion autour de la notion de type, en particulier de quelques-unes de ses représentations en termes de théorie des catégories.

L'idée de type est, on le sait, un héritage de Bertrand Russell qui avait en vue d'éliminer les paradoxes de la théorie des ensembles.

Il subsiste deux traditions vivantes en matière de types, nettement séparées:
- les systèmes de types issus de la logique,
- les types abstraits de données en informatique.

On peut y ajouter (sans jeu de mots!):
- la théorie des esquisses et des types de structures mathématiques, fondée en 1966 par C. Ehresmann.

Les définitions que nous proposerons ont, entre autres, l'avantage de rendre consubstantiels ces trois points de vue.

Pourquoi choisir le langage de la théorie des catégories pour parler de types? Un argument de départ peut être le suivant:
- la notion de type est intuitivement peu différente de celle d'ensemble, or chaque catégorie est une sorte de variante de la théorie des ensembles, il est alors normal de chercher à faire vivre les types dans certaines catégories convenables, la catégorie $\mathbb{Ens}$ n'est alors plus qu'une catégorie parmi les autres, adaptée ou non au type ou système de types considéré.

Nous suggérons donc de chercher systématiquement à définir des assertions du style:
- la catégorie $\mathbb{C}$ est une sémantique pour le type τ (resp. pour le système de types $\mathcal{T}$),

que l'on pourra aussi formuler ainsi:
- le type τ (resp. le système de types $\mathcal{T}$) est $\mathbb{C}$-iste.

C'est ce que nous ferons dans une deuxième partie pour les types du système F.

Pour l'heure, revenons à la question de fond: qu'est-ce un type dans une catégorie $\mathbb{C}$?

La première idée qui vient à l'esprit, c'est de considérer que:

$$\text{TYPE} = \text{OBJET DE } \mathbb{C} .$$

Ce point de vue impose de se donner arbitrairement un certain nombre de types de base, à partir desquels les autres types sont construits par produits, exponentielles, sommes ...

On retrouve ici les limitations d'expressivité du λ-calcul typé ordinaire de Curry-Church, dans lequel il n'est pas possible de manipuler les types de données courants (booléen, entier, liste ...), à moins de considérer ceux-ci comme des types atomiques et d'introduire pour chacun d'eux de nouveaux schémas non-logiques.

C'est précisément l'intérêt du système F , ou λ-calcul typé du deuxième ordre ([Girard71], [Girard86]), que d'échapper à la fiction de l'atomicité, tout en restant purement logique (c'est la version fonctionnelle du calcul propositionnel intuitionniste du deuxième ordre).

Dans ce système, comme dans d'autres voisins, il n'y a plus de types de base: tous les types y sont construits, conformément à leur "géométrie" intrinsèque. Ceci est rendu possible par la considération de types variables, c'est-à-dire de types où figurent librement des *variables de type*: X , Y ... (nous y reviendrons plus en détail dans la deuxième partie).

Comment représenter catégoriquement un type variable τ , avec n variables de type libres? On est d'abord tenté de répondre:

- par un foncteur $\tau : \mathbb{C}^n \rightarrow \mathbb{C}$.

Mais on se rend vite compte qu'imposer la covariance par rapport aux arguments est une condition trop forte (sauf dans le cas de certaines catégories "hémiplégiques": notamment des catégories de plongements - comme dans [Girard86] ou aussi [Coquand88]). Par exemple, il faut pouvoir séparer occurrences négatives et positives d'une même variable par rapport au constructeur $\Rightarrow$.

Une solution consiste alors à dédoubler chaque variable X en X^g et X^d , donc à opter pour un "bifoncteur":

$$\tau : (\mathbf{C}^n)^{op} \times \mathbf{C}^n \to \mathbf{C} \,.$$

Ce point de vue, qui est celui de [Freyd-Girard-Scedrov-Scott88], peut paraître assez artificiel ("schizophrène").
De plus, il est un obstacle à une représentation équationnelle du type T , car l'endofoncteur $(-)^{op}$ de $\mathbf{Cat}$ n'est pas co-représentable (autrement dit: $\mathbf{C}^{op}$ n'est pas " $\mathbf{C}$ à la puissance op " !). Or on sait l'intérêt que les informaticiens (en particulier) portent aux représentations équationnelles des concepts catégoriques ([Curien86], [Coquand-Ehrhard87]).

Autre critique: une telle représentation ne prend pas en compte la structure interne des types, ce qui est un péché contre l'esprit même du système F.

Si l'on veut être plus précis, il faut cesser d'assimiler un type à un seul objet de $\mathbf{C}$ et le voir plutôt comme un *diagramme* dans $\mathbf{C}$, figurant les fonctions de construction, les fonctions d'accès ...

Que faut-il entendre par diagramme ? Au sens strict, un diagramme dans $\mathbf{C}$ est un foncteur de $\mathcal{J}$ dans $\mathbf{C}$, où $\mathcal{J}$ est une petite catégorie (ou encore, ce qui est à peine plus général, un petit compographe - voir la définition plus loin).
Mais il faut à $\mathcal{J}$ une structure supplémentaire, permettant de spécifier que certains des objets et/ou des flèches de $\mathbf{C}$, en lesquels il se réalise, obéissent à telle ou telle propriété universelle.

Ainsi, si l'on n'a à spécifier que des produits, la structure de signature est suffisante (cas des types abstraits de données en informatique). La notion d'esquisse ([Ehresmann68]) permet plus généralement de spécifier des cônes limite projective ou limite inductive. Mais pour spécifier aussi des propriétés universelles plus complexes, car "à plusieurs niveaux" (comme celles définissant une exponentielle ou un objet des entiers naturels), il nous faut recourir à la notion de trame ([Lair87]).

1. Trames, homomorphismes et modèles.

Compographes, foncteurs, transformations naturelles.

Un *compographe* $\mathbb{G}$ est un graphe (orienté), dans lequel (seulement) certains couples de flèches consécutives ont un composé défini. On suppose de plus que l'on dispose pour tout objet G d'une flèche $Id_G : G \rightarrow G$, de sorte que pour toute flèche $g : G \rightarrow G'$, les composés $g.Id_G$ et $Id_{G'}.g$ soient définis et égaux à g. Les graphes, les catégories sont des cas particuliers de compographes. On définit sans difficulté la notion de *foncteur* entre deux compographes $\mathbb{G}$ et $\mathbb{G}'$, ainsi que celle de *transformation naturelle* entre deux foncteurs $F_1, F_2 : \mathbb{G} \rightarrow \mathbb{C}$, si $\mathbb{C}$ est une catégorie.

Trames et homomorphismes de trames (par récurrence sur $n \in \mathbb{N}$).

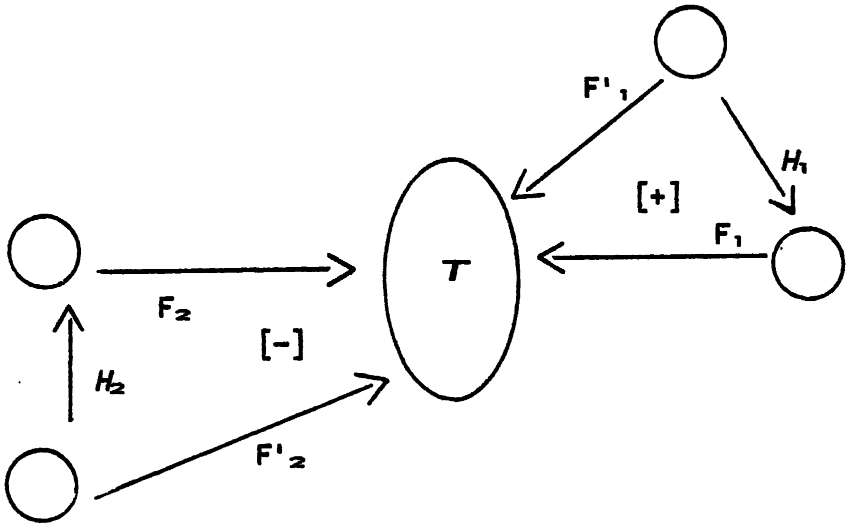
a) Une *trame d'ordre* 0 est un compographe et un *homomorphisme* de trames d'ordre 0 est un foncteur entre compographes.

b) $\mathcal{T} = (\mathbb{T}, \mathcal{H}^-(\mathcal{T}), \mathcal{H}^+(\mathcal{T}))$
est une *trame d'ordre* $\leqslant$ n+1 si, et seulement si:
- $\mathbb{T}$ est un compographe,
- $\mathcal{H}^-(\mathcal{T})$ et $\mathcal{H}^+(\mathbb{T})$ sont des ensembles de triplets $(F', \mathcal{H}, F)$ où
  + $\mathcal{H} : \mathcal{T}'_{\mathcal{H}} \rightarrow \mathcal{T}_{\mathcal{H}}$ est un homomorphisme de trames d'ordre $\leqslant n$,
  + $F' : \mathbb{T}'_{\mathcal{H}} \rightarrow \mathbb{T}$ est un foncteur,
  + $F : \mathbb{T}_{\mathcal{H}} \rightarrow \mathbb{T}$ est un foncteur,
  + $F \circ \mathcal{H} = F'$.

c) $\mathcal{H} = (\mathcal{T}', H, \mathcal{T})$ est un *homomorphisme de trames d'ordre* $\leqslant$ n+1 si, et seulement si:
- $\mathcal{T}'$, $\mathcal{T}$ sont des trames d'ordre $\leqslant$ n+1 ,
- $H : \mathbb{T}' \rightarrow \mathbb{T}$ est un foncteur tel que:
  + $(F'_2, \mathcal{H}_2, F_2) \in \mathcal{H}^-(\mathcal{T}') \Rightarrow (HoF'_2, \mathcal{H}_2, HoF_2) \in \mathcal{H}^-(\mathcal{T})$,
  + $(F'_1, \mathcal{H}_1, F_1) \in \mathcal{H}^+(\mathcal{T}') \Rightarrow (HoF'_1, \mathcal{H}_1, HoF_1) \in \mathcal{H}^+(\mathcal{T})$.

Remarquons que:
- il faut faire attention à la façon de lire cette définition, par exemple, pour avoir la définition d'une trame d'ordre $\leqslant$ 3 , il faut parcourir les alinéas dans l'ordre suivant:

a),b),c),b),c),b),
- si $\mathcal{T}$ est d'ordre 0, on convient que $\mathcal{H}^-(\mathcal{T}) = \mathcal{H}^+(\mathcal{T}) = \emptyset$,
- en remplaçant dans cette définition le mot "compographe" par le mot "graphe", on obtient une structure plus particulière que celle de trame que nous appellerons *graphotrame,*
- si s est un entier naturel, le graphe discret à s objets sera (encore) noté s et la catégorie associée à la relation d'ordre $\leqslant$ sur { 0, 1 ..., s } sera notée $\langle s \rangle$; la trame d'ordre 0 de support s (resp. $\langle s \rangle$) sera aussi notée s (resp $\langle s \rangle$) ,
- on peut représenter une trame par le schéma suivant (remarquer les signes - et +):



Catégorie des modèles d'une trame $\mathcal{T}$ dans une catégorie $\mathbb{C}$.

On note $\mathbb{C}^{\mathcal{T}}$ la sous-catégorie pleine de la catégorie des foncteurs de $\mathcal{T}$ vers $\mathbb{C}$ dont les objets M , appelés *modèles* de $\mathcal{T}$ dans $\mathbb{C}$, sont définis (par récurrence sur l'ordre de $\mathcal{T}$) comme suit:
- $\forall$ (F', *H*, F) $\in \mathcal{H}^-(\mathcal{T})$, MoF' est un modèle de $\mathcal{T}'_H$, MoF est un modèle de $\mathcal{T}_H$, et pour tout modèle N de $\mathcal{T}_H$ dans $\mathbb{C}$ vérifiant NoH = MoF' , il existe dans $\mathbb{C}^{\mathcal{T}_H}$ une unique flèche δ de N vers MoF qui vérifie δoH = $Id_{MoF'}$. (on dit alors que MoF' est un *prolongement co-universel* de MoF le long de *H*),
- $\forall$ (F', *H*, F) $\in \mathcal{H}^+(\mathcal{T})$, MoF' est un modèle de $\mathcal{T}'_H$, MoF est un modèle de $\mathcal{T}_H$, et pour tout modèle N de $\mathcal{T}_H$ dans $\mathbb{C}$ vérifiant NoH = MoF' , il existe dans $\mathbb{C}^{\mathcal{T}_H}$ une unique flèche δ de MoF vers N qui vérifie δoH = $Id_{MoF'}$. (on dit alors que MoF' est un *prolongement universel* de MoF le long de *H*) .

On peut également définir des modèles *faibles*, en supprimant le mot "unique" de cette définition; autrement dit en ne demandant que des *prolongements quasi-co-universels*.

Nous noterons $\mathrm{Mod}(\mathcal{T}, \mathbf{C})$ la classe des objets de $\mathbf{C}^{\mathcal{T}}$ (attention: la notation n'est pas celle de [Lair87]).

En notant $\mathbf{Tram}$ la catégorie des trames, on a donc un bimodule de $\mathbf{Tram}$ à $\mathbf{Cat}$, i.e un foncteur

$$\mathrm{Mod} : \mathbf{Tram}^{op} \times \mathbf{Cat} \to \mathbf{Ens}$$

2. Syntaxes de types et types.

Préliminaires.

Revenons à la présentation des types. Toute la discussion qui a précédé nous amène à voir un type variable comme une application - et non plus un foncteur - telle que:

$$\tau : \mathrm{Mod}(n, \mathbf{C}) \to \mathrm{Mod}(\mathcal{J}, \mathbf{C})$$

où $\mathcal{J}$ est une (petite) trame figurant la "géométrie" du type. Bien entendu, il doit être précisé comment n s'envoie dans $\mathcal{J}$.

On peut facilement aller un peu plus loin et accepter des types variables qui contiendraient des hypothèses sur leurs arguments (non-vacuité, inclusions, ...).
Pour de tels types, il convient d'admettre des arités autres que des entiers naturels, c'est-à-dire de considérer plus généralement des applications

$$\tau : \mathrm{Mod}(\mathcal{I}, \mathbf{C}) \to \mathrm{Mod}(\mathcal{J}, \mathbf{C})$$

associées d'une façon plus ou moins naturelle à un homomorphisme de trames $H : \mathcal{I} \to \mathcal{J}$.

Il y a au moins deux (familles de) telles applications qui sont naturelles (lorsqu'elles sont définies); ce sont les applications "choix de prolongement co-universel (resp. choix de prolongement universel) le long de H ". Elles conduisent à deux notions de types (sans préjuger la possibilité d'en imaginer d'autres).

Syntaxes de types.

Une *syntaxe de types* est un homomorphisme de trames $H: \mathcal{I} \to \mathcal{J}$.

Types inductifs et types projectifs.

Un *type inductif* (resp. *projectif*) dans $\mathbb{C}$ de syntaxe H est une application

$$\tau : \mathrm{Mod}(\mathcal{I}, \mathbb{C}) \to \mathrm{Mod}(\mathcal{J}, \mathbb{C})$$

telle que:
- $\mathrm{Mod}(H, \mathbb{C}) \circ \tau = \mathrm{Id}_{\mathrm{Mod}(\mathcal{I}, \mathbb{C})}$,
- $\forall$ N $\in \mathrm{Mod}(\mathcal{I}, \mathbb{C})$, $\forall$ M $\in \mathrm{Mod}(\mathcal{J}, \mathbb{C})$, on a:

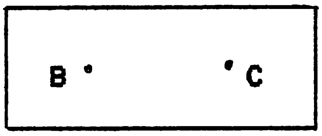
$$\mathrm{M} \circ H = \mathrm{N} \Rightarrow \exists!\, \delta : \tau(\mathrm{N}) \to \mathrm{M} \ / \ \delta \circ H = \mathrm{Id}_{\mathrm{N}} ,$$

(resp. :
- $\forall$ N $\in \mathrm{Mod}(\mathcal{I}, \mathbb{C})$, $\forall$ M $\in \mathrm{Mod}(\mathcal{J}, \mathbb{C})$, on a:

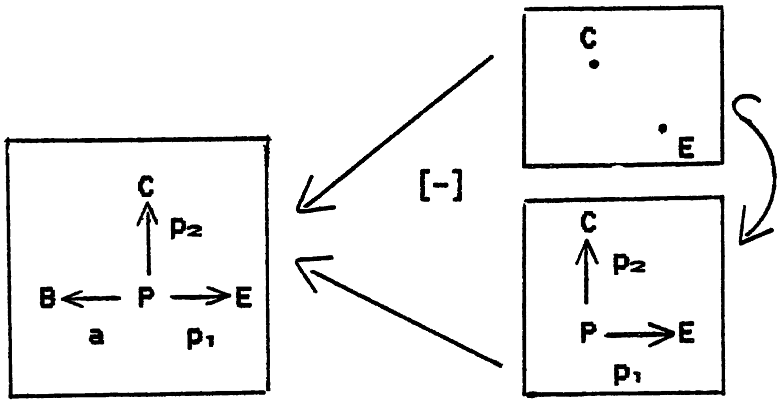
$$\mathrm{M} \circ H = \mathrm{N} \Rightarrow \exists!\, \delta : \mathrm{M} \to \tau(\mathrm{N}) \ / \ \delta \circ H = \mathrm{Id}_{\mathrm{N}} \quad).$$

Classiquement, on voit par exemple que:
- si $\mathbb{I}$ est un compographe, si $\mathbb{J}$ est le compographe $\mathbb{I}^{+}$ obtenu en ajoutant à $\mathbb{I}$ un (nouvel) objet terminal et si H est le foncteur d'inclusion de $\mathbb{I}$ dans $\mathbb{J}$, alors un type inductif dans $\mathbb{C}$ de syntaxe H est un *choix de limites inductives pour les foncteurs de* $\mathbb{I}$ *dans* $\mathbb{C}$,
- si $\mathbb{I}$ est un compographe, si $\mathbb{J}$ est le compographe $\mathbb{I}^{-}$ obtenu en ajoutant à $\mathbb{I}$ un (nouvel) objet initial et si H est le foncteur d'inclusion de $\mathbb{I}$ dans $\mathbb{J}$, alors un type projectif dans $\mathbb{C}$ de syntaxe H est un *choix de limites*

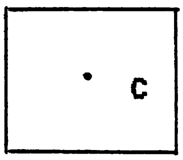
projectives pour les foncteurs de $\mathbb{I}$ *dans* $\mathbb{C}$ *,*
- si $\mathbb{I}$ est la trame d'ordre 0 suivante:



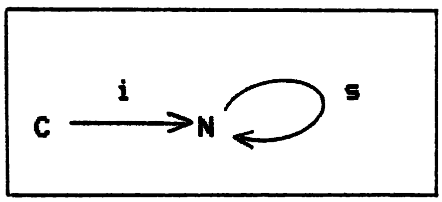
si $\mathcal{J}$ est la trame d'ordre 1 suivante:



et si *H* est l'homomorphisme canonique de $\mathbb{I}$ dans $\mathcal{J}$, alors un type projectif dans $\mathbb{C}$ de syntaxe *H* est un *choix de diagrammes "exponentielle" pour les couples d'objets de* $\mathbb{C}$ *,*
- si $\mathbb{I}$ est la trame d'ordre 0 suivante:



si $\mathcal{J}$ est la trame d'ordre 0 suivante:



et si H est l'homomorphisme canonique de $\mathcal{I}$ dans $\mathcal{J}$, alors un type inductif de syntaxe H est une *structure de catégorie de Peano sur* $\mathbb{C}$ ([Coppey84]),
- si $\mathcal{I}$ est vide, un type inductif (resp. projectif) dans $\mathbb{C}$ de syntaxe *H* (de source $\mathcal{I}$ et de but $\mathcal{J}$) s'identifie à un modèle initial (resp. terminal) de $\mathcal{J}$ dans $\mathbb{C}$; un tel type peut être dit *clos* (exemples: un objet initial, un objet terminal, un objet des entiers naturels dans $\mathbb{C}$).

Catégorie des sémantiques d'un type.

Soit $H : \mathcal{I} \to \mathcal{J}$ une syntaxe de types. La catégorie $\mathbf{S\acute{e}m}^{+}(H)$ (resp. $\mathbf{S\acute{e}m}^{-}(H)$) des sémantiques catégoriques du type inductif (resp. projectif) de syntaxe *H* est définie comme suit:
- ses objets sont les couples $(\mathbb{C}, \tau)$ tels que:
  + $\mathbb{C}$ est une petite catégorie,
  + τ est un type inductif (resp. projectif) dans $\mathbb{C}$ de syntaxe *H*,
- les flèches de $(\mathbb{C}_1, \tau_1)$ vers $(\mathbb{C}_2, \tau_2)$ sont les foncteurs $F: \mathbb{C}_1 \to \mathbb{C}_2$ vérifiant:

$$\mathrm{Mod}(\mathcal{J}, F) \circ \tau_1 = \tau_2 \circ \mathrm{Mod}(\mathcal{I}, F).$$

Si $\Sigma = (\Sigma^{-}, \Sigma^{+})$ est un couple de classes de syntaxes de types, on définit de façon analogue une catégorie $\mathbf{Sem}(\Sigma)$.

Par exemple, la catégorie $\mathbf{Cartf}_{N}$ des petites catégories cartésiennes fermées avec objet des entiers naturels apparaît comme étant $\mathbf{Sem}(\{H_1, H_2, H_3\}, \{H_4\})$, où :
- H_1 spécifie un objet terminal,
- H_2 spécifie des cônes produit,
- H_3 spécifie des diagrammes "exponentielle",
- H_4 spécifie un objet des entiers naturels,

(bien entendu, H_4 spécifie en particulier un objet terminal; on pourrait donc penser faire l'économie de H_1 ; ce n'est pas une bonne idée, car on perdrait une propriété remarquable du système Σ ; son caractère *héréditaire* - voir [Lair87]).

Mentionnons pour finir quelques résultats sur la structure de "sémantique catégorique d'un système de syntaxes de types Σ " comparée à celle de catégorie et à celle de graphe.

Les questions (qui sont évidemment liées à certains problèmes d'équationnalité) se posent ainsi:
1.a. le foncteur d'oubli canonique de $\mathbf{Sem}(\Sigma)$ dans $\mathbf{Cat}$ admet-il un adjoint à gauche ?
1.b. ce foncteur est-il, de plus, monadique (autrement dit: $\mathbf{Sem}(\Sigma)$ est-elle une catégorie doctrinale)?
2.a. le foncteur d'oubli canonique de $\mathbf{Sem}(\Sigma)$ dans $\mathbf{Graph}$ admet-il un adjoint à gauche ?
2.b. ce foncteur est-il, de plus, monadique (autrement dit: $\mathbf{Sem}(\Sigma)$ est-elle une catégorie graphique)?
Les réponses aux questions 1.a et 2.a sont oui (pourvu que Σ soit héréditaire): toute catégorie petite (resp. tout graphe petit) engendre librement une sémantique catégorique pour Σ ([Lair87]) .
En revanche, les réponses aux questions 1.b et 2.b dépendent de Σ . Voici quelques résultats connus:
- la catégorie des catégories à choix de $\mathcal{X}$-limites inductives et $\mathcal{Y}$-limites projectives (où $\mathcal{X}$ et $\mathcal{Y}$ sont des classes de compographes) est doctrinale ([Lair75]),
- la catégorie des catégories à choix de produits (resp. sommes, noyaux, conoyaux, produits fibrés, sommes amalgamées) est doctrinale et graphique ([Burroni81], [Coppey-Lair85]),
- la catégorie des catégories de Peano est doctrinale, mais non graphique ([Coppey-Lair85]),
- la catégorie des catégories cartésiennes fermées est doctrinale et graphique ([MacDonald-Stone84] , [Ageron-Even88]).
Ce genre de questions peut être abordé systématiquement par la théorie des esquisses, qui fournit des critères généraux et commodes pour y répondre ([Lair75] , [Lair79]) .

PARTIE II:

SEMANTIQUE GEOMETRIQUE DU SYSTEME F

On sait (voir par exemple le chapitre 1 de [Lambek-Scott86]) que la structure de *catégorie cartésienne fermée* permet d'interpréter les types et les termes du λ-calcul typé, ou encore, via l'isomorphisme de Curry-Howard, les propositions et les preuves de la logique intuitionniste.

En 1970, Jean-Yves Girard introduisait une extension du λ-calcul typé, le système F ([Girard71]): il s'agit de la version fonctionnelle de la logique (propositionnelle) intuitionniste du deuxième ordre. La question se pose alors de déterminer un cadre mathématique général, du type "catégorie + une certaine structure algébrique", permettant de donner un sens (formel) aux types et aux termes du système F.

Cette interprétation devra préserver les idées sémantiques naîves suggérées par la logique (type = ensemble de preuves) ou par l'informatique (type = ensemble d'algorithmes); elle devra aussi être fidèle au principe qui pour nous est l'esprit même du système F: *présenter chaque type comme construit conformément à sa géométrie intrinsèque*. (Notons que le formalisme à base d'hyperdoctrines, introduit dans [Seely87] et repris dans [Pitts87] et [Coquand-Ehrhard87], ne répond pas à ces préoccupations; de plus, les types y sont des foncteurs - internes - , ce que nous avons critiqué dans la discussion précédente.)

1. Types: taxinomie.

Syntaxe des types du système F.

Voici la syntaxe des types du système F, dans sa version la plus récente et la plus dépouillée ([Girard86] , [Girard88]).
On se donne un ensemble infini $W = \{ X , Y , \ldots \}$ appelé ensemble

des *variables de type*. On définit (inductivement) comme suit l'ensemble F-Typ(W) des *types* du système F en les variables de W , ainsi que pour tout type A l'ensemble W(A) des *variables de type libres dans* A :
- si $X \in W$, alors $X \in$ F-Typ(W) et $W(X) = \{X\}$,
- si $A_1, A_2 \in$ F-Typ(W), alors $A_1 \Rightarrow A_2 \in$ F-Typ(W) et $W(A_1 \Rightarrow A_2) = W(A_1) \cup W(A_2)$,
- si $X \in W$ et si $A \in$ F-Typ(W), alors $\forall X.A \in$ F-Typ(W) et $W(\forall X.A) = W(A)\backslash\{X\}$.

Cette syntaxe est extraordinairement simple: hormis les variables (et les parenthèses), elle admet exactement deux symboles $\Rightarrow$ et $\forall$. Nous convenons que $\Rightarrow$ est prioritaire sur $\forall$.

Taxinomie.

La syntaxe précédente induit la *taxinomie* suivante:
- nous appellerons *$\Rightarrow$-types* (resp. *$\forall$-types*) les types de la forme $A_1 \Rightarrow A_2$ (resp. $\forall X.A$).

On peut qualifier cette taxinomie de *superficielle*, en ce sens qu'elle ne tient compte que du dernier symbole introduit - encore celui-ci n'a-t-il pas nécessairement de signification profonde: sémantiquement, nous pourrons identifier tout type A au $\Rightarrow$-type $(\forall X.X\Rightarrow X) \Rightarrow A$ ainsi qu'au $\forall$-type $\forall X.(A \Rightarrow X) \Rightarrow X$ (si X n'appartient pas à W(A)).

La thèse générale qui préside à notre interprétation peut se résumer dans les trois points suivants:
- le symbole $\Rightarrow$ permet certaines constructions de types de nature projective (stuctures co-libres),
- le symbole $\forall$ permet certaines constructions de types de nature inductive (stuctures libres),
- c'est justement la possibilité d'entremêler ces symboles, d'alterner constructions projectives et constructions inductives, qui donne au système F son énorme pouvoir expressif en matière de types.

On mesure facilement, par exemple, la supériorité de F sur le langage de programmation ML: en effet, celui-ci n'accepte que les quantifications prénexes (toute occurence de $\forall$ doit précéder toute occurence de $\Rightarrow$), ce qui empêche notamment de typer convenablement l'addition entre entiers de Church ([Huet88]).

Par ailleurs, la possibilité d'emboîter, à la façon des poupées russes, constructions projectives et constructions inductives est tout à fait caractéristique des notions de type dans une catégorie, développées dans la première partie de cet article.

Nous allons donc montrer ici que les types du système F peuvent se traduire dans ce formalisme catégorique, plus puissant et plus symétrique, tout en préservant les idées sémantiques naïves transmises par la tradition orale. Tous les types de F prendront alors, nous l'espérons, une signification, une pure écriture *signifiant* à l'esprit la formation d'une image géométrique.

2. Types: géométrie.

Aperçu sur la géométrie des types.

Soit A un type du système F, avec $W(A) = \{X_1, \ldots, X_r\}$. Nous appellerons *géométrie* de A un certain diagramme de trames et d'homomorphismes de trames de la forme:

$$\mathcal{W}(A) \xrightarrow{H(A)} \mathcal{G}(A) \xleftarrow{\Psi(A)} 1$$

où $\mathcal{W}(A)$ est la trame discrète d'ordre 0 dont les objets sont $X_1, \ldots, X_r$.

Le rôle des homomorphismes $H(A)$ et $\Psi(A)$ est de *pointer* à l'intérieur de la trame $\mathcal{G}(A)$ les emplacements des variables libres $X_1, \ldots, X_r$ et celui du type A, "déshabillé" de toute structure. On notera abusivement $\Psi(A)$ l'objet $\Psi(A)(0)$.

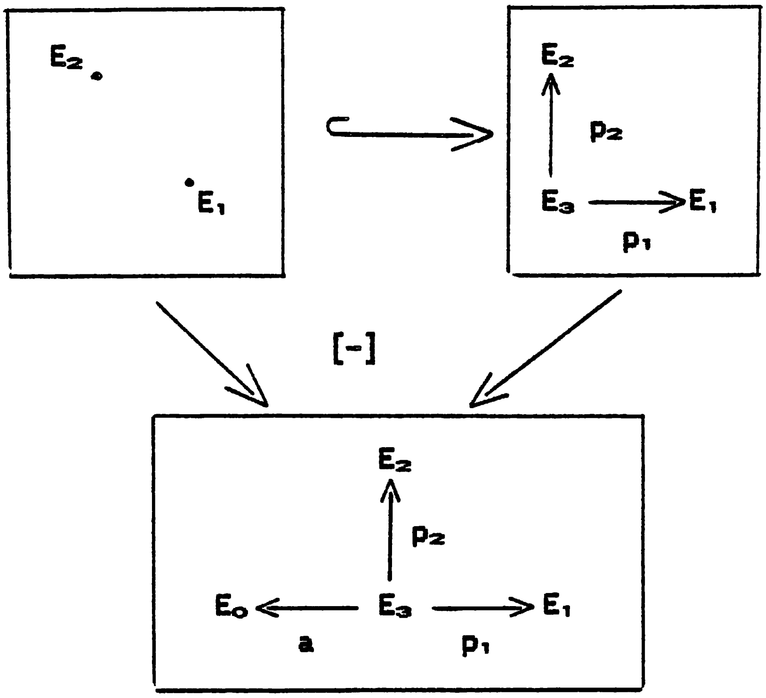
Définition de la géométrie des types.

Voici la définition précise de $\mathbb{G}(A)$, $H(A)$ et $\Psi(A)$, par induction sur l'écriture de A.

- 1er cas: $A = X$ est une variable de type. Dans ce cas, on pose:

$$\mathbb{G}(A) = \omega(A),$$
$$H(A) = Id_{\omega(A)},$$
$$\Psi(A) = X.$$

- 2ème cas: $A = A_1 \Rightarrow A_2$ est un $\Rightarrow$-type. Dans ce cas, soit $\mathbb{E}$ la graphotrame d'ordre 1 suivante:



Alors, on pose:

$$\mathbb{G}(A) = \mathbb{G}(A_1) + \mathbb{E} + \mathbb{G}(A_2)$$

où les symboles "+" indiquent que l'on fait une somme amalgamée dans laquelle:
- on identifie les objets $\Psi(A_1)$ et E_2 ainsi que les objets $\Psi(A_2)$ et E_0,
- pour tout $Y \in W(A_1) \cap W(A_2)$, on identifie les objets

$H(A_1)(Y)$ et $H(A_2)(Y)$.
Si l'on note I_1, I_0, I_2 les trois inductions de $\mathbb{G}(A_1)$, $\mathbb{E}$, $\mathbb{G}(A_2)$ dans $\mathbb{G}(A)$, alors $H(A) : \mathbb{W}(A) \to \mathbb{G}(A)$ est défini sans ambiguïté par:

$$H(A)(Y) = I_j(H(A_j)(Y)) \text{ , pour tout } Y \in W(A_j) \text{ .}$$

Enfin, on pose:

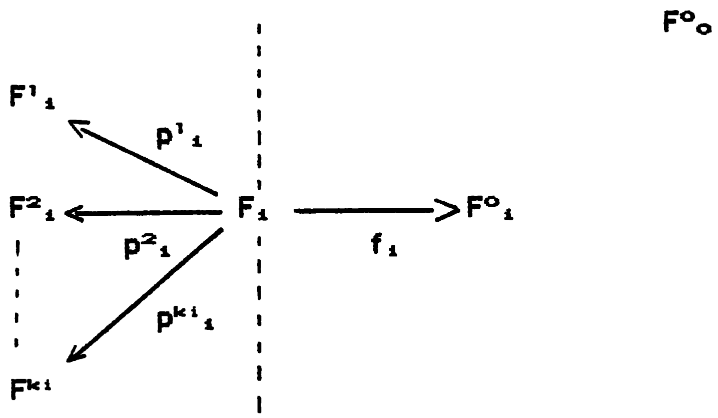
$$\Psi(A) = I_0(E_1).$$

- $3^{\text{ème}}$ cas: A est un $\forall$-type. Alors A s'écrit de manière unique sous la forme:

$$A = \forall X.A_1 \Rightarrow (A_2 \Rightarrow \ldots \Rightarrow (A_n \Rightarrow A^o{}_0) \ldots), \qquad n \in \mathbb{N}$$

$$A_i = A^1{}_i \Rightarrow (A^2{}_i \Rightarrow \ldots \Rightarrow (A^{k_i}{}_i \Rightarrow A^o{}_i) \ldots) \qquad k_i \in \mathbb{N}.$$

Pour $n \in \mathbb{N}$ et $(k_1, \ldots ,k_n) \in \mathbb{N}^n$, notons $\mathbb{F}(k_1, \ldots ,k_n)$ la graphotrame d'ordre 1 suivante:



Soit $K = \{(i,j) \ / \ 1 \leq i \leq n \text{ et } 0 \leq j \leq k_i\} \cup \{(0,0)\} \subset \mathbb{N}^2$. On pose:

$$\mathbb{G}(A) = \mathbb{F}(k_1, \ldots k_n) + \Sigma_{(i,j) \in K} \mathbb{G}(A^j{}_i),$$

où le symbole "+" indique que l'on effectue une somme amalgamée dans laquelle:
- pour tout $(i,j) \in K$, on identifie les objets $\Psi(A^j{}_i)$ et $F^j{}_i$,

- à chaque fois qu'il existe $(i,j) \in K$, $(i',j') \in K$ et $Y \in W(A^j{}_i) \cap W(A^{j'}{}_{i'})$, on identifie les objets $H(A^j{}_i)(Y)$ et $H(A^{j'}{}_{i'})(Y)$.

Pour $(i,j) \in K$, notons donc $I^j{}_i$ l'*induction* de $\mathbb{G}(A^j{}_i)$ dans $\mathbb{G}(A)$. Alors $H(A) : \mathbb{W}(A) \to \mathbb{G}(A)$ est défini sans ambiguïté par:

$$H(A)(Y) = I^j{}_i(H(A^j{}_i)(Y)), \text{ pour tout } Y \in W(A^j{}_i) .$$

Enfin, on pose:

$$\mathbb{K}(A) = I^o{}_o(\mathbb{K}(A^o{}_o)) .$$

Remarques.

Pour conclure, notons que:
- les trames de la forme $\mathbb{G}(A)$ sont des graphotrames,
- les (grapho)trames $\mathbb{E}$ et $\mathbb{F}(k_1, \ldots k_n)$ sont elles-mêmes de la forme $\mathbb{G}(A)$: ceci révèle bien que le système F est fondé sur le choix de certains (constructeurs de) types, choix qui, finalement, est à peine moins arbitraire que celui de types atomiques clos dans le λ-calcul typé du premier ordre, mais comment y échapper tant qu'une syntaxe de types reste une écriture ?

3. Types: sémantique.

Préliminaires.

Conformément au programme présenté dans la première partie de cet article, nous allons maintenant chercher à définir l'assertion:

la catégorie $\mathbb{C}$ est une sémantique pour le type A.

En fait, nous nous intéresserons au cas où A est un $\forall$-type. Bien entendu, le cas des $\Rightarrow$-types pourrait être traité de façon rigoureusement symétrique, mais il est beaucoup plus

banal, car il se ramène simplement à demander l'existence de certaines exponentielles.
Nous supposerons donc toujours, a priori, que $\mathbb{C}$ est une catégorie cartésienne fermée, ce qui permettra de mettre l'accent sur le constructeur le plus original: $\forall$.

Soient donc A un $\forall$-type du système F et $\mathbb{C}$ une catégorie cartésienne fermée. On dispose des deux ensembles suivants:
- $\mathrm{Mod}(\mathbb{G}(A), \mathbb{C})$ = ensemble des modèles de $\mathbb{G}(A)$ dans $\mathbb{C}$,
- $\mathrm{Mod}(\mathbb{W}(A), \mathbb{C})$ = ensemble des assignations des variables libres de A dans $\mathbb{C}$,

ainsi que de l'application "assignation sous-jacente":

$$\begin{array}{rcl} \mathrm{Mod}(H(A), \mathbb{C}) : \mathrm{Mod}(\mathbb{G}(A), \mathbb{C}) & \to & \mathrm{Mod}(\mathbb{W}(A), \mathbb{C}) . \\ M & \mapsto & M \circ H(A) \end{array}$$

Remarquons que cette application est l'induction aux objets du foncteur d'oubli

$$U(A) = \mathbb{C}^{H(A)} : \mathbb{C}^{\mathbb{G}(A)} \to \mathbb{C}^{\mathbb{W}(A)} .$$

Une idée naturelle consisterait à dire que $\mathbb{C}$ est une sémantique pour le type A si, et seulement si, $U(A)$ admet un adjoint à gauche $L(A)$. Mais cette condition est beaucoup trop forte: elle impose en effet que A soit "covariant" par rapport à toutes ses variables. C'est pourquoi il faut se retourner vers des définitions ensemblistes.

Types schématisables et sémantisables.

On dit que A est un type *schématisable* dans $\mathbb{C}$ si, et seulement si:
- l'application $\mathrm{Mod}(H(A), \mathbb{C})$ est surjective,

ceci est équivalent à dire qu'il existe (au moins) une application:

$$\mathrm{choix}(A) : \mathrm{Mod}(\mathbb{W}(A), \mathbb{C}) \to \mathrm{Mod}(\mathbb{G}(A), \mathbb{C}),$$

vérifiant la condition (a) suivante:

a) $$\mathrm{Mod}(H(A),\ \mathbb{C}) \circ \mathrm{choix}(A) = \mathrm{Id}_{\mathrm{Mod}(\mathbb{W}(A),\mathbb{C})}$$

On dit que A est ***sémantisable*** dans $\mathbb{C}$ si, et seulement si:
- on peut choisir cette application de sorte que soient vérifiées la condition (a) précédente et la condition (b) suivante:

b) $$\forall\ N \in \mathrm{Mod}(\mathbb{W}(A),\ \mathbb{C})\ ,\quad \forall\ M \in \mathrm{Mod}(\mathbb{G}(A),\ \mathbb{C}),$$
$$M\mathrm{o}H(A) = N \ \Rightarrow\ \exists!\ \delta : \mathrm{choix}(A)(N) \xrightarrow[\mathbb{C}^{\mathbb{G}(A)}]{} M,$$

telle que $\delta\mathrm{o}H(A) = \mathrm{Id}_N$.

Avec la terminologie de la première partie , il est clair qu'un $\forall$-type A est sémantisable dans $\mathbb{C}$ si, et seulement si, $\mathbb{C}$ est sous-jacente à une sémantique catégorique du type inductif de syntaxe $H(A)$; on dit alors que celle-ci est une *sémantique catégorique* de A et l'on pose

$$\mathbf{S\acute{e}m}(A) = \mathbf{S\acute{e}m}^{+}(H(A))\ .$$

Par abus de langage, on dira aussi que $\mathbb{C}$ est une *sémantique catégorique pour le type* A si $\mathbb{C}$ est la catégorie sous-jacente à un objet de $\mathbf{S\acute{e}m}(A)$; mais cet abus peut être dangereux ! En effet, les deux notions (avec ou sans choix) sont structurellement très différentes (et sont les objets de catégories qui ne sont même pas équivalentes).
Enfin, on laisse le lecteur préciser lui-même ce qu'il faut entendre par sémantique catégorique pour un $\Rightarrow$-type.

Exemple.

La classe d'exemples la plus importante est celle des types "naturels" qui seront étudiés en détail dans la section suivante. Donnons seulement ici un exemple très différent, inspiré de [Girard86].

Considérons le type (clos):

$$A_0 = \forall X.(X \Rightarrow (X \Rightarrow X)) \Rightarrow (((X \Rightarrow X) \Rightarrow X) \Rightarrow X).$$

La géométrie de ce type est:

$$\mathbb{G}(\Lambda_o) = \begin{tikzcd} X \times X \arrow[r, "r"] \arrow[r, bend left, "p_1"] \arrow[r, bend right, "p_2"'] & X & X^X \arrow[l, "s"'] \\ & X^X \times X \arrow[u, shift right, "q_2"] \arrow[u, shift left, "a"'] \arrow[ur, "q_1"'] & \end{tikzcd}$$

$\Psi(\Lambda_o) = X$.

Par définition, un modèle M de $\mathbb{G}(\Lambda_o)$ dans une catégorie cartésienne fermée $\mathbb{C}$ s'appelle une *λ-structure dans* $\mathbb{C}$ (pour la raison que ces structures permettent d'interpréter la formation des termes du λ-calcul pur, voir [Barendregt84]).
Le type Λ_o est clairement schématisable dans toute catégorie cartésienne fermée $\mathbb{C}$, car on dispose toujours de la λ-structure dégénérée sur l'objet terminal. Il est sémantisable si et seulement si $\mathbb{C}$ possède une "λ-structure initiale".
Notons au passage que si une λ-structure vérifie

$$\mathcal{C}(M(r))\circ M(s) = Id ,$$

(où $\mathcal{C}$: $Hom(M(X)\times M(X),\ M(X)) \Rightarrow Hom(M(X),\ M(X)^{M(X)})$ est la bijection de Curry), alors M valide la β-conversion et, dans ce cas, on dit que M est une *λ-algèbre* (voir [Barendregt84]). Il est facile de voir que les λ-algèbres sont aussi les modèles d'une certaine trame, mais celle-ci *n'est pas* la géométrie d'un type du système F.

Présentation équationnelle des types sémantisables.

Nous allons montrer maintenant, en utilisant des méthodes de [Lair79], que la notion de type sémantisable peut être présentée de façon *purement équationnelle*.

Fixons quelques notations:
- $\mathbb{O}(A)$ est la trame discrète d'ordre 0 qui a les mêmes objets que $\mathbb{G}(A)$,
- il est facile de construire, à partir de deux copies de $\mathbb{O}(A)$,

une trame $\mathbb{O}_2(A)$ dont les modèles s'identifient aux transformations naturelles entre modèles de $\mathbb{O}(A)$,
- il est facile de construire, à partir de deux copies de $\mathbb{G}(A)$, une trame $\mathbb{G}_2(A)$ dont les modèles s'identifient aux transformations naturelles δ entre modèles de $\mathbb{G}(A)$ qui vérifient $\delta \circ H(A) = Id$,
- de même, il est facile de construire, à partir de trois copies de $\mathbb{G}(A)$, une trame $\mathbb{G}_3(A)$ dont les modèles s'identifient aux triangles commutatifs de telles transformations naturelles.
(On constate que $\mathbb{G}_2(A)$, $\mathbb{G}_3(A)$ ne sont plus nécessairement des graphotrames.)

De plus, on utilisera les homomorphismes de trames suivants, où les schémas symbolisent l'injection de ce qui est en gras dans le tout (par exemple, $D(A)$ est l'injection de la seconde copie de $\mathbb{G}(A)$ dans $\mathbb{G}_2(A)$):

- $D(A) : \mathbb{G}(A) \to \mathbb{G}_2(A)$ injection
- $G(A) : \mathbb{G}(A) \to \mathbb{G}_2(A)$ injection
- $M(A) : \mathbb{O}_2(A) \to \mathbb{G}_2(A)$ injection
- $B(A) : \mathbb{W}(A) \to \mathbb{G}(A)$ injection canonique
- $P(A) : \mathbb{O}(A) \to \mathbb{G}(A)$ injection canonique
- $I(A) : \mathbb{O}_2(A) \to \mathbb{O}(A)$ identifie les objets homologues
- $E(A) : \mathbb{G}_2(A) \to \mathbb{G}_3(A)$ injection
- $F(A) : \mathbb{G}_2(A) \to \mathbb{G}_3(A)$ injection
- $J(A) : \mathbb{G}_2(A) \to \mathbb{G}_3(A)$ injection

Dans ces conditions, on dispose du théorème qui suit.

- Théorème. *Soient* A *un* $\forall$*-type du système* F *et* $\mathbb{C}$ *une catégorie cartésienne fermée. Le type* A *est sémantisable dans* $\mathbb{C}$ *si et seulement si il existe trois applications:*

$$\text{choix}(A) : \text{Mod}(\mathbb{W}(A), \mathbb{C}) \to \text{Mod}(\mathbb{G}(A), \mathbb{C})$$

$$\mathrm{fact}(A) \;:\; \mathrm{Mod}(\mathbb{G}(A), \mathbb{C}) \to \mathrm{Mod}(\mathbb{G}_2(A), \mathbb{C})$$

$$\mathrm{unic}(A) \;:\; \mathrm{Mod}(\mathbb{G}_2(A), \mathbb{C}) \to \mathrm{Mod}(\mathbb{G}_3(A), \mathbb{C})$$

vérifiant les sept équations suivantes:

a) $\mathrm{Mod}(H(A), \mathbb{C}) \circ \mathrm{choix}(A) = \mathrm{Id}_{\mathrm{Mod}(\mathbb{W}(A), \mathbb{C})}$

b_1) $\mathrm{Mod}(D(A), \mathbb{C}) \circ \mathrm{fact}(A) = \mathrm{Id}_{\mathrm{Mod}(\mathbb{G}(A), \mathbb{C})}$

b_2) $\mathrm{Mod}(G(A), \mathbb{C}) \circ \mathrm{fact}(A) = \mathrm{choix}(A) \circ \mathrm{Mod}(B(A), \mathbb{C})$

b_3) $\mathrm{Mod}(M(A), \mathbb{C}) \circ \mathrm{fact}(A) \circ \mathrm{choix}(A)$

$$=$$

$\mathrm{Mod}(I(A), \mathbb{C}) \circ \mathrm{Mod}(F(A), \mathbb{C}) \circ \mathrm{choix}(A)$

b_4) $\mathrm{Mod}(E(A), \mathbb{C}) \circ \mathrm{unic}(A) = \mathrm{fact}(A) \circ \mathrm{Mod}(G(A), \mathbb{C})$

b_5) $\mathrm{Mod}(F(A), \mathbb{C}) \circ \mathrm{unic}(A) = \mathrm{Id}_{\mathrm{Mod}(\mathbb{G}_2(A), \mathbb{C})}$

b_6) $\mathrm{Mod}(J(A), \mathbb{C}) \circ \mathrm{unic}(A) = \mathrm{fact}(A) \circ \mathrm{Mod}(D(A), \mathbb{C})$,

- Preuve. L'assertion b) se réécrit ici:

$\forall\, M \in \mathrm{Mod}(\mathbb{G}(A), \mathbb{C})\ \ \exists!\, \delta : \mathrm{choix}(A)(M\circ H(A)) \to M$ tel que $\delta \circ H(A) = \mathrm{Id}_{M\circ H(A)}$.

Soient donc $M \in \mathrm{Mod}(\mathbb{G}(A), \mathbb{C})$ et $N = \mathrm{choix}(A)(M\circ H(A))$. Alors b_1) et b_2) expriment exactement l'existence d'(au moins) une factorisation $\delta = \mathrm{fact}(A)(M) : N \to M$, telle que $\delta \circ H(A)$ soit égal à $\mathrm{Id}_{M\circ H(A)}$. De plus, les équations b_4), b_5), b_6) jointes à b_3) traduisent équationnellement que toute autre transformation naturelle $\delta' : N \to M$, telle que $\delta' \circ H(A)$ soit égale à $\mathrm{Id}_{M\circ H(A)}$, est égale à δ ; il suffit de vérifier que ce sont des conditions nécessaires, puis de constater que l'application de $\mathrm{unic}(A)$ à δ' donne le diagramme suivant:

$$\begin{tikzcd} N \arrow[r, Rightarrow, "Id"] \arrow[rr, bend left=50, Rightarrow, "\delta"] \arrow[rr, phantom, bend left=25, "="] & N \arrow[r, Rightarrow, "\delta'"] & M \end{tikzcd}$$

qui force $\delta = \delta'$. Fin de la preuve.

Remarques.

Les trois lois (ou opérations algébriques) qui apparaissent dans la présentation équationnelle précédente ont pour arités certaines trames: $\mathbb{W}(A)$, $\mathbb{G}(A)$, $\mathbb{G}_2(A)$. Nous dirons qu'on a une présentation équationnelle tramique.

Pour répondre aux préoccupations des informaticiens, il serait intéressant d' "éliminer la composition" de cette présentation équationnelle, autrement dit de disposer d'une présentation équationnelle *graphotramique*; une telle présentation dispense de devoir tester la commutativité de diagrammes. Or, si $\mathbb{W}(A)$ et $\mathbb{G}(A)$ sont effectivement des graphotrames, nous avons noté que $\mathbb{G}_2(A)$ n'en est pas une. Toutefois, dans certains cas, la donnée d'un modèle de $\mathbb{G}_2(A)$ équivaut à celle d'un modèle d'une graphotrame $\mathbb{G}_{3/2}(A)$: c'est le cas si A est un type naturel (voir le chapitre suivant). On peut alors écrire une présentation équationnelle qui est de la même nature que celle de la structure de catégorie cartésienne fermée dans [Curien86] ou [Lambek-Scott86]. Mais pour $A = \Lambda_0$, ceci n'est pas faisable.

Notons, au passage, que dans [Lambek-Scott86] (resp. [Coquand-Ehrhard87]), les auteurs fournissent une "présentation équationnelle" des catégories cartésiennes fermées (resp. des hyperdoctrines de Seely) en se réclamant de la "notion générale" (resp. de la "tradition") des algèbres graphiques de [Burroni81]. Mais les présentations équationnelles qu'ils indiquent ne suffisent absolument pas à prouver qu'on a bien des algèbres graphiques, en particulier parceque ce sont des présentations graphotramiques et non graphiques! On trouvait déja une mise en garde à ce sujet dans [Dubuc-Kelly83]. Pour une preuve du fait que les petites catégories cartésiennes fermées sont les objets d'une catégorie d'algèbres graphiques voir [MacDonald-Stone86] ou, pour une preuve détaillée, [Ageron-Even88].

4. Types naturels.

Les types que nous appellerons naturels sont probablement les types les plus intéressants du système F. Ils permettent de représenter, en particulier, toutes les structures de données (inductives) usuelles: voir [Girard88]. D'autre part, ils obéissent à des propriétés structurelles simples que nous allons étudier.

Notations et conventions.

Fixons tout d'abord quelques notations et conventions, systématiquement utilisées dans ce n°4, mais aussi aux n^{os} 5 et 6.

Soit A un $\forall$-type du système F .
On considére les homomorphismes de trames:

$\Psi(A) : \langle 1\rangle \to \mathbb{G}(A)$ déjà défini,
$\Phi(A) : \langle 2\rangle \to \mathbb{G}_2(A)$

tel que $\Phi(A)(0) = G(A)(\Psi(A))$ et $\Phi(A)(1) = D(A)(\Psi(A))$.

Si $\mathbb{C}$ est une sémantique catégorique (avec choix) pour A , on pose:

$\underline{\text{choix}}(A) = \text{Mod}(\Psi(A), \mathbb{C}) \circ \text{choix}(A)$
$\underline{\text{fact}}(A) = \text{Mod}(\Phi(A), \mathbb{C}) \circ \text{fact}(A)$.

On notera encore plus simplement $\underline{A} = \underline{\text{choix}}(A)$. Alors:
(i) si A est clos, $\underline{A}$ s'identifie à un objet de $\mathbb{C}$, que l'on notera encore $\underline{A}$,
(ii) si A a exactement une variable libre X, $\underline{A}$ s'identifie à une application de $\text{Ob}(\mathbb{C})$ dans $\text{Ob}(\mathbb{C})$, que l'on notera encore

$\mathbb{A}$,
(iii) si A a au plus une variable libre X , on convient d'identifier $\mathbb{A}$ à une application de $Ob(\mathbb{C})$ dans $Ob(\mathbb{C})$, qui est celle définie en ii) si A a exactement une variable libre et qui est l'application constante de valeur $\mathbb{A}$ si A est clos .

Enfin on utilisera les mêmes lettres X, Y, ... pour désigner tantôt des variables de type, tantôt des objets de $\mathbb{C}$.

Variables complètement positives.

Soit A un type du système F . On définit inductivement l'ensemble $CP(A)$ des variables de type *complètement positives* dans A par:

$$CP(X) = W$$
$$CP(A_1 \Rightarrow A_2) = CP(A_2) \setminus W(A_1)$$
$$CP(A) = \bigcap_{(i,j)\in K} CP(A^j{}_i) \quad \text{si } A \text{ est un } \forall\text{-type.}$$

Types naturels.

On dit qu'un $\forall$-type A du système F est *naturel* si, et seulement si il vérifie les deux conditions suivantes:

- $A = \forall X.A_1 \Rightarrow (A_2 \Rightarrow \ldots \Rightarrow (A_n \Rightarrow X) \ldots) \qquad n \in \mathbb{N}$

où chaque type A_i est de la forme:

$$A_i = A^1{}_i \Rightarrow (A^2{}_i \Rightarrow \ldots \Rightarrow (A^{k_i}{}_i \Rightarrow X) \ldots) \qquad k_i \in \mathbb{N}$$

avec X complètement positive dans chacun des $A_i{}^j$,

- tous les $\forall$-sous-types de A sont naturels.

- Théorème préparatoire. *Soit* A *un type naturel du système* F *admettant au plus une variable libre* X, *supposée complètement positive dans* A . *Soit* $(\mathbb{C}, \text{choix}(A))$ *une sémantique catégorique pour* A . *Alors, l'application*

$$\mathbb{A}: Ob(\mathbb{C}) \to Ob(\mathbb{C})$$

est l'induction aux objets d'un endofoncteur de $\mathbb{C}$.

- Preuve. Soit $x : X \to X'$ une flèche de $\mathbb{C}$. On voudrait définir $\mathbb{A}(x) : \mathbb{A}(X) \to \mathbb{A}(X')$. On dispose de M, modèle de $\mathbb{G}(A)$ initial parmi ceux qui sont construits sur X, et de M' modèle de $\mathbb{G}(A)$ initial parmi ceux qui sont construits sur X'. A partir de M' et de x, il est possible de construire un autre modèle M_1 de $\mathbb{G}(A)$ construit sur X et tel que l'on ait $M(\Psi(A)) = \mathbb{A}(X')$. Alors il existe une unique transformation $\delta : M \to M_1$ telle que $\delta_X = Id_X$; on obtient donc une flèche de $\mathbb{A}(X)$ vers $\mathbb{A}(X')$ que l'on appelle $\mathbb{A}(x)$. La fonctorialité se vérifie alors facilement. Fin de la preuve.

Remarquons qu'il est, de même, bien connu que pour tout objet C de $\mathbb{C}$, l'application:

$$\begin{array}{ccc} Ob(\mathbb{C}) & \to & Ob(\mathbb{C}) \\ X & \mapsto & X^C \end{array}$$

est l'induction aux objets d'un endofoncteur de $\mathbb{C}$ (démonstration analogue).

Sémantisabilité des types naturels.

Nous avons d'abord la proposition suivante.

- Proposition. *Soit* A *un type naturel. Alors* A *est schématisable dans toute catégorie cartésienne fermée* $\mathbb{C}$.

- Preuve. Il existe toujours un modèle vérifiant

$$M(\Psi(A)) = 1 .$$

Fin de la preuve.

Considérons maintenant un type naturel A, que, pour simplifier, nous supposerons clos. Soit $\mathbb{C}$ une catégorie cartésienne fermée qui soit déjà une sémantique (avec choix) pour tous les types $A^j{}_i$: d'après le théorème préparatoire et la remarque qui le suit, on peut associer à chaque $A^j{}_i$ un

endofoncteur de $\mathbb{C}$ noté $T^j{}_i$. On considère alors la famille d'endofoncteurs $(T_1, \ldots, T_n)$ où $T_i = T^1{}_i \times \ldots \times T^{k_i}{}_i$. Enfin, on note $Alg(T_1, \ldots, T_n)$ la catégorie ainsi définie:
- ses objets sont les cônes inductifs de $\mathbb{C}$ de la forme

$$x = (x_i : T_i(X) \rightarrow X)_{1 \leqslant i \leqslant n}$$

- une flèche de $x = (x_i : T_i(X) \rightarrow X)$ vers $y = (y_i : T_i(Y) \rightarrow Y)$ est une flèche $h : X \rightarrow Y$ telle que $h.x_i = y_i.T_i(h)$ pour $1 \leqslant i \leqslant n$.

Moyennant ces notations, nous pouvons établir le théorème "de sémantisabilité" suivant.

- Théorème. *Le type naturel* A *est sémantisable dans* $\mathbb{C}$ *si et seulement si la catégorie* $Alg(T_1, \ldots, T_n)$ *admet un objet initial.*

- Preuve. Le type A est sémantisable si, et seulement si:

$$\mathbb{C}^{\mathbb{G}(A)} \text{ admet un objet initial.}$$

Or les catégories $\mathbb{C}^{\mathbb{G}(A)}$ et $Alg(T_1, \ldots, T_n)$ sont isomorphes (remarquer qu'une transformation entre deux modèles de $\mathbb{G}(A)$ est entièrement définie par sa trace h en $\mathbb{V}(A)$). Fin de la preuve.

Nous pouvons également établir les deux lemmes qui suivent.

- Lemme 1. *Soit* T *un endofoncteur d'une catégorie* $\mathbb{C}$. *Supposons que* $Alg(T)$ *admette un objet initial* $a: T(A) \rightarrow A$. *Alors* a *est un isomorphisme de* $T(A)$ *sur* A .

- Preuve. Par initialité de a, il existe une unique flèche $h : A \rightarrow T(A)$ telle que $h.a = T(a).T(h)$, ou encore $h.a = T(a.h)$. On en déduit l'égalité suivante: $(a.h).a = a.T(a.h)$. Mais on a aussi évidemment: $Id_A.a = Id_{T(A)}$. En utilisant à nouveau l'initialité de a , on conclut que $a.h = Id_A$ et on obtient aussi $h.a = T(a.h) = T(Id_A) = Id_{T(A)}$. Alors a est bien un isomorphisme, d'inverse h . Fin de la preuve.

- Lemme 2. *Soient* $T_1, \dots, T_n$ *des endofoncteurs d'une catégorie* $\mathbb{C}$ *. Supposons que pour tout objet* X *de* $\mathbb{C}$ *, la somme* $T_1(X) + \dots + T_n(X)$ *existe, et que* $Alg(T_1, \dots, T_n)$ *admette un objet initial* a *. Alors* a *est un cône somme.*

- Preuve. Soit Σ le foncteur $T_1 + \dots + T_n$: par définition d'une somme, on a $Alg(T_1, \dots, T_n) \cong Alg(\Sigma)$. Supposons maintenant $a = (a_1, \dots, a_n)$ initial dans $Alg(T_1, \dots, T_n)$. Alors $(a_1, \dots, a_n)$ est initial dans $Alg(\Sigma)$; par le lemme précédent, c'est un isomorphisme, d'où il résulte que $(a_1, \dots, a_n)$ est un cône somme. Fin de la preuve.

On en déduit le corollaire qui suit.

- Corollaire. *Si le type naturel* A *est sémantisable dans* $\mathbb{C}$ *et si* $\mathbb{C}$ *admet les sommes finies, alors l'objet* A *est solution de l'équation de point fixe:*

$$X \cong T_1(X) + \dots + T_n(X) \qquad X \in Ob(\mathbb{C}).$$

- Preuve. Ceci résulte immédiatement du théorème et des deux lemmes précédents. Fin de la preuve.

Exemples.

1) Considérons le type naturel clos:

$$ent = \forall X. X \Rightarrow ((X \Rightarrow X) \Rightarrow X).$$

Ce type est sémantisable dans $\mathbb{C}$ si, et seulement si, $\mathbb{C}$ admet un "objet des entiers naturels". Si $\mathbb{C}$ admet de plus des sommes finies, l'objet $N = \underline{ent}$ satisfait l'équation $N \cong 1+N$. Pour ces raisons, ent est appelé "type des entiers naturels" (on donne parfois ce nom au type $ent' = \forall X.(X \Rightarrow X) \Rightarrow (X \Rightarrow X)$ qui a la même géométrie).

2) Considérons le type naturel clos $nil = \forall X. X$. Ce type est sémantisable dans $\mathbb{C}$ si, et seulement si, $\mathbb{C}$ admet un objet initial 0 . On a alors $\underline{nil} \cong 0$ et nil s'interprète comme "type vide".

3) Considérons le type naturel clos:

$$\text{bool} = \forall X. X \Rightarrow (X \Rightarrow X) \ .$$

Ce type est sémantisable dans $\mathbb{C}$ si, et seulement si, la somme $1 + 1$ existe . On a alors $\underline{\text{bool}} \cong 1 + 1$ et bool s'interprète comme "type des booléens".

4) Considérons le type naturel à deux variables libres:

$$\text{somme}(Y,Z) = \forall X. (Y \Rightarrow X) \Rightarrow ((Z \Rightarrow X) \Rightarrow X) \quad .$$

Ce type est sémantisable dans $\mathbb{C}$ si, et seulement si, $\mathbb{C}$ a les sommes de deux objets. On a alors, pour Y et $Z \in Ob(\mathbb{C})$, $\underline{\text{somme}}(Y,Z) \cong Y + Z$ et somme(Y,Z) s'interprète comme "type somme des types Y et Z" .

5) Considérons le type naturel clos:

$$\text{arbrebin} = \forall X. X \Rightarrow ((X \Rightarrow (X \Rightarrow X)) \Rightarrow X) \ .$$

Si $\mathbb{C}$ en est une sémantique et admet les sommes finies, $\underline{\text{arbrebin}}$ est solution de $X \cong 1 + X^2$. Et arbrebin s'interprète comme "type des arbres binaires".

6) Considérons le type naturel à une variable libre:

$$\text{list}(Y) = \forall X. X \Rightarrow ((X \Rightarrow (Y \Rightarrow X)) \Rightarrow X).$$

Si $\mathbb{C}$ en est une sémantique et admet les sommes finies, et si $Y \in Ob(\mathbb{C})$, $\underline{\text{list}}(Y)$ est solution de $X \cong 1 + Y \times X$. Et list(Y) s'interprète comme "type des listes d'entités de type Y ".

7) Considérons le type naturel à trois variables libres:

$$\text{arbre}(S, \Omega, V) = \forall X. (V \Rightarrow X) \Rightarrow ((\Omega \Rightarrow ((S \Rightarrow X) \Rightarrow X)) \Rightarrow X)$$

Si $\mathbb{C}$ en est une sémantique et admet les sommes finies et si $S, \Omega, V \in Ob(\mathbb{C})$, alors $\underline{\text{arbre}}(S, \Omega, V)$ est solution de $X \cong V + \Omega \times X^S$. Et arbre(S, Ω, V) s'interprète comme "type des arbres à embranchements de type S , à sommets internes de type Ω et à sommets externes de type V ".

8) Considérons le type naturel clos:

$$\mathrm{on} = \forall X. X \Rightarrow ((X \Rightarrow X) \Rightarrow (((\mathrm{ent} \Rightarrow X) \Rightarrow X) \Rightarrow X)) .$$

Si $\mathbb{C}$ en est une sémantique et admet les sommes finies, $\underline{\mathrm{on}}$ est solution de $X \cong 1+X+X^N$, N étant un objet des entiers naturels.
Dans [Coquand-Huet87], on est interprété comme "type des ordinaux", ou plus précisément "type des ordinaux dénombrables présentés comme limites de suites fondamentales de Cantor".

9) Considérons le type naturel clos:

$$\mathrm{arbrefini} = \forall X. (\mathrm{list}(X) \Rightarrow X) \Rightarrow X.$$

Si $\mathbb{C}$ en est une sémantique, l'objet $\underline{\mathrm{arbrefini}}$ est solution de $X \cong \underline{\mathrm{list}}(X)$ et arbrefini s'interprète comme "type des arbres abstraits à embranchements finis".

10) et 11). Le cas des types:

$$\mathrm{un} = \forall X. X \Rightarrow X$$

$$\mathrm{prod}(Y,Z) = \forall X. (Y \Rightarrow (Z \Rightarrow X)) \Rightarrow X$$

est un peu particulier. Ces types sont sémantisables dans toute catégorie cartésienne fermée, avec $\underline{\mathrm{un}} = 1$ et $\underline{\mathrm{prod}}(Y,Z) = Y \times Z$ pour $Y, Z \in \mathrm{Ob}(\mathbb{C})$.
Il est clair que ces types sont, "profondément", des types projectifs et non inductifs. Mais une "astuce" (inspirée, paraît-il, de Russel) consistant à leur ajouter un étage projectif "artificiel" permet de les intégrer au système F.

5. termes.

Préliminaires.

Voici quelle est l'interprétation informatique (algorithmique) des types de la forme $\forall X. A(X)$: le fait que $\forall X. A(X)$ soit sémantisable dans $\mathbb{C}$ exprime que les algorithmes uniformes de type $A(X)$ (j'entends par là les algorithmes qui existent "à coup sûr", quel que soit l'objet X de $\mathbb{C}$) peuvent être décrits au moyen de certains algorithmes d'un certain type clos. Ceux-ci peuvent être vus comme des substrats opératoires abstraits, mais implantables dans le langage de $\mathbb{C}$.

Dans le système F , les algorithmes sont représentés par les termes: voir leur syntaxe dans [Girard 88].

Nous allons donc dire un mot de la sémantique des termes du système F ; nous nous bornerons ici à l'interprétation (dans un cas particulier suffisamment générique) des schémas d'*extraction* et d'*abstraction universelle*.

Extraction.

Soient A un type du système F ayant au plus la variable libre X et $\mathbb{C}$ une sémantique catégorique pour A et $\forall X.A$. On suppose que 1 est générateur dans $\mathbb{C}$. Alors on a la règle d'extraction:

$$\frac{u : 1 \to \underline{\forall X.A} \qquad X \in Ob(\mathbb{C})}{\varepsilon((u,X) : 1 \to \underline{A}(X)}$$

Cette opération correspond à la construction suivante
Soit $u : 1 \to \underline{\forall X.A}$. Considérons l'application:

$$\begin{array}{llll} \mathcal{E}_u : & Mod(\mathbb{G}(\forall X.A), \mathbb{C}) & \to & Fl(\mathbb{C}) \\ & M & \mapsto & \underline{fact}(\forall X.A)(M).u \end{array}$$

Pour tout objet X de $\mathbb{C}$, $\mathcal{E}_u$ induit, par définition de $\mathbb{G}(\forall X.A)$, une application:

$$\mathcal{E}_{u,X} : \prod_{1\leq i\leq n} Hom_{\mathbb{C}}\Big(\prod_{1\leq j\leq k_i} \underline{A}^j{}_i(X) , \underline{A}^o{}_i(X)\Big) \to Hom_{\mathbb{C}}(1, \underline{A}^o{}_o(X))$$

donc, par bijection de Curry, une application:

$$\mathcal{E}_{u,X}{}^{\sim} : Hom_{\mathbb{C}}\Big(1 , \prod_{1\leq i\leq n} \underline{A}_i(X)\Big) \to Hom_{\mathbb{C}}(1 , \underline{A}^o{}_o(X)) .$$

Comme on suppose que 1 est générateur, celle-ci détermine à son tour une unique flèche de $\mathbb{C}$:

$$\varepsilon^{\sim}(u,X) : \prod_{1\leq i\leq n} \underline{A}_i(X) \rightarrow \underline{A}^{\circ}{}_0(X)$$

et donc finalement une flèche:

$$\varepsilon(u,X) : 1 \rightarrow \underline{A}(X).$$

Abstraction universelle

Soit A un type du système F ayant au plus la variable libre X et soit $\mathbb{C}$ une sémantique catégorique pour A et $\forall X.A$. Alors on a la règle d'abstraction universelle:

$$\frac{t = (t_X : 1 \rightarrow \underline{A}(X))_{X \in ob(\mathbb{C})}}{\Lambda X.t : 1 \rightarrow \underline{\forall X.A}}$$

La construction correspondante est laissée au lecteur.

6. Etude d'un type non ensembliste.

Types ensemblistes.

Disons d'un type A du système F qu'il est *ensembliste* si et seulement si:

- il est sémantisable dans la catégorie (cartésienne fermée) $\mathbb{Ens}$. L'existence de types non ensemblistes montre en particulier que l'idée géniale de Heyting, consistant à identifier une formule à l'ensemble de ses démonstrations, ne peut pas être prise au pied de la lettre.

Exemple de type non ensembliste.

Dans ce qui suit, nous considérons le type suivant:

$$L_1 = \forall X.\mathrm{list}(X) \Rightarrow (X \Rightarrow \mathrm{list}(X)).$$

- Théorème. *Le type* L_1 *n'est pas ensembliste.*

- Preuve. Il est clair que list(-) est un constructeur de type ensembliste, c'est à dire que pour tout ensemble X , il existe dans $\mathbb{E}\mathrm{ns}$ un "objet des listes d'entités de type X ":

$$1 \xrightarrow{(\ )^X} \underline{\mathrm{list}}(X) \xleftarrow{\mathrm{append}^X} \underline{\mathrm{list}}(X)\times X$$

De plus, chaque $l \in \underline{\mathrm{list}}(X)$ a une longueur $\mathrm{LONG}(l) \in \mathbb{N}$ déterminée, telle que:

$$\mathrm{LONG}((\ )^X) = 0 \quad \text{et} \quad \mathrm{LONG}(\mathrm{append}^X(\xi,x)) = \mathrm{LONG}(\xi)+1 \ .$$

(Remarquons en passant que le terme $\Lambda X.\mathrm{append}^X$ est précisément de type L_1 , tandis que $\Lambda X.(\ )^X$ est de type $L_0 = \forall X.\mathrm{list}(X)$ qui, lui, est ensembliste.)
Supposons que L_1 soit un type ensembliste, ce qui revient à dire que la trame $\mathbb{G}(L_1)$ admet un modèle initial dans $\mathbb{E}\mathrm{ns}$. Alors il existe

$$(A \ , \ a \in A \ , \ \alpha \in \underline{\mathrm{list}}(A))$$

tel que pour tout

$$(X \ , \ x \in X \ , \ \xi \in \underline{\mathrm{list}}(X))$$

il existe un et un seul triplet d'applications

$$\begin{aligned}(h &: A \to X \ , \\ j &: \underline{\mathrm{list}}(A) \to \underline{\mathrm{list}}(X) \ , \\ k &: \underline{\mathrm{list}}(A)\times A \to \underline{\mathrm{list}}(X)\times X \ )\end{aligned}$$

tel que

$$j((\ )^A) = (\ )^X$$

$$j \circ \mathrm{append}^A = \mathrm{append}^X \circ k$$
$$j \circ pr^1_{\underline{list}(A),A} = pr^1_{\underline{list}(X),X} \circ k$$
$$h \circ pr^2_{\underline{list}(A),A} = pr^2_{\underline{list}(X),X} \circ k$$
$$h(a) = x$$
$$j(\alpha) = \xi.$$

Nous allons montrer par récurrence que ces conditions impliquent:

$$\forall \lambda \in \underline{list}(A) \ , \ \mathrm{LONG}(j(\lambda)) = \mathrm{LONG}(\lambda).$$

Si $\lambda = (\ )^A$, c'est clair.
Si $\lambda = \mathrm{append}^A(\lambda_0, a_0)$, où on suppose $\mathrm{LONG}(j(\lambda_0)) = \mathrm{LONG}(\lambda_0)$, alors:

$$\begin{aligned} j(\lambda) &= j(\mathrm{append}^A(\lambda_0, a_0)) \\ &= \mathrm{append}^X(k(\lambda_0, a_0)) \\ &= \mathrm{append}^X(j(pr^1(\lambda_0, a_0)) \ , \ h(pr^2(\lambda_0, a_0))) \\ &= \mathrm{append}^X(j(\lambda_0) \ , \ h(a_0)) \end{aligned}$$

et donc $\mathrm{LONG}(j(\lambda)) = \mathrm{LONG}(j(\lambda_0))+1 = \mathrm{LONG}(\lambda_0)+1 = \mathrm{LONG}(\lambda)$.
En particulier, on doit donc avoir $\mathrm{LONG}(\alpha) = \mathrm{LONG}(\xi)$. Mais ξ peut avoir n'importe quelle longueur, d'où la contradiction. Fin de la preuve.

On a cependant le résultat plus faible suivant:
- soit (X, x, ξ) avec $\mathrm{LONG}(\xi) = q$ et supposons (A, a, α) tel que:

$$\mathrm{CARD}(A) = q+1$$
$$\alpha = (a, a_1, \ldots, a_q)$$
$$\mathrm{CARD}(\{a, a_1, \ldots, a_q\}) = q+1 \ ,$$

alors il existe un unique (h, j, k) "faisant tout commuter".

Autrement dit, si la trame $\mathbb{G}(L_1)$ ne possède pas de modèle initial dans $\mathbf{Ens}$, elle possède quand même un certain "diagramme localement initial" (voir [Guitart-Lair80] pour cette notion) de modèles, ce diagramme étant - et c'est cela qui est remarquable - petit (et même dénombrable). On peut dire en ce sens que $\mathbf{Ens}$ est une "sémantique locale" pour le type L_1 .

L'interprétation algorithmique est ici probablement la suivante:
- certains algorithmes uniformes de type:

$$\mathrm{list}(X) \Rightarrow (X \Rightarrow \mathrm{list}(X))$$

ne peuvent être décrits sans recourir à une notion extérieure, à savoir la longueur de la liste prise en argument.

REFERENCES BIBLIOGRAPHIQUES

[Ageron-Even] Pierre Ageron et Christian Even:

Etude de la catégorie des catégories cartésiennes fermées, à paraître.

[Barendregt84] Hendrik Pieter Barendregt:

The lambda calculus, deuxième édition, North-Holland, 1984.

[Burroni81] Albert Burroni:

Algèbres graphiques, Cahiers de Topologie et Géométrie Différentielle, vol. XXII-3, Amiens, 1981.

[Coppey84] Laurent Coppey:

Catégories de Peano, catégories algorithmiques, récursivité, Diagrammes 12, Paris, 1984.

[Coppey-Lair85] Laurent Coppey et Christian Lair:

Algébricité, monadicité, esquissabilité et non-algébricité, Diagrammes 13, Paris, 1985.

[Coquand-Huet87] Thierry Coquand et Gérard Huet:

Concepts mathématiques et informatiques formalisés dans le calcul des constructions, in Logic Colloquium 85, edited by Paris Logic Group, North-Holland, 1987.

[Coquand-Ehrhard87] Thierry Coquand et Thomas Ehrhard:

An equational presentation of higher order logic, in Proceedings of Category Theory and Computer Science, edited by D. H. Pitt, A. Poigné, D.E.Rydeheard, Lect. Notes in Comp. Sci. 283, Springer, 1987.

[Coquand] Thierry Coquand:

Categories of embeddings, à paraître.

[Curien86] Pierre-Louis Curien:

Categorical combinators, Sequential Algorithms and Functional Programming, Pitman, 1986.

[Dubuc-Kelly83] Eduardo Dubuc et Max G. Kelly:

A presentation of topoï as algebraic relative to categories or graphs, Journal of Algebra, vol. 81-2, 1983.

[Ehresmann68] Charles Ehresmann:

Esquisses et types des structures algébriques, Bul. Instit. Polit. Iaşi, vol. XIV, 1968.

[Freyd-Girard-Scedrov-Scott] Peter J. Freyd, Jean-Yves Girard, Andre Scedov et Philip J. Scott:

Semantic parametricity in polymorphic λ-calculus, à paraître.

[Girard71] Jean-Yves Girard:

Une extension de l'interprétation fonctionnelle de Gödel à l'analyse et son application à l'élimination des coupures dans l'analyse et la théorie des types, in Proceedings of the second Scandinavian Logic Symposium, Oslo, 1970, edited by J.E Fenstad, North-Holland, 1971.

[Girard86] Jean-Yves Girard:

The system F of variable types, fifteen years later, Theoretical Computer Science, vol. 45, 1986.

[Girard] Jean-Yves Girard:

Typed λ-calculus, Cambridge Tracts in Theoretical Computer Science, à paraître.

[Guitart-Lair80] René Guitart et Christian Lair:

Calcul syntaxique des modèles et calcul des formules internes, Diagrammes 4, Paris, 1980.

[Huet] Gérard Huet:

A uniform approach to type theory, à paraître.

[Lair75] Christian Lair:

Esquissabilité et triblabilité, 2ième colloque sur les catégories, Amiens 75, Cahiers de Topologie et Géométrie Différentielle, vol. XVI-3, Amiens, 1975.

[Lair87] Christian lair:

Trames et sémantiques catégoriques des systèmes de trames, Diagrammes 18, Paris, 1987.

[Lair79] Christian Lair:

Condition syntaxique de triplabilité d'un foncteur algébrique esquissé, Diagrammes 1, Paris, 1979.

[Lambek-Scott86] Joachim Lambek et Philip J. Scott:

Introduction to higher order categorical logic, Cambridge Univ. Press, 1986.

[MacDonald-Stone84] John MacDonald et Arthur Stone:

Topoï over graphs, Cahiers de Topologie et Géométrie Différentielle, vol. XXV-1, Amiens, 1984.

[Pitts87] Andrew M. Pitts:

Polymorphism is set-theoretic constructively, in Proceedings of Category Theory and Computer Science, edited by D. H. Pitt, A. Poigné, D.E. Rydeheard, Lect. Notes in Comp. Sci. 283, Springer, 1987.

[Reynolds84] John C. Reynolds:

Polymorphism is not set-theoretic, in Semantics of Data Types, edited by G. Kahn et al. , Lect. Notes in Comp. Sci. 173, Springer, 1984.

REFERENCES BIBLIOGRAPHIQUES

[Seely87] R. A. G. Seely;

Categorical semantics for higher order polymorphic λ- calculus, The Journal of Symbolic Logic, vol. 52-4, 1987 .

UNIVERSITE DE CAEN

DEPARTEMENT DE MATHEMATIQUES

ESPLANADE DE LA PAIX

14032 CAEN CEDEX

FRANCE