Structure sémantique des textes mathématiques
Le cas des démonstrations

Laurent Roussarie

Réunion DémoNat
7 avril 2005 – Nancy, Loria

Corpus

Théorème 1

(1) Pour toutes suites u et v et pour tout réel l, si v est extraite de u et si u converge vers l, alors v converge vers l.

Preuve de René :

(2) a. Soient u,v,l. On suppose que v est extraite de u et que u converge vers l. Montrons que v converge vers l.

b. Soit $e >> zero$. On cherche $n1$ tq pour tout $n > n1$ $d(v_n, l) << e$.

c. Puisque v est extraite de u, soit f croissante tq pour tout n $v_n = u_f(n)$.

d. Puisque u converge vers l soit $n0$ tq pour tout $n > n0$ $d(u_n, l) << e$.

e. Soit $n > n0$, montrons que $d(v_n, l) << e$.

f. Montrons que l'on a $f(n) > n0$.

g. Puisque f est croissante on a $f(n) > f(n0)$.

h. Puisque f est croissante, par le lemme 1 on a $f(n0) >= n0$.

i. On conclut alors par la transitivité.

j. On a donc $d(u_f(n), l << e$.

k. On conclut par H2.

Preuve d'étudiant 1 :

(3) a. Soit $e > 0$, $(Vn)n$ une sous suite extraite de $(Un)n$ et tel que $(Un)n$ converge vers l, montrons que $(Vn)n$ converge vers l, c'est-à-dire : on cherche $n1$ tel que pour tout $n > n1$ $d(Vn, l) < e$.

b. Comme $(Vn)n$ est une suite extraite alors il existe une fonction f croissante tel que $Vn = U(f(n))$, pour tout n.

c. Comme $e > 0$ et $(Un)n$ converge vers l, alors il existe $n0$ tel que pour tout $n > n0$ $d(Un, l) < e$.

d. Soit $n > n0$, montrons qu'alors $d(Vn, l) < e$.

e. Mais maintenant, si $f(n) > n0$, comme $Vn = Uf(n)$, alors on aura démontré que $(Vn)n$ converge vers l.

f. Comme f est croissante et que $n > n0$, on a $f(n) > f(n0) >= n0$, d'où $f(n) > n0$. Comme $f(n) > n0$ alors $d(Uf(n), l)n0$, $(Vn)n$ converge vers l.

Preuve d'étudiant 2 :

(4) a. Soit u une suite de E convergeant vers l et v une suite extraite de u. Montrons que v converge aussi vers l.

b. Soit e > 0, montrons qu'il existe n0 tel que pour tout n > n0 d (v(n), l) « e.

c. Comme v est une suite extraite de u, il existe une application f croissante et pour tout n, v(n) = u(f(n)).

d. L'application f étant croissante, on a : f(n) >=n pour tout n.

e. La suite u est une suite qui converge vers l. Donc pour e, il existe n0 tel que pour tout n > n0 , (d (u n) l) « e.

f. Soit n, on a : f(n)>= n> n0, donc d(u(f(n)), l)«e. On a aussi v(n) = u(f(n)) donc d(v(n),l)«e.

g. On a donc montré que pour tout e >0, il existe n0 tel que pour tout n>n0, d(v(n),l)«e, c'est-à-dire v(n) converge vers l. Cqfd.

Théorème 2

(5) Soit u une suite et l un réel. Si toute suite extraite de u qui converge converge vers l alors u converge vers l.

Preuve de René :

(6) a. Soient u et l. On suppose que toute suite extraite de u convergente converge vers l. Montrons que u converge vers l.

b. On raisonne par l'absurde.

c. D'après le lemme 2, soit $e >> zero$ et v tq v est extraite de u et pour tout n $d(v_n, l) <<$ e.

d. Puisque E est compact, soit $v0$ une suite extraite de v qui converge.

e. Puisque $v0$ est extraite de v soit f croissante tq pour tout n $v0_n = v_f(n)$.

f. D'apres le lemme 3 $v0$ est extraite de u.

g. D'après l'hypothèse, $v0$ converge vers l.

h. Soit donc $n0$ tq pour tout $n > n0$ $d(v0_n, l) << e$.

i. D'apres le lemme successeur on a $s\ n0 > n0$.

j. Comme $v0_(Sn0) = v_f(Sn0)$ et $d(v_f(Sn0), l) << e$ et $d(v_f(Sn0), l) >> e$ on conclut à une absurdité.

Preuve de Patrick :

(7) a. soit u une suite, soit l un réel, supposons que toute suite extraite de u qui converge converge vers l.

b. Supposons que u ne converge pas vers l.

c. Alors il existe un $e >> zero$ et une suite extraite v de u telle que $\forall n\ d(vn)l >> e$ [I].

d. Or E est compact, donc il existe w extraite de v qui converge.

e. w est extraite de u et converge, donc w converge vers l par hypothèse.

f. Soit f telle que $\forall n\ wn = u(fn)$.

g. Il existe donc $n0$ tel que $\forall n > n0\ d(u(fn)) << e$.

h. En particulier $d(u(f(Sn0)))l << e$.

i. ceci contredit I. Donc u converge vers l.

Preuve de Etudiant 1 :

(8) a. Soit une suite $(Un)n$ et l dans E, on suppose que toute suite $(Vn)n$ extraite de $(Un)n$ convergente, converge vers l.

b. Montrons par l'absurde que $(Un)n$ converge vers l.

c. Comme $(Un)n$ ne converge pas vers l, d'après le lemme 2, il existe $e > 0$ et il existe une suite extraite $(Vn)n$ de $(Un)n$ tel que $(Vn)n$ ne converge pas vers l.

d. De plus, comme E est compacte, on peut trouver une suite extraite $(V0n)n$ de $(Vn)n$ tel que $(V0n)n$ converge.

e. Dès lors, comme $(V0n)n$ est une suite extraite de $(Vn)n$ et que $(Vn)n$ est une suite extraite de $(Un)n$ alors $(V0n)n$ est aussi une suite extraite de $(Un)n$, d'après le lemme 3.

f. Comme $(V0n)n$ est une suite extraite de $(Vn)n$ alors il existe f, une fonction croissante, tel que pour tout n : $V0n = Vf(n)$; en particulier pour $n = S(n0)$.

g. $n0$ n'existe pas !!! Donc ceci ne passe pas : deduce v0 (S n0) = v (f (S n0)).

1 Enoncés des théorèmes

Les deux exemples du corpus ne posent pas problème particulier par rapport aux réflexions sur les textes de définitions ([3]). D'autant plus que c'est un français très logique du 1er ordre : la transcription est assez directe.

DRS-output attendue (raccourci de notation : u::`"a"` $= u$ s'appelle a) :

(1) Pour toutes suites u et v et pour tout réel l, si v est extraite de u et si u converge vers l, alors v converge vers l.

$$\begin{array}{|l|}\hline \\ \hline \begin{array}{|ll|}\hline u_1\ u_2\ u_3 & \\ \hline u_1::\texttt{"u"} & \text{suite}(u_1)\\ u_2::\texttt{"v"} & \text{suite}(u_2)\\ u_3::\texttt{"l"} & \text{réel}(u_3)\\ \multicolumn{2}{|l|}{\text{extraite-de}(u_2,u_1)}\\ \multicolumn{2}{|l|}{\text{converger}(u_1,u_3)}\\ \hline \end{array} \Rightarrow \begin{array}{|l|}\hline \\ \hline \text{converger}(u_2,u_3)\\ \hline \end{array} \\ \hline \end{array}$$

ou

$$\begin{array}{|l|}\hline \\ \hline \begin{array}{|ll|}\hline u_1\ u_2\ u_3 & \\ \hline u_1::\texttt{"u"} & \text{suite}(u_1)\\ u_2::\texttt{"v"} & \text{suite}(u_2)\\ u_3::\texttt{"l"} & \text{réel}(u_3)\\ \hline \end{array} \Rightarrow \begin{array}{|l|}\hline \\ \hline \begin{array}{|l|}\hline u_4\ u_5\ u_6\\ \hline u_4::\texttt{"v"}\\ u_5::\texttt{"u"}\\ \text{extraite-de}(u_4,u_5)\\ u_6::\texttt{"l"}\\ \text{converger}(u_5,u_6)\\ \hline \end{array} \Rightarrow \begin{array}{|l|}\hline u_7\ u_8\\ \hline u_7::\texttt{"v"}\\ u_8::\texttt{"l"}\\ \text{converger}(u_7,u_8)\\ \hline \end{array} \\ \hline \end{array} \\ \hline \end{array}$$

Il est envisageable qu'un analyseur compositionnel sorte une DRS plus compliquée (ie pas aussi compacte), plus proche de la forme de surface, mais dont la sémantique est équivalente.

En guise d'illustration, je donne la représentation du second théorème dans la version variante, la SDRT.

(5) Soit u une suite et l un réel. Si toute suite extraite de u qui converge converge vers l alors u converge vers l.

$$\begin{array}{|l|}\hline \pi_0 \\ \hline \pi_0 : \begin{array}{|l|}\hline \pi_1\ \pi_2^* \\ \hline \pi_1 : \begin{array}{|ll|}\hline u_1\ u_2 & \\ \hline u_1::\texttt{"u"} & \text{suite}(u_1)\\ u_2::\texttt{"l"} & \text{réel}(u_2)\\ \hline \end{array} \\ \pi_2^* : \begin{array}{|l|}\hline \pi_2\ \pi_2' \\ \hline \pi_2 : \begin{array}{|l|}\hline \\ \hline \begin{array}{|l|}\hline u_3\ u_4\\ \hline \text{suite}(u_3)\\ \text{extraite}(u_3,u_4)\\ u_4::\texttt{"u"}\\ \text{converger}(u_3)\\ \hline \end{array} \Rightarrow \begin{array}{|l|}\hline u_5\\ \hline \text{converger}(u_3,u_5)\\ u_5::\texttt{"l"}\\ \hline \end{array} \\ \hline \end{array} \quad \pi_2' : \begin{array}{|l|}\hline u_6\ u_7\\ \hline u_6::\texttt{"u"}\\ u_7::\texttt{"l"}\\ \text{converger}(u_6,u_7)\\ \hline \end{array} \\ \textit{Conseq}(\pi_2,\pi_2') \\ \hline \end{array} \\ \textit{Conseq}(\pi_1,\pi_2^*) \\ \hline \end{array} \\ \hline \end{array}$$

Avec postulat de signification entre converger/1 et converger/2 :
$\Box\forall x[\text{converger}(x) \rightarrow \exists y\, \text{converger}(x, y)]$

2 Quelques cas

2.1 *Puisque*, *Comme*, etc.

Comme/Puisque ϕ, (alors) ψ.

Interprétation :

1. ϕ est connue (ie présupposée) ;
2. (a) la vérité de ψ découle de la vérité de ϕ et
   (b) ψ.
   $= \text{cause}(\phi, \psi) \wedge \psi$

Le prédicat 'cause' est d'ordre supérieur[1] et sa sémantique exacte est probablement un abîme philosophique. Par ailleurs on aurait intérêt à tenir compte de deux sources causales : la cause générale et la cause particulière[2]. On aura alors un prédicat à 3 places : $\text{cause}(L, \phi, \psi)$ où L est une loi — celle en vertu de laquelle ψ découle de ϕ (ce qui revient alors à une sorte de notation-raccourci de modus ponens). Dans l'expression de la cause en LN, L est souvent implicite ou alors verbalisée en « par définition », « par propriété », « par le lemme 1 », « par transitivité », etc.

Exemple (sans traduire le présupposé) :

(2g) Puisque f est croissante on a $f(n) > f(n0)$.
$\rightsquigarrow \text{cause}(L, \phi, \psi) \wedge \psi$, avec :
$\phi = \text{croiss}(f)$,
$\psi = f(n) > f(n0)$ et
$L = \forall f[\text{croiss}(f) \leftrightarrow \forall x, y(x > y \rightarrow f(x) > f(y))]$

Evidemment en français, on dira « par définition » pour exprimer L plutôt que d'expliciter son contenu (car L est bien la définition de $\lambda f.\text{croiss}(f)$).

Remarque : $\phi \wedge \psi \wedge \text{cause}(\phi, \psi)$ est la sémantique donnée par [2] pour la relation rhétorique *Result*. On devrait pouvoir adapter assez facilement l'existent de la SDRT en se donnant *Result*/3.

$$[\![\mathit{Result}(\lambda, \pi_1, \pi_2)]\!] = [\![K_{\pi_1} \wedge K_{\pi_2} \wedge \text{cause}(K_\lambda, K_{\pi_1}, K_{\pi_2})]\!]$$

La différence entre les RD *Result* et *Conseq* tient principalement au statut discursif (illocutoire) des termes liés par les relations. $\mathit{Conseq}(\pi_1, \pi_2)$ « fabrique » une *supposition*, viz. π_1, car la sémantique de la relation est celle de l'implication matérielle. Dans $\mathit{Result}(\_, \pi_1, \pi_2)$, π_1 n'est pas simplement supposé, il est *posé* (au moins localement), et la RD ajoute qu'il y a une relation de cause à effet de π_1 à π_2.

[1] Et en toute rigueur, il ne devrait pas porter sur les formules (ϕ, ψ) directement, mais plutôt sur les propositions associées : $\text{cause}({}^\wedge\phi, {}^\wedge\psi)$. Car 'cause' n'est pas vérifonctionnel.

[2] Si je lâche un objet au-dessus du sol, la cause particulière de sa chute est le fait que je l'ai lâché et la cause générale est la loi de la gravité

Présupposé. *Comme* et *puisque* présupposent le premier terme de la relation, contrairement à *car* ou *donc*. Le présupposé fonctionne comme une évocation en arrière-plan de ce qui est déjà connu et n'intervient pas dans le contenu sémantique stricto sensu[3]. Suivant [5], la SDRT ([1, 2]) traite les présuppositions comme des anaphores propositionnelles, à lier rhétoriquement dans le contexte.

Voici plus ou moins comment serait traitée la phrase (2g) par [1, 2] :

(2g) Puisque f est croissante on a $f(n) > f(n0)$.

$$\text{psp} : \begin{array}{|l|}\hline \pi_1\ x\ R \\ \hline \pi_1 : \begin{array}{|l|}\hline u \\ \hline u{::}\texttt{"f"} \\ \text{croiss}(u) \\ \hline\end{array} \\ R=? \qquad x=? \\ R(x,\pi_1) \\ \hline\end{array} \qquad \text{pos} : \begin{array}{|l|}\hline \pi_2 \\ \hline \pi_2 : \begin{array}{|l|}\hline \ldots \\ \hline \text{“}f(n) > f(n0)\text{”} \\ \hline\end{array} \\ \mathit{Result}(\_,\pi_1,\pi_2) \\ \hline\end{array}$$

La véritable contribution sémantique de (2g) est la boîte de droite (posée, ie assertée)[4] ; c'est la partie qui compte dans l'avancement de la preuve. La boîte de gauche (présupposé) doit être liée (ie intégrée) au contexte du discours ; c'est ce qu'indique les équations « anaphoriques » non résolues $R =?$ et $x =?$. Ce liage garantit l'indispensable satisfaction sémantique des présuppositions.

Or il se trouve que PhoX tient compte de ce procédé et cette structuration de l'information :

(2g) Puisque f est croissante on a $f(n) > f(n0)$.
`by H1 deduce f n > f n0.`

`H1` étiquette la présupposition qui va être vérifiée dans la base du prover (du moins c'est ce que je présume naïvement)[5]. Et on a une analogie frappante (mais en fait normale) entre la structure de la commande et la structure prévue par la SDRT, à savoir la SDRS « pos. » ci-dessus :

$\pi_1 \sim$ `H1`,
$\pi_2 \sim$ `f n > f n0` et
$\mathit{Result}(\ldots) \sim$ `by ...deduce ...`

Récapitulons le processus. Sous PhoX :

1. une proposition, eg une hypothèse, ϕ est donnée au prover, qui lui colle l'étiquette `H1`.
2. plus tard vient l'énoncé « puisque ϕ, ψ » et le prouveur doit reconnaître que ϕ c'est `H1` et tirer (déduire) ψ de `H1` (en fait son contenu, ie ϕ) (= `by H1 deduce` ψ).

En SDRT :

1. une phrase ϕ du texte est intégrée à la structure de discours sous forme d'une SDRS étiquetée : $\pi_0 : \phi$
2. plus tard vient la phrase « puisque ϕ, ψ » qui est analysée en deux portions : $\pi_1 : \phi$ et $\pi_2 : \psi$ (en simplifiant)
3. on s'assure que le contenu de la présupposition π_1 (ϕ) est satisfiable dans le contexte ; c'est le cas grâce à π_0 qui est déjà là et qui dit la même chose que π_1.

[3]Cf. eg [4, 5].

[4]En SDRT orthodoxe, l'analyse serait peut-être plus complexe, avec éventuellement une copie du contenu de π_1 dans la SDRS de doite. Mais c'est difficile d'en être sûr.

[5]**Question :** que se passe-t-il si un utilisateur de PhoX présuppose (linguistiquement, ie avec un *puisque*), par erreur ou à tort, une proposition qui n'a pas été démontrée auparavant et qui ne figure pas dans les connaissances préalables (définitions, lemmes) ?

4. on ajoute $\pi_2 : \psi$ et $\mathit{Result}(\_, \pi_1/\pi_0, \pi_2)$ dans la structure sémantique du discours.

L'analogie n'est donc pas fortuite.

Finalement, la partie importante du travail de l'analyseur textuel est de retrouver que la proposition « f est croissante » s'appelle bien `H1` dans le contexte, c'est-à-dire que π_1 se résout en π_0[6]. Car au niveau de la représentation sémantique, il suffit de faire cet appariement pour satisfaire la présupposition (le prover, lui, fait le boulot à fond). En pratique, c'est à cela et seulement à cela que sert la SDRS psp ci-dessus (pas besoin de l'insérer dans la structure du texte, une « copie » y figure déjà).

Proposition : je propose de modifier légèrement l'analyse de *puisque* par rapport à la version ci-dessus (2g) :

(9) « *puisque* ϕ*, (alors)* ψ » $\rightsquigarrow$
$$\begin{array}{|l|}\hline \pi_1\ \pi_2 \\ \hline \pi_2 : \psi \\ \mathit{Result}(\_, \pi_1, \pi_2) \\ \pi_1 =? \\ \hline \end{array}$$

où $\pi_1 =?$ est une pseudo-condition qui incite à retrouver le bon antécédent, c'est-à-dire l'hypothèse adéquate présente dans le contexte antérieur (et ϕ doit nous aider pour cela). On se dispense donc d'insérer ϕ en soi dans la structure du discours.

Dans une approche très PhoX-orientée, il suffit de résoudre $\pi_1 =?$ en quelque chose qui ressemble à $\pi_1 =$ `H1` (à condition que l'analyseur textuel dispose des étiquettes `H1`, `H2`, `H` etc.). Mais ça vaut le coup qu'on s'attarde un peu sur le statut sémantique de ces étiquettes. Cf. § 2.3.

Mais auparavant une remarque sur *donc*.

2.2 *Ergo*

Donc n'est pas présuppositionnel[7]. Mais à l'arrivée on pourra le faire fonctionner de manière très similaire :

(10) « *donc* ψ » $\rightsquigarrow$
$$\begin{array}{|l|}\hline \pi_1\ \pi_2 \\ \hline \pi_2 : \psi \\ \mathit{Result}(\_, \pi_1, \pi_2) \\ \pi_1 =? \\ \hline \end{array}$$

La différence avec *puisque*, c'est quon ne présuppose pas ici une proposition ϕ puisqu'on n'a pas une telle proposition dans l'input. Ici $\pi_1 =?$ est une condition standard de la SDRT qui dit simplement qu'il faut trouver un site d'accueil à cette boîte dans la structure discursive du contexte. Résoudre $\pi_1 =?$, c'est résoudre l'attachement rhétorique de π_2 (via *Result*).

2.3 Nommage de propositions

En maths, des propositions peuvent avoir un nom (nom propre ou surnom temporaire dans une preuve). Exemples : `H1`, `Lemme1`, `E_compact`, etc.

Ce n'est pas commun linguistiquement, et pas vraiment prévu par les théories sémantiques. En SDRT on peut intégrer cela, mais il ne faut pas confondre ces noms avec les étiquettes π qui sont des *tokens* d'actes de langage ; ils n'ont de valeur qu'au sein de l'énonciation d'un discours, alors qu'une proposition (qui est un sens) vaut ce qu'elle

[6] Bien entendu, pour une preuve directement rédigée en new_command dans PhoX, le pb ne se pose pas.

[7] En tout cas, pas comme *puisque*

vaut « éternellement ». Ainsi pour une proposition ϕ, on pourra avoir dans un discours donné $\pi_5 : \phi$, $\pi_{11} : \phi$, $\pi_{26} : \phi$ (si ϕ est répétée).

Je propose donc d'introduire un nouvel élément de notation dans les SDRS. Si une proposition (phrase) ϕ du texte de preuve est nommée par exemple `P`, on note dans la SDRS :

$$\begin{array}{|l|}\hline \pi_i \\ \hline \pi_i : \phi \\ \pi_i :: \texttt{"P"} \\ \hline \end{array}$$

En considérant que :: s'interprète comme une fonction $(::(\pi_i) = \texttt{"P"})$[8]. C'est donc au système de s'assurer que deux propositions différentes ne portent pas le même nom, si besoin est. On donc supposer raisonnablement qu'à partir d'un nom (`"P"`) on peut retrouver la proposition (ϕ) qu'il nomme en passant par le token d'acte de langage (π_i) qui a énoncé cette proposition.

Par conséquent, l'équation présuppositionnelle en suspend dans l'entrée sémantique de *puisque* en (9) pourra se résoudre avantageusement ainsi :

(2g) « *Puisque f est croissante on a $f(n) > f(n0)$* »

$$\rightsquigarrow \begin{array}{|l|}\hline \pi_1\ \pi_2 \\ \hline \pi_2 : \boxed{\ \ldots\ } \\ Result(\_, \pi_1, \pi_2) \\ \pi_1 =? \\ \hline \end{array} \longrightarrow \begin{array}{|l|}\hline \pi_1\ \pi_2 \\ \hline \pi_2 : \boxed{\ \ldots\ } \\ Result(\_, \pi_1, \pi_2) \\ \pi_1 :: \texttt{"H1"} \\ \hline \end{array}$$

Ce sera suffisant, sachant qu'il existe déjà dans le contexte un π_i tel que $\pi_i :: \texttt{"H1"}$, l'antécédent attendu (ou alors dans le cas de $\pi_1 :: \texttt{"lemme2"}$, on sait qu'il y a une proposition qui porte directement ce nom dans la base chargée).

Remarque Cette convention de nommage est identique au nommage des objets dénotés pas des GN. C'est délibéré, car les propositions en question sont souvent évoquées via des GN (*le lemme 2*, *l'hypothèse*, mais aussi dans des tournures comme *par définition*).

(11) *... le lemme successeur...* $\rightsquigarrow \lambda P \begin{array}{|l|}\hline u \\ \hline u :: \texttt{"successeur"} \\ \text{lemme}(u) \\ P(u) \\ \hline \end{array}$

comme pour tout GN nom propre, ex :

(12) *... la fonction f_0...* $\rightsquigarrow \lambda P \begin{array}{|l|}\hline u \\ \hline u :: \texttt{"f0"} \\ \text{fonction}(u) \\ P(u) \\ \hline \end{array}$

2.4 *D'après*

Dans le cas des textes de preuve, *d'après N* nous donne un bel exemple d'interface (interférence ?) syntaxe–pragmatique.

En effet, sémantiquement, « d'après N, ψ » devra fonctionner comme « puisque ϕ, ψ », sauf que N est un GN qui nomme (et réfère à) une proposition alors que ϕ est la proposition en soi, explicitée. Remarquons cependant que les GN définis sont traditionnellement analysés comme présuppositionnels.

[8]En fait, en faisant cela, je nomme l'acte de langage (π_i) et pas directement la proposition. Mais cette solution peut être suffisante pour les besoins de l'application.

Mais l'avantage du nom N c'est qu'il donne directement la solution de l'équation présuppositionnelle de *puisque*.

(13) *D'après le lemme 3, on a* $\psi \rightsquigarrow$
$$\begin{array}{|l|}\hline \pi_1\ \pi_2 \\ \hline \pi_1\text{::}\texttt{"lemme3"} \\ \pi_2 : \boxed{\phi} \\ \mathit{Result}(\_, \pi_1, \pi_2) \\ (\text{ou } \mathit{Result}(\pi_1, \_, \pi_2)\,?) \\ \hline \end{array}$$

Bien entendu, ce qui n'est pas simple c'est la question du mapping systématique entre les relations rhétoriques ou sémantiques et les commandes de new_command.

2.5 *Montrons*

C'est une annonce de (sous-)preuve. On peut rendre compte de cet effet d'annonce de la manière suivante en SDRT :

(14) Montrons que v converge vers l.
$$\begin{array}{|l|}\hline \pi_1\ x \\ \hline \pi_1 : \begin{array}{|l|}\hline u_1\ u_2 \\ \hline u_1\text{::}\texttt{"v"} \\ u_2\text{::}\texttt{"l"} \\ \text{converger}(u_1, u_2) \\ \hline \end{array} \\ \mathit{Elaboration}(\pi_1, x) \qquad x = ? \\ \hline \end{array} \quad \text{ou} \quad \lambda x \begin{array}{|l|}\hline \pi_1 \\ \hline \pi_1 : \begin{array}{|l|}\hline u_1\ u_2 \\ \hline u_1\text{::}\texttt{"v"} \\ u_2\text{::}\texttt{"l"} \\ \text{converger}(u_1, u_2) \\ \hline \end{array} \\ \mathit{Elaboration}(\pi_1, x) \\ \hline \end{array}$$

A cause de la non-saturation de la structure, l'énoncé n'est pas une assertion : il dénote l'ensemble des propositions qui élaborent la première proposition. Assez proche de ce que l'on cherche. Car finalement montrer ϕ ce n'est pas seulement dire que ϕ est vraie (car il suffirait alors d'asserter ϕ), c'est surtout « re-dire » ϕ *autrement*. Et la structure (14) dénote un ensemble de π, c'est-à-dire de discours.

Il ne s'agit pas là exactement de *Elaboration* avec la sémantique de [2]. Il me semble que dans les textes de preuve, l'élaboration, qui est de la reformulation, s'appuie sur une sémantique d'équivalence logique (à vérifier).

Variantes L'expression « Montrons que ϕ » est un impératif, donc ce n'est pas une véritable assertion. Mais des variantes d'annonces sont des phrases grammaticalement déclaratives :

(2) b. On cherche $n1$ tq pour tout $n > n1$ $d(v_n, l) << e$.

(3) b. ... montrons que $(Vn)n$ converge vers l, c'est-à-dire : on cherche $n1$ tel que pour tout $n > n1$ $d(Vn, l) < e$.

e. Mais maintenant, si $f(n) > n0$, comme $Vn = Uf(n)$, alors on aura démontré que $(Vn)n$ converge vers l.

Mais, bien que des phrases déclaratives, ces énoncés sont, pragmatiquement, difficilement falsifiables (niables), et on peut les voir comme des actes de langages indirects, à savoir des assertions de surfaces réinterprétées comme des (auto-)injonctions.

Littéralement, « on cherche N tq $\phi(N)$ » donnera :

(15)
$$\begin{array}{|l|}\hline o \\ \hline \text{chercher}(o, {}^\wedge \begin{array}{|l|}\hline u \\ \hline u\text{::}\texttt{"N"} \\ \phi(u) \\ \hline \end{array}) \\ \hline \end{array}$$
(o = constante = *on*, *nous*, locuteur)

Mais cela revient à une analyse cataphorique de « montrons/on va montrer qu'il existe N tq $\phi(N)$ » :

(16)
$$\begin{array}{|l|}\hline \pi_1\ x \\ \hline \pi_1 : \begin{array}{|l|}\hline u \\ \hline u::\texttt{"N"} \\ \phi(u) \\ \hline \end{array} \\ \mathit{Elaboration}(\pi_1, x) \\ x = ? \\ \hline \end{array}$$

La SDRT permet de représenter des performatifs, sans risquer des paradoxes sémantiques. Remarquons que dans « Montrons que ϕ », le locuteur s'engage – par anticipation – sur la vérité de ϕ. Il y a donc une certaine cohérence à utiliser *Elaboration* (qui est une relation qui pose la vérité de ses termes). Ainsi on peut proposer :

(17) Montrons que $\phi \rightsquigarrow$
$$\begin{array}{|l|}\hline \pi_1\ x \\ \hline \pi_1 : \boxed{\phi} \\ \mathit{Elaboration}(\pi_1, x) \\ x = ? \\ \hline \end{array}$$

3 Macro-structures des preuves

(18) Soit u une suite de E convergeant vers l et v une suite extraite de u.

$$\begin{array}{|l|}\hline \pi_0 \\ \hline \pi_0 : \begin{array}{|l|}\hline \pi_1\ x_1 \\ \hline \pi_1 : \begin{array}{|ll|}\hline u_1\ u_2\ u_3\ u_4 & \\ \hline u_1::\texttt{"u"} & \text{suite}(u_1) \\ u_2::\texttt{"E"} & \text{de ?}(u_1, u_2) \\ u_3::\texttt{"l"} & \text{conv.}(u_1, u_3) \\ u_4::\texttt{"v"} & \text{s-extraite}(u_4, u_1) \\ \hline \end{array} \\ \pi_1::\texttt{"H"} \\ \mathit{Conseq}(\pi_1, x_1) \qquad x_1 = ? \\ \hline \end{array} \\ \hline \end{array}$$

Rq : cette DRS donne : $\forall u, E, l, v \ldots$ Tout est quantifié universellement.

Il faudrait peut-être couper π_1 en 2 pour pouvoir nommer deux hypothèses distinctes. Cela se fait facilement en embriquant deux *Conseq* à la suite.

(19) Montrons que v converge aussi vers l.

$$\begin{array}{|l|}\hline \pi_2^* \\ \hline \pi_2^* : \begin{array}{|l|}\hline \pi_2\ x_2 \\ \hline \pi_2 : \begin{array}{|ll|}\hline u_5\ u_6 & \\ \hline u_4::\texttt{"v"} & \text{conv.}(u_5, u_6) \\ u_6::\texttt{"l"} & \\ \hline \end{array} \\ \mathit{Elaboration}(\pi_2, x_2) \qquad x_2 = ? \\ \hline \end{array} \\ \hline \end{array}$$

Avec $x_1 = \pi_2^*$ ou π_2

(20) Soit e > 0, montrons qu'il existe n0 tel que pour tout n > n0 d (v(n), l) << e.

π_3^*

π_3^* :
- $\pi_3\ \pi_4^*$
  - π_3 : [$u_7\ u_8$ | u_7::`"e"` u_8::`"0"` ; $<(u_7, u_8)$]
  - π_4^* : [$\pi_4\ x_4$ |
    - π_4 : [u_9 | u_9::`"n0"` ;
      [u_{10} | u_{10}::`"n"` ; $>(u_{10}, u_9)$] $\Rightarrow$ [... | ... ; "d(v(u_{10}),l) << e"]]
    - *Elaboration*(π_4, x_4) $x_4 =?$]
  - *Conseq*(π_3, π_4^*)

Avec (sûrement) $x_2 = \pi_3^*$.

(21) Comme v est une suite extraite de u, il existe une application f croissante et pour tout n, v(n) = u(f(n)).

π_5^*

π_5^* :
- $\pi_5\ \pi_6$
  - π_5 : [u_{15} | u_{15}::`"f"` appli(u_{15}) ; croiss(u_{15}) ;
    [u_{16} | u_{16}::`"n"`] $\Rightarrow$ [... | "v(n) = u(f(n))"]]
  - π_5::`"H2"`
  - *Result*$(\_, \pi_6, \pi_5)$ $\pi_6 =?/$::`"H"`

Avec $x_4 = \pi_5^*$. De plus, on attend la résolution $\pi_6 =?$ en π_6::`"H"`.

Ici, pour coller à PhoX, ça bloque un peu car π_5 n'est pas tant une assertion que la pose d'une hypothèse. Il faudrait donc avoir aussi sous π_5^* une relation *Conseq*(π_5, x_5). Peut-on faire cette prédiction à tous les coups ? Est-ce l'énoncé existentiel qui l'induit ?

(22) L'application f étant croissante, on a : f(n) >=n pour tout n.

π_7^*

π_7^* :
- $\pi_7\ \pi_8$
  - π_7 : [... | "f(n) >=n pour tout n"]
  - *Result*$(\_, \pi_8, \pi_7)$ $\pi_8 =?/$::`"H2"`

Là, si on ne peut pas faire $x_5 = \pi_7^*$, on ne sait pas trop à quoi rattacher π_7^* et c'est au mécanisme d'update général de la SDRT de répondre.

Rq : si au lieu de résoudre π_8::`"H2"` on résoud $\pi_8 = \pi_5$, on a un rattachement rhétorique plutôt convenable, mais avec *Result* au lieu de *Conseq* (PhoX attendrait *Conseq* pour faire le `assume` de (21)).

L'étudiant propose qqch qui aboutit à $Result(\pi_8', \pi_8, \pi_7)$ avec π_8'::`"lemme1"`.

(23) a. La suite u est une suite qui converge vers l.

b. Donc pour e, il existe n0 tel que pour tout n > n0 , (d (u n) l) << e.

π_9 π_{10}

π_9 : [u_{20} u_{21} | u_{20}::`"u"` suite(u_{20}) ; u_{21}::`"l"` conv.(u_{20}, u_{21}) ; u_{20} =?]

π_9::`"H0"`

π_{10} : [| 'il existe n0 tq...']

$Result(\_, \pi_9, \pi_{10})$

(23a) est une rappel de l'hypothèse `H` (ou `H0`). On se retrouve dans une configuration logique de la forme $H \rightarrow (\ldots \rightarrow (\ldots \rightarrow (H \ldots)))$, ce qui est valide. D'après les commandes de la preuve sous PhoX, cette phrase ne devrait pas être traduite en soi, mais traitée comme une présupposition (et NB : le GN sujet [la suite u] est présuppositionnel). Pourtant, il est vrai que de tels énoncés de rappel ne sont pas marginaux ni inutiles pour le texte en LN. La solution pour l'interface avec le prouveur est de reconnaître cette hypothèse (ie retrouver son nom, `"H0"`). On remarque aussi que le premier argument de *Result* (ici _) devrait être la définition de la convergence de suite.

Dans (23b), l'analyse sémantique de « pour e » n'est pas triviale[9]. En fait e est un paramètre donné au _ de $Result(\_, \pi_9, \pi_{10})$, pour valider la déduction de π_{10}.

(24) a. Soit n, on a : f(n)>= n> n0, donc d(u(f(n)), l)<<e.

b. On a aussi v(n) = u(f(n)) donc d(v(n),l)<<e.

Ici l'analyse est assez similaire aux segments vus précédemments (avec des *Conseq* (soit) et des *Result* (donc)). Là encore on n'a pas d'indice de rattachement rhétorique du bloc.

(25) On a donc montré que pour tout e >0, il existe n0 tel que pour tout n>n0, d(v(n),l)<<e, c'est-à-dire v(n) converge vers l. [Cqfd.]

(25) a un statut de conclusion (à tiroir). Et on ne sait pas exactement à quel site de la structure de discours rattacher ce bloc. C'est une conclusion de toute la preuve, donc on pourrait (ou devrait) remonter très haut (eg π_0). Une analyse plausible pourra être $Result(\pi_0, \pi_{(25)})$. L'interprétation d'une SDRS se fait en profondeur d'abord, donc la preuve sera lue en entier avant sa conclusion.

π_0^*

π_0^* : [π_0 $\pi_{(25)}$ | π_0 : [...] ; $\pi_{(25)}$: [conclusion/récap] ; $Result(\pi_0, \pi_{(25)})$]

Grammaticalement, une conclusion peut se repérer par l'usage du passé composé, plutôt rare dans les textes de preuve, sauf pour les récapitulations.

[9]Même si on voit bien ce que ça doit donner dans la preuve.

Références

[1] Nicholas Asher and Alex Lascarides. The semantics and pragmatics of presupposition. *Journal of Semantics*, 15(3):239–300, 1998.

[2] Nicholas Asher and Alex Lascarides. *Logics of Conversation*. Cambridge University Press, Cambridge, 2003.

[3] Laurent Roussarie. Structure sémantique des textes mathématiques – le cas des définitions. Technical report, DemoNat, novembre 2004.

[4] Robert C. Stalnaker. Assertion. In P. Cole, editor, *Pragmatics*, volume 9 of *Syntax and Semantics*, pages 315–332. Academic Press, New York, 1978.

[5] Rob A. van der Sandt. Presupposition projection as anaphora resolution. *Journal of Semantics*, 9(4):333–377, 1992.