

La modélisation de l'apprenant dans un EIAH

Hamid Chaachoua,

IUFM de Grenoble, Equipe MetAH, Laboratoire Leibniz.

Introduction

La modélisation des connaissances de l'élève est depuis longtemps une composante majeure des travaux de la communauté AI-ED [Wenger 1987]. Elle a aussi pris une place centrale dans différentes théories de didactiques, d'abord comme cadre de recherche [Vergnaud 1991, Balacheff 1995], et ensuite comme outil pour décrire l'état des connaissances des élèves à propos d'un concept [Artigue et Robinet 1982] ou pour prendre des décisions didactiques dans les situations d'apprentissages [Tahri 1993, Chaachoua et Lima 2003].

Pour modéliser les connaissances de l'apprenant, nous nous sommes référés à la théorie du champs conceptuel de Vergnaud (1991). Le point de départ de cette théorie est que les conduites des élèves (hésitations, erreurs, décisions...) dans des situations de résolution de problèmes sont structurées par des schèmes. L'auteur définit le «schème» par "l'organisation invariante de la conduite pour une classe de situations donné. *C'est dans les schèmes qu'il faut rechercher les connaissances-en-acte du sujet, c'est-à-dire les éléments cognitifs qui permettent à l'action du sujet d'être opératoire*" (ibid, p.136). Un schème repose sur :

- un ensemble d'invariants opératoires (concepts-en-acte et théorèmes-en-acte),
- des anticipations du but à atteindre,
- des règles d'actions qui permettent de générer les actions du sujet,
- des inférences ou des raisonnements qui permettent de calculer les règles d'actions et donc de mettre en œuvre le schème dans chaque situation particulière.

Un théorème-en-acte est un invariant de type proposition, il peut être vrai ou faux. *"Les concepts se développent dans l'action et sous tendent les formes d'organisation de l'activité que sont les schèmes. Il n'y a pas d'action possible sans propositions tenues pour vraies sur le réel. Ce sont justement ces propositions tenues pour vraies que j'appelle théorèmes-en-acte, y compris pour d'autres domaines d'activité que les mathématiques.*

Leur portée est souvent locale (elle l'est toujours dans la phase d'émergence); ils peuvent rester implicites; ils peuvent même être faux" (Vergnaud 2001).

Par exemple¹, un schème de résolution des équations de degré 1 de la forme $ax+b=c$ repose sur des théorèmes-en-acte comme "on conserve l'égalité en soustrayant b des deux côtés" et des règles d'actions comme "si $a+b=c$ alors $a+b-b=c-b$ ".

Dans cette recherche nous proposons de déterminer automatiquement les règles d'actions et les théorèmes-en-actes relatifs à une action que nous appelons "mouvement dans les équations". La détermination automatique dépend également de la modélisation informatique de l'apprenant.

Depuis janvier 2003, nous avons engagé un travail de modélisation de comportements des élèves en algèbre, dans le cadre d'un projet financé par le ministère de la recherche². Nous présentons, dans cet article, le modèle et les mécanismes que nous avons développés pour les équations et inéquations de degré 1, ainsi que leur application sur des productions d'élèves faites avec un micromonde d'algèbre Aplusix³ (cf. annexe A) en France et au Brésil. Le diagnostic automatique a été fait par un logiciel Anaïs développé à cet effet.

La modélisation d'un apprenant dans un EIAH

Balacheff (1994) distingue trois univers relatifs à un dispositif informatique : l'univers interne de la machine, l'interface comme lieu de communication entre l'utilisateur et le dispositif informatique et enfin l'univers externe, dans lequel on trouve l'utilisateur. La modélisation informatique de l'apprenant est implémenté au niveau de l'univers interne. On distingue deux types de modélisations : le modèle comportemental qui consiste à représenter des observables extérieurs et le modèle épistémique qui est une interprétation de ce qui se produit au niveau comportemental.

Le modèle comportemental est obtenu à partir de l'ensemble des événements en ignorant des événements non pertinents et en remplaçant certaines séquences d'événements par des « *descripteurs de plus haut niveau déterminé par le langage de commande du système* » (Balacheff, 1994). Si ce traitement permet d'obtenir une description des observables qui

¹ Tiré de (Vergnaud, 1991, p. 137)

² ACI « Ecole et Sciences Cognitives ». Le rapport peut être consulté à l'adresse : <http://www-leibniz.imag.fr/NEWLEIBNIZ/LesCahiers/2005/Cahier123/ResumCahier123.html>

³ <http://applusix.imag.fr/>

doit permettre de rejouer la session de l'apprenant de façon fidèle au regard de la problématique dans laquelle on se place, alors on dit qu'il y a homomorphisme de comportement.

Le modèle épistémique est obtenu à partir d'un traitement du modèle comportemental. C'est une reconstruction à partir des observables du processus de résolution de problème, ou la caractérisation de l'état de connaissance de l'élève. « *La relation qu'entretient le modèle épistémique avec le modèle (conception) élaboré par le didacticien constitue un épistémorphisme si elle permet de mettre en évidence que le modèle épistémique rend compte des propriétés structurelles et conceptuelles décrites par le modèle de conception attribué à l'élève* » (Balacheff, 1994).

La qualité didactique d'un diagnostic obtenu par EIAH dépend de ces deux modélisations. Le travail présenté ci-dessous est le fruit d'une collaboration étroite entre informaticiens et didacticiens pour réaliser le modèle comportemental et le modèle épistémique. La validation de ces modèles a été faite en confrontant le diagnostic automatique et le diagnostic manuel.

Méthodologie

Cette recherche est le fruit d'un travail collaboratif entre didacticiens⁴ et informaticien⁵. Au début de ce travail, nous avons construit *a priori* des théorèmes-en-actes possibles d'élèves sur un concept, en associant des comportements possibles de l'élève aux différents contextes dans lesquels ces comportements se produisent. Ensuite, nous avons mis en œuvre ce modèle de façon automatique pour voir s'il produisait une discrimination significative entre les élèves.

Les données ont été recueillies avec le logiciel Aplusix. Chaque action de l'élève est enregistrée dans un fichier "protocoles brut". Le logiciel Aplusix permet de voir en différé la forme finale des exercices; son magnétoscope permet d'observer le détail des actions. Un autre fichier est constitué, à partir du fichier "protocoles brut", sur lequel se base le logiciel de diagnostic Anais. Ce fichier ne conserve que les étapes validées par l'élève, les

•
⁴ M. Bittar (Universidade Federal de Mato Grosso do Sul, Campo Grande, Brésil) et moi même

⁵ J.F. Nicaud (Equipe Metah, Laboratoire Leibniz)

corrections, les hésitations et le temps ne sont pas pris en compte. Nous avons travaillé sur ce fichier, que nous appelons "protocole".

L'analyse manuelle est faite à partir du magnétoscope et du fichier "protocole" pour identifier les transformations erronées (A), ensuite on a progressé le long d'une spirale composée du codage des règles en langage informatique (B) et de leur analyse avec affinage du diagnostic automatique (C).

La méthodologie adoptée peut être résumée par la représentation schématique ci-dessous, extraite de (M.C. Croset, 2005).

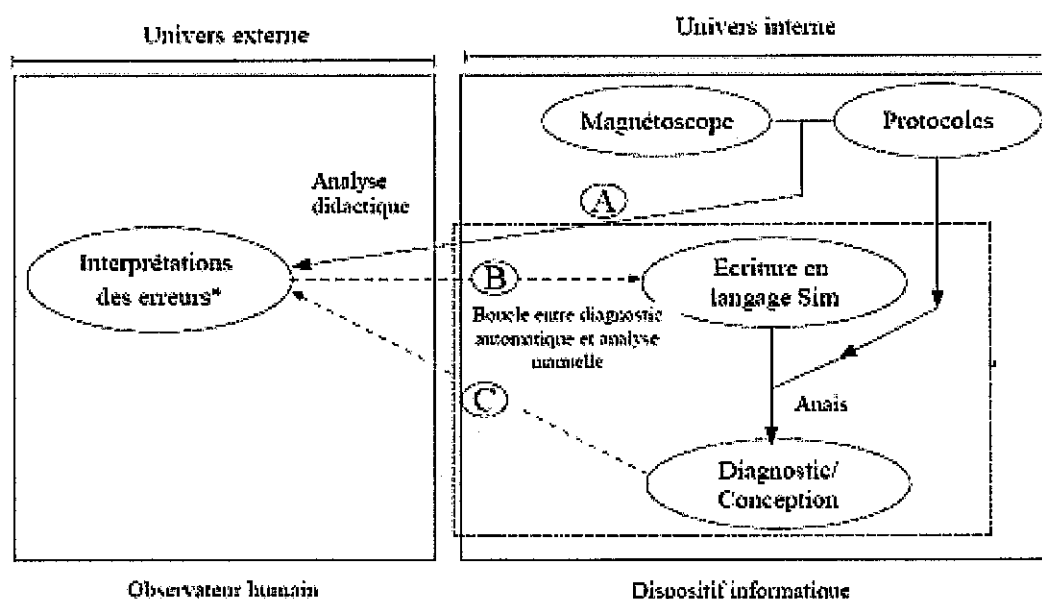


Figure 1

Une modélisation par des règles de réécriture

Principes généraux

Une très grande partie des calculs d'algèbre formelle s'effectuent en appliquant des règles de réécriture, selon le principe du remplacement d'égaux [Dershowitz et Jouannaud 1989]. Une règle de réécriture R est un objet de la forme $A \rightarrow B$. Elle est applicable à une expression E si A s'unifie à une sous-expression U de E . L'application consiste à remplacer U dans E par B . Lorsque l'on utilise un résolveur de référence, les règles sont issues d'identités fournies par des axiomes ou des théorèmes. Ainsi, l'identité $A(B+C)=AB+AC$ produit-elle une règle de développement $A(B+C) \rightarrow AB+AC$ et une règle de factorisation

$AB+AC \rightarrow A(B+C)$. Un raisonnement algébrique est principalement présenté par une succession d'étapes de calcul dont chacune est produite à partir de la précédente en appliquant plusieurs règles de réécriture. Par exemple, le passage de l'étape $2x(3x^2-4)$ à l'étape $6x^3-8x$ peut être produit par l'application de la règle de développement ci-dessus et de règles de réduction.

Lorsqu'un élève résout un exercice d'algèbre formelle, il produit aussi des étapes de calcul. Notre modèle cherche à interpréter ces productions d'étapes par l'application de règles de réécriture correctes ou erronées, à l'instar d'Anderson et al. [1990]. Par exemple, si l'élève est passé de $2x(3x^2-4)$ à $6x^3-4$, nous pouvons interpréter cette transformation par l'application de la règle erronée $A(B+C) \rightarrow AB+A$ suivie de l'application de règles de réduction correctes.

La résolution des équations et inéquations de degré 1

La stratégie principale de résolution des équations de degré 1, consiste à développer les deux membres, le cas échéant, puis à isoler la variable en utilisant des théorèmes permettant d'effectuer des opérations sur les deux membres :

On ne change pas les solutions d'une équation en ajoutant ou en retranchant un même nombre aux deux membres de l'équation.

On ne change pas les solutions d'une équation en multipliant ou en divisant par un même nombre non nul les deux membres de l'équation.

Ils se traduisent par les règles d'« opérations sur les deux membres » suivantes :

Règle 1: $A=B \rightarrow A+C=B+C$

Règle 2: $A=B \rightarrow A-C=B-C$

Règle 3: $A=B \rightarrow AC=BC \ (C \neq 0)$

Règle 4: $A=B \rightarrow A/C=B/C \ (C \neq 0)$

L'application de ces règles se combine à des réductions, ce qui donne naissance à des règles compilées plus efficaces [Anderson 1983]. La règle 1 produit la règle 5 : $A+C=B \rightarrow A=B-C$ et la règle 6 : $C=B \rightarrow 0=B-C$. Dans ces règles, C est enlevé d'un membre pour être placé dans l'autre. Nous parlons de « mouvement », plus précisément de mouvement additif ici. A côté de ces deux règles de mouvement additif de gauche vers la droite, il faut ajouter deux règles (règles 7 et 8) de mouvement additif de droite vers la gauche.

De façon analogue, la règle 4 produit la règle 9 : $AC=B \rightarrow A=B/C \ (C \neq 0)$ et la règle 10 : $C=B \rightarrow 1=B/C \ (C \neq 0)$. Nous parlons ici de mouvement multiplicatif de C. Ces règles

doivent être complétées par des règles dans lesquelles le membre de gauche est une fraction, C étant le numérateur ou un facteur du numérateur, puis par les règles semblables de droite vers la gauche.

De façon analogue, la règle 3 produit des règles de mouvement multiplicatif d'un terme pris au dénominateur.

Pour les inéquations, les règles effectuant un mouvement multiplicatif se dédoublent selon le signe de C. Par exemple, l'équivalent de la règle 9, pour la relation « < », est constitué de : $AC < B \rightarrow A < B/C \ (C > 0)$ et $AC < B \rightarrow A < B/C \ (C < 0)$.

On dénombre ainsi 20 *règles détaillées* pour les mouvements dans les équations et 40 pour chaque inéquation.

Nous avons présenté ci-dessus les règles de réécriture propres aux équations et aux inéquations de degré 1. Les autres règles (factorisations, développements, réductions, etc.) s'appliquent à des sous-expressions situées dans un membre ; elles ne sont pas propres aux équations et aux inéquations de degré 1.

Une règle unique de mouvement

Pour simplifier le discours, nous utilisons, ci-dessous, le terme « équation » à la place de « équation et inéquation de degré 1 ».

Le nombre très important des règles de mouvement conduit à s'interroger sur la pertinence du modèle introduit en 3.2. Un professeur ou un élève a-t-il toutes ces règles détaillées en tête lorsqu'il résout des équations ? Certainement pas sous cette forme. Analysons ces notions de mouvement pour en faire ressortir les principaux traits et envisageons en même temps des mouvements erronés.

Un mouvement additif porte sur une sous-expression que nous appelons argument : c'est l'objet qui passe d'un membre dans l'autre. L'argument est additif, ce qui signifie qu'il se trouve dans une somme située dans un membre de l'équation ou qu'il est lui-même un membre de l'équation. Quand le mouvement est effectué correctement, l'argument est encore additif, dans l'autre membre, et il a changé de signe syntaxique (le signe syntaxique provient de la présence ou non de « - » devant l'argument, par exemple : $-3x$ a pour signe syntaxique « - »).

Un mouvement multiplicatif a aussi un argument. L'argument peut être « multiplicatif au numérateur », ce qui signifie qu'il est un facteur d'un membre de l'équation ou du numérateur d'une fraction qui est un membre de l'équation ; l'argument peut être « multiplicatif au dénominateur », ce qui signifie qu'il est un facteur du dénominateur d'une fraction qui est un membre de l'équation. Quand le mouvement est effectué correctement, l'argument est encore multiplicatif dans l'autre membre (au dénominateur s'il vient du numérateur et au numérateur s'il vient du dénominateur), il n'a pas changé de signe syntaxique, mais, dans le cas d'une inéquation, le sens de l'inéquation a changé si le signe sémantique (nombre positif ou négatif) de l'argument est négatif.

Nous pouvons maintenant décrire les mouvements avec une seule règle, intitulée *Mouvement*, auquel on associe un vecteur de sept variables, variables dont les noms sont indiqués ci-dessous, suivis des valeurs qu'elles peuvent prendre :

Symbole de relation : parmi ($= \neq < \leq > \geq$).

Orientation horizontale du mouvement : parmi (GaucheDroite DroiteGauche).

Orientation verticale du mouvement : parmi (NumVersNum, NumVersDeno, DenoVersNum, DenoVersDeno), « Num » signifiant numérateur et « Deno » signifiant dénominateur.

Position de l'argument à l'origine

"PosOrgArgEstAdd" si la position d'origine de l'argument est additive.

"PosOrgArgEstMult" si la position d'origine de l'argument est multiplicative.

Position finale de l'argument

"PosFinaleArgEstAdd" : la position finale de l'argument est additive.

"PosFinaleArgEstMult" : la position finale de l'argument est multiplicative.

Changement de signe de l'argument

"ChangeSigneArg" : le signe de l'argument est changé lors du mouvement.

"ChangePasSigneArg" : le signe de l'argument n'est pas changé.

Changement de sens

"ChangePasSens" : le sens de l'inégalité n'est pas changé.

"ChangeSens" : le sens de l'inégalité est changé.

Par exemple, la transformation erronée $2x-4 \leq 5 \rightarrow 2x \geq 5-4$ est représentée par un *Mouvement* de -4 de vecteur : ($\leq$, GaucheDroite, NumVersNum, PosOrgArgEstAdd, PosFinaleArgEstAdd, ChangePasSigneArg, ChangeSens). La règle détaillée correspondante peut être produite à partir du vecteur, c'est : $A+C \leq B \rightarrow A \geq B+C$.

Détermination a priori des théorèmes-en-actes

Les actions de l'élève relatives au mouvement d'un argument dans une équation ou inéquation peuvent porter sur trois traits : signe de l'argument dans la position finale, opérateur associé à l'argument dans la position finale et, pour les inéquations, sens de l'inégalité. Nous désignons ces trois traits respectivement par « signe dans mouvement », « sens dans mouvement » et « opérateur dans mouvement ».

Une conception du mouvement d'un argument est caractérisée essentiellement par les invariants opératoires (TeA et règles d'action) relatifs aux traits : «signe dans mouvement, sens dans mouvement et opérateur dans mouvement ». Pour déterminer ces invariants opératoires, nous cherchons des corrélations entre les actions de l'élève et le contexte dans lequel on rencontre cette action de façon stable, le contexte étant des éléments de la situation où l'élève produit telle ou telle action. Par exemple, il est important de savoir quand est ce qu'un élève change ou non le signe de l'argument dans le mouvement.

On peut exprimer les invariants opératoires par des règles de la forme si <contexte> alors <action> : il s'agit des règles d'action. Ensuite, par un processus de diagnostic par généralisation du contexte, on construit des TeA.

Description des actions

A chaque trait sont associées les actions suivantes :

Actions associées au Signe

ChangeSigneArg : le signe de l'argument est changé lors du mouvement.

ChangePasSigneArg : le signe de l'argument n'est pas changé.

Actions associées au Sens

ChangePasSens : le sens de l'inégalité n'est pas changé.

ChangeSens : le sens de l'inégalité est changé.

Actions associées à l'Opérateur

Additif : l'argument est additif dans la position finale ($a+x$)

Multiplicatif-Numérateur : l'argument est multiplicatif au numérateur dans la position finale ($a*x$)

Multiplicatif-Dénominateur : l'argument est multiplicatif au dénominateur dans la position finale (x/a)

Ces actions relatives à l'opérateur se déduisent de la position finale de l'argument.

Description des contextes

On spécifie des éléments du contexte pouvant avoir une influence sur les actions. Cette spécification faite par l'observateur repose en partie sur une analyse épistémique et sur les résultats des travaux sur les erreurs des élèves. La difficulté est de trouver un niveau de granularité pertinent pour les interprétations. En effet, plus il est fin, mieux il permet de rendre compte du contexte, mais il risque de ne pas mettre en évidence les régularités entre le contexte et les actions. A contrario, s'il n'a pas beaucoup de trait, les régulations constatées risquent de ne pas être discriminantes pour les élèves. Le niveau de granularité est déterminé par un aller retour entre le modèle et les résultats des diagnostics.

Pour le mouvement, nous avons retenu comme éléments du contexte : type de relation, position de l'argument à l'origine, signe de l'argument, nature de l'argument et position relative de l'argument par rapport à la relation. Les valeurs prises par chacun de ces éléments du contexte sont :

Type de relation

Equation : si le mouvement se fait dans une relation ($= \neq$).

Inéquation : si le mouvement se fait dans une relation ($< \leq > \geq$).

Position de l'argument à l'origine

PosOrgArgEstAdd : si la position d'origine de l'argument est additive.

PosOrgArgEstMult-Num : si la position d'origine de l'argument est multiplicative au numérateur.

PosOrgArgEstMult-Deno : si la position d'origine de l'argument est multiplicative au dénominateur.

Les valeurs PosOrgArgEstMult-Num et PosOrgArgEstMult-Deno peuvent être remplacées par une seule valeur PosOrgArgEstMult si on pense que les positions Numérateur et Dénominateur n'ont pas d'influence sur les valeurs de l'action.

Signe de l'argument

SigneArgEstPlus : si le signe syntaxique de l'argument est positif.

SigneArgEstMoins : si le signe syntaxique de l'argument est négatif.

Nature de l'argument

ArgEstEntier : si l'argument est un entier.

ArgEstFraction : si l'argument est sous une forme de fraction rationnelle.

ArgEstDécimal : si l'argument est sous une forme d'une écriture décimale.

ArgEstRadical : si l'argument contient un radical.

Cet élément intervient dans le cas où l'argument est numérique.

Position de l'argument par rapport à la relation

PosOrgArgEstGauche : si la position d'origine de l'argument est à gauche de la relation.

PosOrgArgEstDroite : si la position d'origine de l'argument est à gauche de la relation.

Dans le cadre de cette recherche, nous avons fait le choix de négliger les deux éléments suivants du contexte : « Nature de l'argument » et « Position de l'argument relative par rapport à la relation ».

Règles d'actions

Pour chaque trait du mouvement, nous définissons un Vecteur du Comportement Local, que l'on note VCLP(trait), à partir des éléments du contexte qui nous semblent pertinents pour le trait et des actions associées au trait. Le vecteur obtenu permet de déterminer les règles d'actions relatives au trait. Ce vecteur propose un découpage et une réorganisation des faits relevés par un observateur, il s'agit d'un niveau comportemental de la modélisation de l'élève (Wenger 1987, Dillenbourg et Self 1992). Le choix des événements qui doivent être pris en compte pour ce niveau est le résultat des décisions de l'observateur, comme le souligne (Balacheff 1994, p. 26) « la modélisation comportementale exige donc un premier niveau d'interprétation, celui de l'organisation du réel ».

Nous présentons ci-dessous les VCLP (trait) pour les trois traits du mouvement. Ces vecteurs sont décrits par (Nom du vecteur, Eléments du contexte, Actions).

Signe

VCLP(signe) = (VCLP-Signe-X, Type de relation, Position de l'argument à l'origine, Signe de l'argument, Action sur le signe). On obtient 16 valeurs possibles de ce vecteur dont 8 expriment des règles correctes.

Sens

VCLP(sens) = (VCLP-Sens-X, inéquation, Position de l'argument à l'origine, Signe de l'argument, Action sur le sens). On obtient 8 valeurs possibles de ce vecteur dont 4 expriment des règles correctes.

Opérateur

VCLP(opérateur) = (VCLP-Opérateur-X, Type de relation, Position de l'argument à l'origine, Signe de l'argument, Position finale de l'argument). On obtient 18 valeurs possibles de ce vecteur, dont 6 expriment des règles correctes.

Par exemple, le vecteur (VCLP-Signe-1a equation PosOrgArgEstAdd SigneArgEstPlus ChangePasSigneArg) peut être interprété par la règle d'action : « Si l'argument à déplacer, d'un membre d'une équation à l'autre membre, est en position additive et de signe positif, alors on ne change pas de signe après le mouvement », ou encore « si $a+b=c$ alors $b=c \square a$ ou $b=a \square c$ où $\square$ désigne un opérateur parmi +, *, / ».

Les VCLP sont diagnostiqués quand il y a une corrélation entre le contexte et l'action selon, un mécanisme décrit dans le paragraphe 6.

Théorèmes-en-acte (TeA)

Pour chaque trait du mouvement, nous avons défini des TeA globaux C_i (cf. annexe B), puis des TeA $C_i-x-y-z$ sur des domaines plus restreints. x , y et z précisent le contexte parmi les valeurs suivantes :

- type de relation : Eq (pour équation) ou Ineq (pour inéquation) ou vide
- position d'origine de l'argument : Add (pour additif), Mult (pour multiplicatif), Num (pour numérateur) ou Deno (pour dénominateur);
- signe de l'argument : plus ou moins

Ces TeA sont organisés selon une hiérarchie relative au contexte. Si l'on choisit un ordre de restriction des contextes, on peut organiser ces TeA selon une hiérarchie dont les racines sont des *TeA globaux*, chacun ayant des descendants, comme indiqué sur les treillis en annexes C.

Nous définissons le niveau de profondeur du contexte par « profondeur 0 » si le TeA diagnostiqué est global (C_i), « profondeur 1 » si le TeA diagnostiqué est (C_i-x), « profondeur 2 » s'il est (C_i-x-y) et ainsi de suite.

Le niveau le plus profond est déterminé à partir des VCLP diagnostiqués. Ce sont les TeA de niveau le moins profond qui sont attribués à l'élève, car ce sont ceux pour lesquels le contexte est le plus large.

Un TeA est correct s'il est diagnostiqué directement à partir des VCLP corrects ou si tous ses descendants sont corrects. Dans ce cas, son domaine de validité est caractérisé par le contexte dans lequel il est défini (cf. les treillis en annexes C) : les TeA sont représentés en jaune).

Dans les autres cas, le domaine de validité du TeA est celui du TeA correct qui est son plus proche descendant.

Par exemple : le TeA « conservation de l'opérateur » (CO), qui consiste à reporter le même opérateur dans l'autre membre, n'est pas correct. Son domaine de validité est le contexte additif.

Un mécanisme de détermination des théorèmes-en-acte

Dans ce paragraphe, et pour des raisons de simplifications, nous nous limitons au trait "changement de signe". Le diagnostic se fait en 4 étapes :

Etape 1 : Diagnostic local.

Cette étape consiste, pour chaque transformation $A \rightarrow B$ effectuée par l'élève, à rechercher une suite de règles plausibles dont l'application fait passer de A à B (pendant l'utilisation d'Aplusix, les transformations sont matérialisées comme indiqué sur la figure 2). Pour cela, une base de règles correctes et erronées a été mise en place (contenant la règle unique de mouvement, cf. 4.3) et un interpréteur de règles a été réalisé. Le diagnostic lui-même s'effectue à l'aide d'un logiciel Anaïs, selon une technique de recherche heuristique [Pearl 1984], en développant un arbre d'expressions pouvant être produites à partir de A jusqu'à ce que l'on atteigne B ; à chaque instant, le choix du nœud sur lequel on applique toutes les règles possibles étant réalisé par une heuristique.

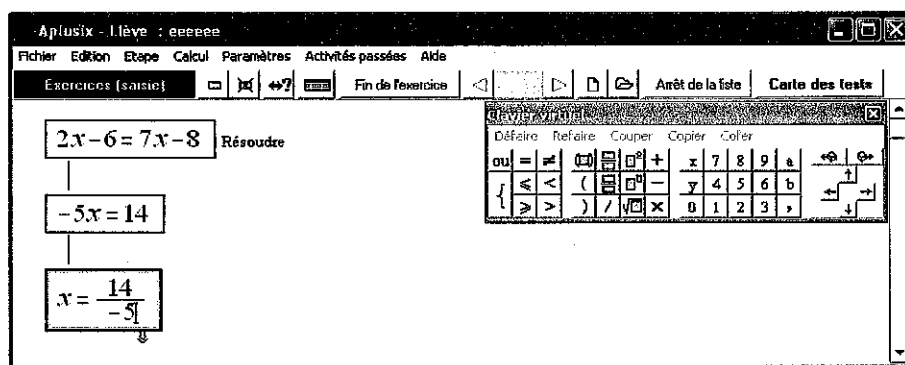


Figure 2. Copie d'écran d'Aplusix. L'élève résout une équation en plusieurs étapes.

Par exemple, la transformation $2x-6=7x-8 \rightarrow -5x=-14$ est diagnostiquée par l'application de quatre règles :

Mouvement additif de -6 sans changer de signe (erroné)

Mouvement additif de $7x$ en changeant de signe (correct)

Groupement additif $2x-7x \rightarrow -5x$ (correct)

Somme de 2 entiers $-8-6 \rightarrow -14$ (correct)

Etape 2 : Calcul des occurrences des contextes.

On calcule de façon automatique par le logiciel Anaïs qui a pris chaque règle de mouvement apparaissant dans un diagnostic local et l'a comptabilisée dans le contexte qui lui est adapté.

Etape 3 : Détermination automatique des VCLP.

On détermine tous les contextes et, pour chacun d'eux, par exemple (equation, PosOrgArgEstAdd, SigneArgEstPlus), on dénombre les VCLP qui entrent dans le contexte. On obtient n VCLP dont n_1 avec ChangePasSigneArg et n_2 avec ChangeSigneArg.

Nous attribuons trois statuts aux données par rapport à un contexte :

- insuffisantes si $n \leq 3$,
- suffisantes-instables si $n \geq 4$ et ($n_1/n \leq 0.75$ et $n_2/n \leq 0.75$),
- suffisantes-stables si $n \geq 4$ et ($n_1/n \geq 0.75$ ou $n_2/n \geq 0.75$).

Dans le premier cas, nous n'avons pas suffisamment de données pour produire un diagnostic sur les connaissances de l'élève. Dans le deuxième cas, nous avons des données significatives mais elles ne permettent pas de déterminer une corrélation entre le contexte et les actions. Enfin, le troisième cas nous permet de faire l'hypothèse d'une certaine stabilité du comportement de l'élève et on peut donc déduire une corrélation entre le contexte et l'action dont le nombre d'occurrences est le plus grand. Par exemple, si pour le contexte (equation, PosOrgArgEstAdd, SigneArgEstPlus) on a des données suffisantes stables avec n_2 majoritaire, on a une corrélation entre ce contexte et l'action ChangeSigneArg, et on considère (equation, PosOrgArgEstAdd, SigneArgEstPlus, ChangeSigneArg) comme un comportement pertinent pour le diagnostic des théorèmes-en-acte, car il exprime une régularité dans le comportement de l'élève. En fait ce comportement pertinent diagnostiqué est une règle d'action, au sens définit plus haut.

Etape 4 : Détermination automatique des théorèmes-en-acte et généralisation.

Le passage des VCLP aux TeA est effectué par des règles comme :

SI VCLP-Signe-3a ET VCLP-Signe-4B ALORS ValeurAbsolue-eq-mult

(VCLP-Signe-3a equation PosOrgArgEstMult SigneArgEstPlus ChangePasSigneArg)

(VCLP-Signe-4b equation PosOrgArgEstMult SigneArgEstMoins ChangeSigneArg)

qui expriment qu'on conserve le signe de l'argument dans le contexte équation et multiplicatif.

Un coefficient est associé aux TeA se trouvant en conclusion de ces règles. Lorsque le passage des VCLP aux TeA est réalisé, une propagation des coefficients est effectuée dans les hiérarchies de TeA. Les TeA qui sont produits comme résultats du processus sont ceux qui ont le contexte le plus large, c'est-à-dire qui sont les plus haut dans la hiérarchie. En d'autres termes, si un TeA T1 est élu et si son père T2 est aussi élu, T1 n'est pas présenté comme résultat car il est inclus dans T2.

Etude expérimentale

Nous avons mis en place des expérimentations pour déterminer les théorèmes-en-acte des élèves relatifs au changement de signe dans les équations, selon le modèle décrit ci-dessus. Les exercices ont porté sur des équations et inéquations dont la résolution nécessite des mouvements additifs et multiplicatifs d'arguments qui peuvent être de signe "+" ou "-". Pour ne pas mettre les élèves en difficulté sur des tâches qui ne concernent pas directement les théorèmes-en-acte sur les mouvements, nous avons choisi des exercices avec des coefficients entiers et avec des développements simples.

Dans une expérimentation mise en place dans une classe de seconde en France, au mois de mars 2003, nous avons proposé un nombre d'exercices suffisant pour pouvoir vérifier les stabilités des comportements des élèves. Ainsi, dans le test, il y avait 12 équations et 12 inéquations à résoudre. Pour chacune d'elles, nous avons déterminé le nombre et le signe du terme à déplacer selon la nature du mouvement. Par exemple, pour l'équation $x - 1 = 3x + 5$, nous avons a priori un mouvement additif d'un terme de signe "-", un autre d'un terme de signe "+" et un mouvement multiplicatif d'un terme de signe "-". Bien entendu, l'élève peut faire plus de mouvements en déplaçant un terme d'un membre à l'autre plusieurs fois.

Une expérimentation analogue a été mise en place au Brésil auprès de 32 classes de Quatrième, environ 1340 élèves âgés de 13-14 ans. Nous avons fait deux tests en deux séances différentes : le premier sur les équations de degré 1 et le deuxième sur les inéquations du degré 1. Chaque test avait 16 exercices dont les caractéristiques étaient les mêmes que celles de l'expérimentation réalisée en France.

Le tableau 1 donne les diagnostics obtenus pour la classe de seconde de France et trois classes du Brésil.

Le nombre de données insuffisantes est très élevé dans les deux classes 8A et 8B du Brésil. Cela est dû à des difficultés techniques au début de l'expérimentation, ce qui n'a pas laissé beaucoup de temps aux élèves de faire suffisamment d'exercices.

	8A Brésil	8B Brésil	8C Brésil	Seconde France
Nombre d'élèves	35	31	27	27
Données Insuffisantes	16	19	5	2
Données Instables	8	5	2	0
SigneCorrect	2	3	7	23
SigneCorrect-add	1	0	0	1
SigneCorrect-eq	4	2	2	0
SigneCorrect-eq-add	4	2	7	0
ValeurAbsolue			1	1
ConservationSigne			1	
ConservationSigne-ineq-add			2	

Tableau 1. *Résultats des calculs de théorèmes-en-acte pour le changement de signe.*

Dans les deux 8C et Seconde, le nombre de données stables est très important. Le tableau 1 montre des différences entre ces deux classes quant aux théorèmes-en-acte relatifs aux signes. Ainsi près de 85% des élèves de la classe de Seconde ont un théorème-en-acte correct contre 25% des élèves de la classe 8C. Pour cette dernière, on peut dire que le cas des mouvements additifs dans les équations est bien réussi par 16 élèves, soit 60%.

D'autres expérimentations ont été faites en France et au Brésil et ont donné lieu à des analyses⁶ quantitatives et comparatives entre les classes et entre les deux pays. Elles concernaient 3186 élèves, dont 2503 élèves brésiliens et 683 élèves français.

Conclusion

La modélisation des connaissances de l'élève occupe une place importante dans les recherches en AI-ED et en didactique des mathématiques. Pour certains domaines, le processus de diagnostic présente une certaine complexité, car il exige un découpage et une réorganisation des productions de l'élève puis une interprétation dans le modèle utilisé. Dans cet article, nous avons présenté un travail de modélisation des élèves en algèbre et montré en quoi les deux environnements informatiques Aplusix, pour le recueil des données, et Anaïs, pour leurs traitements, facilitent le processus de diagnostic des théorèmes-en-acte.

L'élaboration du modèle proposé a pris en compte deux préoccupations : l'une d'ordre didactique, en se plaçant dans un certain paradigme d'apprentissage, et l'autre d'ordre informatique, en imposant un modèle computationnel. Il faut noter que le modèle a été élaboré avec un niveau de granularité plus fin que lorsque l'on effectue une détermination totalement manuelle des théorèmes-en-acte.

La question principale que nous nous posions était la suivante : « Est-ce que le modèle que nous avons construit, de façon abstraite, pouvait être mis en œuvre de façon automatique et produire une discrimination significative entre les élèves ? ». La réponse est oui. L'automatisation du diagnostic, sur un plus grand nombre de données a confirmé nos premiers résultats. De plus, le diagnostic peut être fait sur n'importe quelle base d'exercices.

Le travail réalisé concerne les mouvements dans la résolution des équations de degré 1. Il se poursuit sur l'étude d'autres concepts comme la réduction, le développement, la factorisation...

•
⁶ Les résultats sont dans le rapport sur :
<http://www-leibniz.imag.fr/NEWLEIBNIZ/LesCahiers/2005/Cahier123/ResumCahier123.html>.

Bibliographie

- [Anderson 1983] ANDERSON J. R., *The Architecture of Cognition*. Harward University Press.
- [Anderson et al. 1990] ANDERSON J.R., BOYLE C. F., CORBETT A. T., LEWIS M. W., *Cognitive Modeling and Intelligent Tutoring*. Artificial Intelligence, Vol 42, no 1.
- [Artigue 1988] ARTIGUE M., Ingénierie didactique. *Recherches en Didactique des Mathématiques*, vol.9 n°3, pp.281-308.
- [Artigue et Robinet 1982] ARTIGUE M. ET ROBINET J., Conception du cercle chez des enfants de l'école élémentaire. *Recherches en Didactique des Mathématiques*, 3(2) 5-64.
- [Balacheff 1994] BALACHEFF N. Didactique et intelligence artificielle. *Recherches en didactiques des mathématiques* 14 (1/2) 9-42.
- [Balacheff 1995] BALACHEFF N., Conception, connaissance et concept. In Denise Grenier (ed.) *Séminaire Didactique et Technologies cognitives en mathématiques* (pp.219-244). Grenoble. IMAG.
- [Balacheff 2000] BALACHEFF N., Les connaissances, pluralité de conceptions (le cas des mathématiques). In : Tchounikine P.(ed.) *Actes de la conférence Ingénierie de la connaissance* (IC 2000, pp.83-90). Toulouse.
- [Brousseau 1986] BROUSSEAU G., [Fondements et méthodes de la didactique des mathématiques. *Recherches en Didactique des Mathématiques*, 7(2) 5-64.
- [Chaachoua et Lima 2003] CHAACHOUA H., Lima I., De la modélisation des conceptions des élèves à la prise de décisions didactiques par l'enseignant : le rôle d'un environnement informatique. *Actes du colloque ITEM 2003*, Reims, 20-22 juin 2003
- [Croset 2005] CROSET M.C., Modélisation didactique et informatique des connaissances des élèves en algèbre. Cas de développement et de réduction en algèbre. Mémoire Master2. Recherche, EIAHD, UJF, Grenoble.
- [Dershowitz et Jouannaud 1989] DERSHOWITZ N, JOUANNAUD J.P., Rewrite Systems. In *Handbook of Theoretical Computer Science*, Vol B, Chap 15. North-Holland.
- [Nicaud et al. 2004] NICAUD, J.F., BOUHINEAU, D., AND CHAACHOUA H., Mixing microworld and CAS features in building computer systems that help students learn algebra, in *International Journal of Computers for Mathematical Learning*, Volume 9, Issue 2, Springer-Verlag

- [Pearl 1984] PEARL J., *Heuristics*. Addison-Wesley, London.
- [Tahri 1993] TAHRI S., *Modélisation de l'interaction didactique : un tuteur hybride sur Cabri-géomètre pour l'analyse des décisions didactiques*. Thèse. Grenoble : Université Joseph Fourier.
- [Vergnaud 1991] VERGNAUD G., La théorie des champs conceptuels, *Recherches en Didactique des Mathématiques*, vol.10 n°2-3, pp.133-169
- [Vergnaud 2001] VERGNAUD G., Forme opératoire et forme prédicative de la connaissance, Conférence publiée dans les Actes du Colloque GDM-2001 Jean Portugais (Ed) *La notion de compétence en enseignement des mathématiques, analyse didactique des effets de son introduction sur les pratiques et sur la formation*.
- [Wenger 1987] WENGER E., *Artificial Intelligence and Tutoring Systems*. Los Altos : Morgan Kaufmann Pub. Inc.

Annexe A : Présentation du logiciel Aplusix

Aplusix est un logiciel d'aide à l'apprentissage de l'algèbre formelle avec lequel l'élève développe ses propres calculs. Il est doté pour cela d'un éditeur avancé d'expressions algébriques qui affiche les expressions sous la forme habituelle (en deux dimensions) et permet leur écriture et leur modification sous cette forme, en respectant la structure des expressions algébriques.

Le logiciel permet à l'élève de construire les étapes (symbolisées par des boîtes rectangulaires) de calcul de son choix et peut vérifier les calculs effectués en calculant l'équivalence des deux expressions. La figure 1 ci-dessous montre la vérification de l'équivalence des équations entre chaque étape. Aplusix apporte aussi des informations sur l'expression courante avec des indicateurs comme on peut le voir sur la figure 1. Ces indicateurs concernent le degré d'avancement du développement, de la factorisation ou de la réduction d'une expression algébrique. Dans le même esprit, un indicateur est associé à la résolution des équations. Dans la figure 1, l'élève a effectué un premier pas de calcul juste marqué par une flèche d'équivalence noire et un deuxième pas incorrect marqué par une flèche rouge et barrée. La barre d'état indique que l'expression courante est bien formée, non réduite, non développée et que l'équation n'a pas encore significativement progressé vers une solution.

Lorsque l'élève a terminé un exercice, il l'indique au logiciel qui vérifie que l'expression obtenue est une forme résolue du problème et que le chemin qui conduit de l'énoncé à cette expression n'est composé que d'expressions équivalentes. Le logiciel transmet alors un message textuel lui donnant une information sur l'état de la résolution de l'exercice : l'exercice est effectivement terminé, ou bien il reste une action encore à faire comme développer, réduire, ... Si l'exercice n'est pas fini, l'élève a le choix entre laisser l'exercice en l'état ou essayer de trouver une solution.

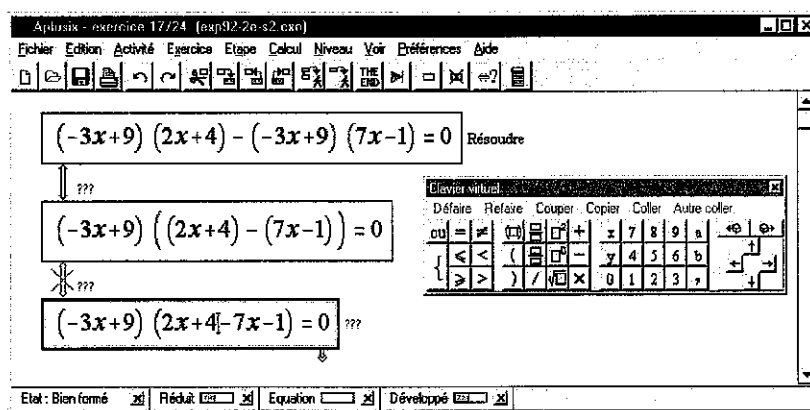


Figure 1 : un exemple de raisonnement avec Aplusix

Le logiciel Aplusix propose des commandes telles que *calculer* (effectuer un calcul numérique), *réduire*, *développer* qui permettent à l'élève de sous-traiter des tâches à l'ordinateur, à l'instar des logiciels de calcul formel, avec des fonctionnalités plus limitées. Ces commandes permettent au logiciel de prendre en charge des tâches algébriques. Elles permettent ainsi, notamment, de développer, réduire ou ordonner des expressions en fournissant directement le résultat du calcul. Outre la possibilité de paramétrer le logiciel du point de vue de la plupart des caractéristiques précédentes, le professeur dispose de deux outils importants : l'un permet de constituer des fichiers d'exercices pour les donner ensuite à résoudre aux élèves, l'autre est un « magnétoscope » qui permet de rejouer toutes les actions des élèves.

Annexe B : Les TeA globaux

Les 5 composants principaux (TeA globaux) du trait « Signe dans mouvement »

SigneCorrect : Traitement correct du signe de l'argument.

ValeurAbsolue : Changement du signe de l'argument si et seulement si il est négatif.

ValeurAbsoluePartielleM : Changement du signe de l'argument si et seulement si « il est négatif et sa position à l'origine est multiplicative » ou si « sa position à l'origine est additive ».

ConservationSigne : Jamais de changement du signe l'argument.

ChangementSigne : Toujours changement du signe l'argument.

Les 4 composants principaux (TeA globaux) du trait « Sens dans mouvement »

SensCorrect : Traitement correct du sens de l'inégalité.

SensUnifié : Changement du sens de l'inégalité si et seulement si le signe de l'argument est négatif.

ConservationSens : Jamais de changement du sens de l'inégalité.

ChangementSens : Toujours changement du sens de l'inégalité.

Les 5 composants principaux (TeA globaux) du trait « Opérateur dans mouvement »

OperateurCorrect : Traitement correct de l'opérateur.

OperateurUnifieA : La position finale de l'argument est toujours additive.

$$(a+x=b \rightarrow x=b \pm a \quad ax=b \rightarrow x=b \pm a \quad x/a=b \rightarrow x=b \pm a)$$

OperateurUnifieMN : La position finale de l'argument est toujours multiplicative au numérateur

$$(a+x=b \rightarrow x=ab \quad ax=b \rightarrow x=ab \quad x/a=b \rightarrow x=ab)$$

OperateurUnifieMD : La position finale de l'argument est toujours multiplicative au dénominateur

$$(a+x=b \rightarrow x=b/a \quad ax=b \rightarrow x=b/a \quad x/a=b \rightarrow x=b/a)$$

ConservationOperateur : Report de l'opérateur.

$$(a+x=b \rightarrow x=b \pm a \quad ax=b \rightarrow x=ab \quad x/a=b \rightarrow x=b/a)$$

Annexes C

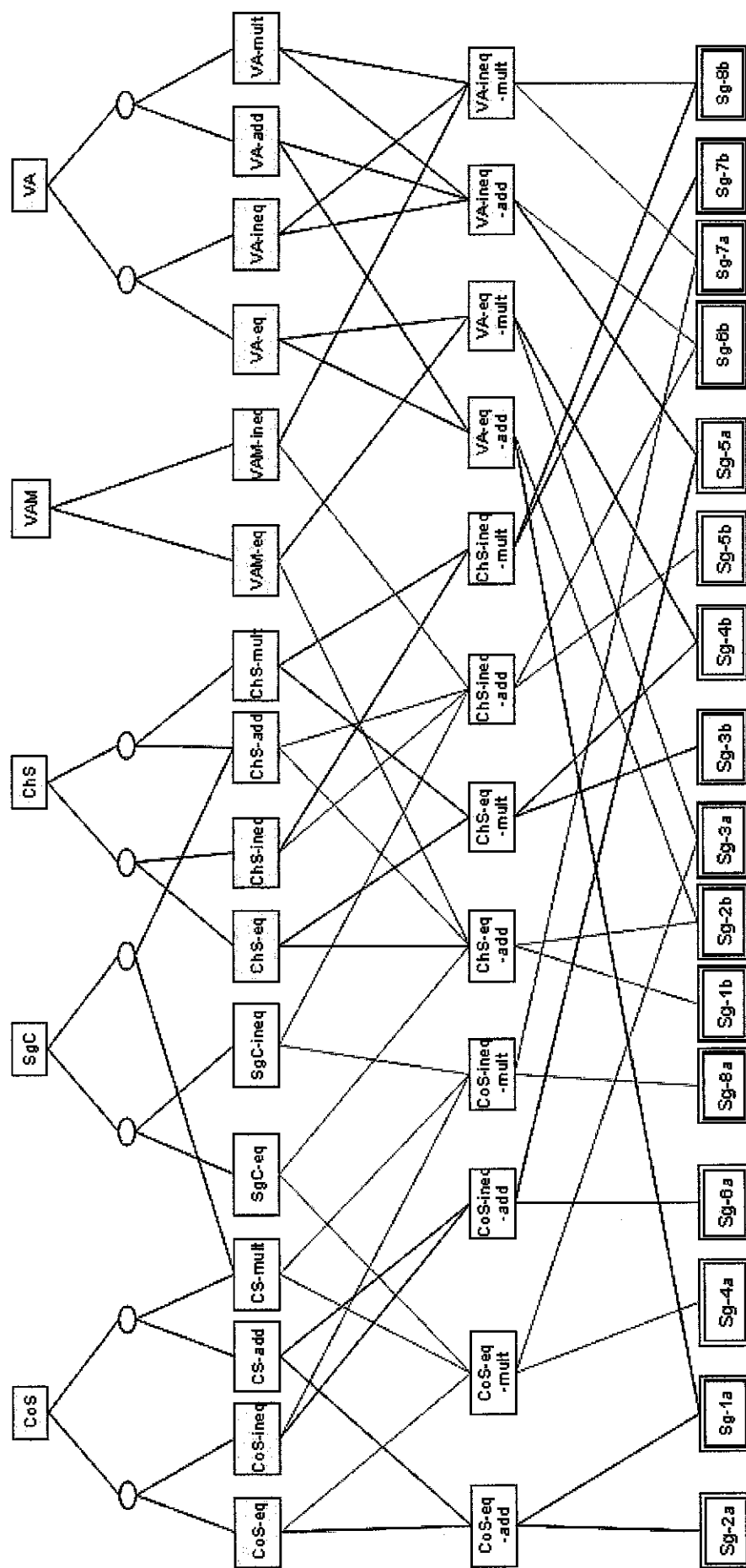
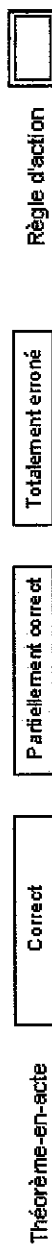


Figure Treillis des théorèmes-en-acte du trait « Signe dans mouvement »



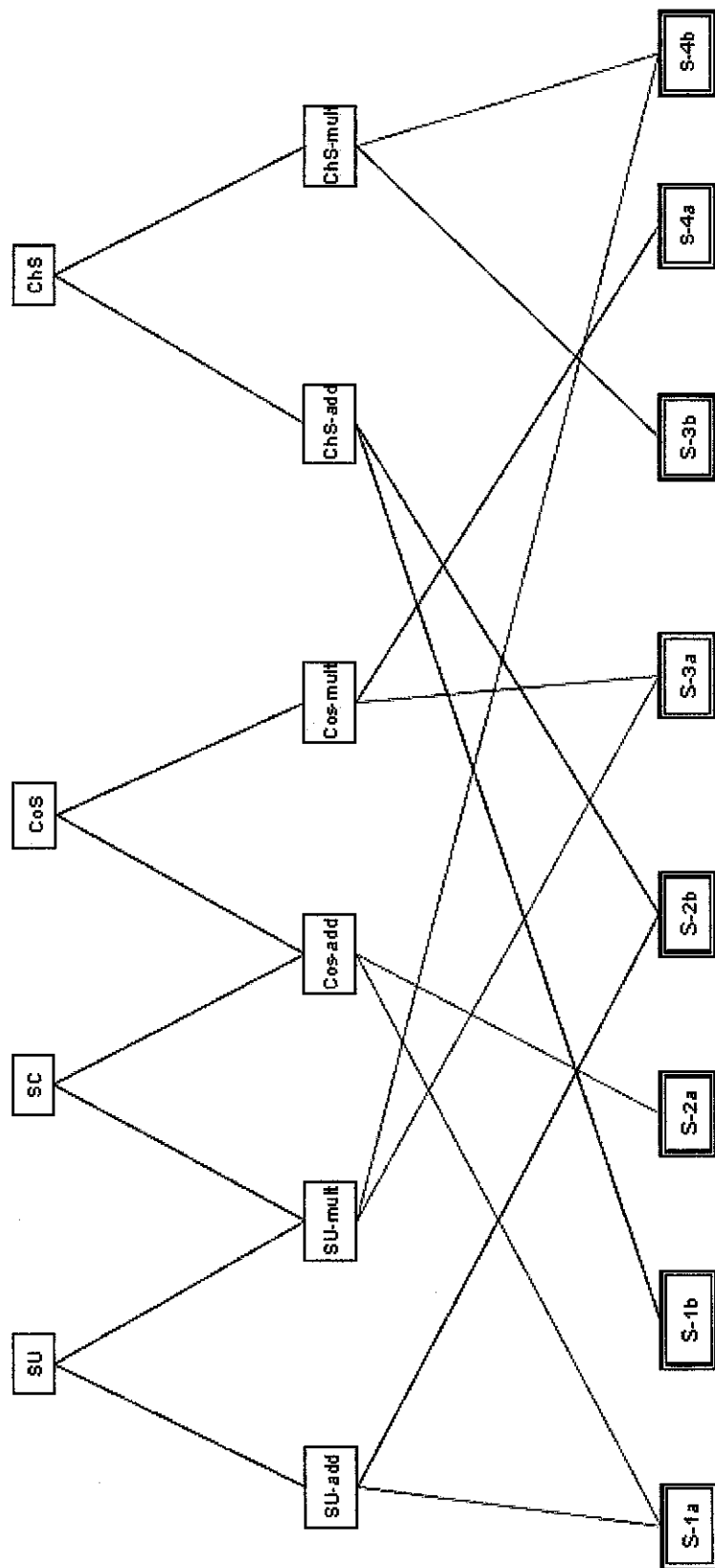


Figure . Treillis des théorèmes-en-acte du trait « Sens dans mouvement »

