



**HAL**  
open science

## Modélisation de profils d'élèves à partir de traces de résolution de problèmes

Vivien Robinet, Gilles Bisson, Benoît Lemaire, Mirta Gordon

► **To cite this version:**

Vivien Robinet, Gilles Bisson, Benoît Lemaire, Mirta Gordon. Modélisation de profils d'élèves à partir de traces de résolution de problèmes. *Les Cahiers Leibniz*, 2006, 150, pp.1-24. hal-00257168

**HAL Id: hal-00257168**

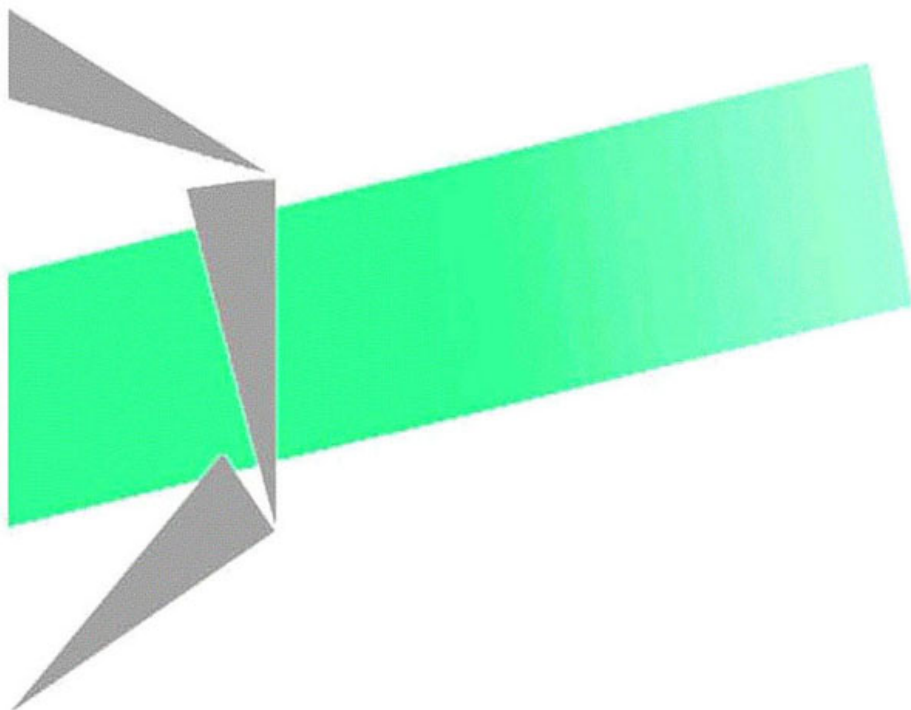
**<https://telearn.hal.science/hal-00257168>**

Submitted on 18 Feb 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Les cahiers Leibniz



Modélisation de profils d'élèves à partir de  
traces de résolution de problèmes

Vivien Robinet, Gilles Bisson,  
Benoît Lemaire, Mirta B. Gordon

Laboratoire Leibniz-IMAG, 46 av. Félix Viallet, 38000 GRENOBLE, France -  
ISSN : 1298-020X

n° 150  
Juin 2006

Site internet : <http://www-leibniz.imag.fr>



# Modélisation de profils d'élèves à partir de traces de résolution de problèmes

Vivien Robinet, Gilles Bisson, Benoît Lemaire, Mirta B. Gordon.

Email : {prénom.nom@imag.fr}

Equipe Apprentissage - Laboratoire Leibniz-IMAG, UMR 5522

46 Avenue Félix Viallet - 38031 GRENOBLE Cedex

**Résumé** : La modélisation automatique des connaissances de l'apprenant est nécessaire dès lors que l'on cherche à accompagner un apprentissage dans le cadre des EIAH. Dans ce rapport, nous présentons un travail en cours dans lequel des techniques de classification automatique permettent d'engendrer des modèles d'élèves interprétables par un enseignant ou un didacticien et utilisables éventuellement par un « tuteur » automatique. L'approche proposée a été appliquée au domaine de la résolution d'équations ou d'inéquations par des classes de 4<sup>ème</sup> et 3<sup>ème</sup>. Dans les expérimentations qui servent de support à l'apprentissage automatique, l'élève résout de manière libre un grand nombre de problèmes (comme  $5x-3 < -2x+11$ ) et nous cherchons à repérer et modéliser ses connaissances à partir de ses traces de résolution.

**Mot-clés** : EIAH, Modélisation didactique, Construction de profils d'élèves, Classification automatique, Apprentissage de l'algèbre, APLUSIX.

## 1 Introduction

Les Environnements Informatiques pour l'Apprentissage Humain (EIAH) ont pour objectif de faciliter l'acquisition de nouvelles connaissances. Dans un tel contexte, il est important d'avoir la capacité de diagnostiquer automatiquement les acquis et déficits de l'apprenant afin de faciliter et de guider cet apprentissage. Ce rapport présente les premiers résultats d'une méthode générale de caractérisation du profil cognitif d'un apprenant à partir des traces de ses résolutions de problèmes. Les traces sont ici vues comme des suites d'exemples de comportements élémentaires, caractérisés par un ensemble de descripteurs décrivant le contexte courant, les actions effectuées par l'apprenant et la situation résultante. La méthode cherche ensuite à repérer des régularités par une technique de classification ascendante permettant de regrouper ces observations par niveaux de généralité croissants. Cela conduit à l'élaboration d'*attitudes individuelles* ou *collectives* qui correspondent à des comportements typiques mis en œuvre par le (ou les) apprenant(s) face à un contexte donné.

Prenons un exemple. Dans le cas de la conduite automobile, le fait de mettre son clignotant gauche, de jeter un coup d'oeil à gauche et de prendre la file intérieure pour faire le tour d'un rond-point constitue une attitude, dès lors que suffisamment d'exemples de ce comportement auront pu être repérés. Différentes variables peuvent caractériser une attitude :

- tout d'abord, une attitude peut être *correcte* ou *incorrecte*. L'intérêt de repérer automatiquement les attitudes incorrectes est important dans le contexte éducatif ;
- ensuite une attitude peut également être caractérisée par un *degré de confiance*, dépendant du nombre d'observations qui la fondent ;
- enfin, une attitude (qu'elle soit correcte ou non), peut être soit *cohérente* si l'apprenti l'a systématiquement mise en œuvre soit *incohérente*. C'est le cas de l'apprenti qui prend parfois la file intérieure du rond-point et parfois la file extérieure dans des situations qui nous paraissent similaires. La difficulté dans ce cas sera de faire la part entre une vraie incohérence chez l'apprenti et un défaut dans notre description du problème.

Dans le cas de la conduite automobile, le contexte pourrait être caractérisé par des descripteurs comme la densité du flux automobile, la présence de panneaux, la nature du carrefour, la direction souhaitée, etc. L'action serait, elle, décrite par des descripteurs comme l'usage des clignotants, la réduction de la vitesse, le choix de la file, le freinage ou non, etc.

Plus précisément notre approche est basée sur trois grandes étapes (figure1) :

- une phase dépendante de l'EIAH. Elle vise à recueillir les exemples de comportements d'apprenant (les *observables*) à partir desquels seront construites les *attitudes* ;
- une phase dépendante du domaine d'application. Il s'agit d'encoder les *observables* sous forme de *cas* en utilisant le langage de représentation du système d'apprentissage ;
- une phase générique qui consiste à catégoriser les cas de manière à engendrer les *attitudes individuelles* et *collectives* et à les exprimer sous forme de notices en langage naturel.

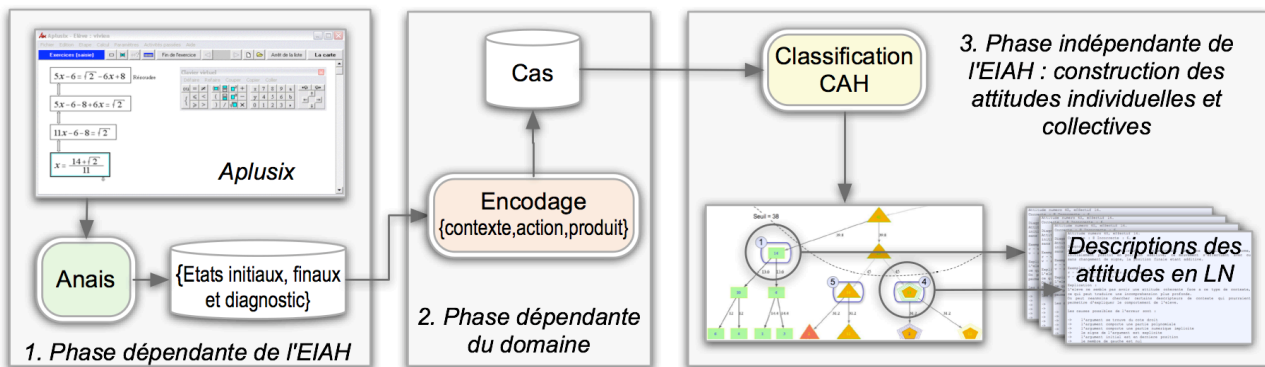


Figure 1. Architecture générale de notre approche

Ce rapport est structuré de la manière suivante. Dans une première partie (§ 2) nous présentons le contexte du travail, qui porte sur l'enseignement de l'algèbre et plus particulièrement la résolution d'équations ou d'inéquations. Nous abordons ensuite la modélisation de l'élève (§ 3), avec un bref état de l'art sur les systèmes d'EIAH capables de modéliser le comportement d'un apprenant, suivie d'une description de la représentation des données utilisées ici. La partie (§ 4) explique la méthode de classification permettant d'engendrer les *attitudes* et la partie (§ 5) présentent les résultats obtenus et montre comment ces *attitudes* peuvent être mises sous une forme intelligible pour un enseignant ou un didacticien. Enfin, dans la conclusion nous discutons de la généralité de notre approche et de son utilisation possible pour mettre en œuvre des tuteurs automatiques.

## 2 Contexte du travail

### 2.1 L'éditeur Aplusix

Le logiciel APLUSIX (Nicaud *et al.*, 2002) est un logiciel d'EIAH ayant pour objectif d'aider à acquérir et à renforcer les connaissances de base d'un élève en le confrontant à un environnement d'apprentissage de l'algèbre de type *micro-monde*, c'est-à-dire, dans lequel l'élève se forme en explorant librement le domaine dans lequel il travaille. L'acquisition et le renforcement des connaissances s'effectuent tout naturellement en résolvant des exercices d'algèbre. L'élève résout chaque problème *étape par étape* comme s'il le faisait sur une feuille de papier (cf. figure 2). Selon les *consignes* adoptées par l'enseignant pour chaque session de travail, l'élève peut demander au logiciel de vérifier son travail en continu, à la demande ou rétroactivement à la fin de son exercice. Ce logiciel impose néanmoins de ne pouvoir passer à l'étape suivante que lorsque la ligne en cours est correctement formatée (parenthèses fermées, expression complète, ...).

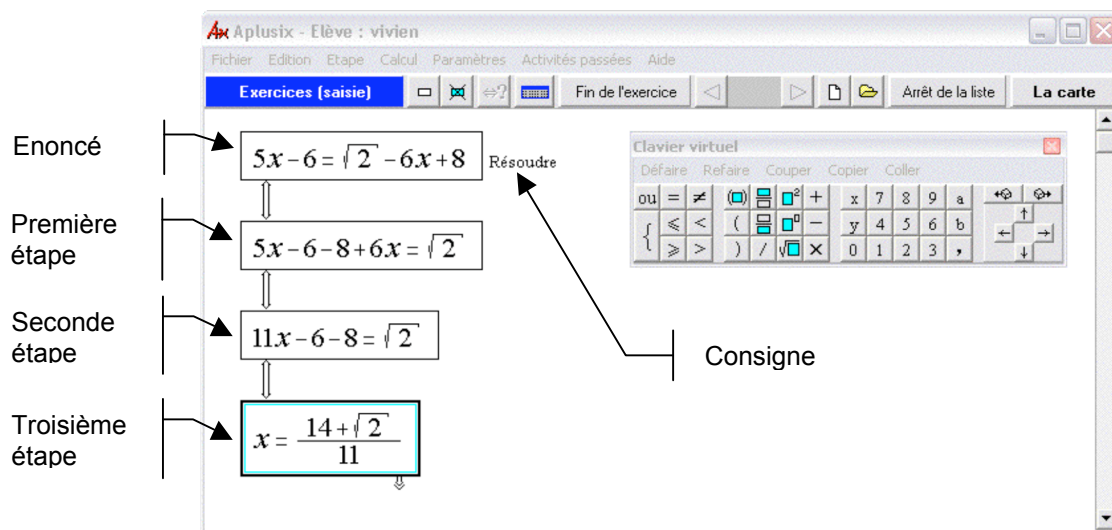


Figure 2 : Copie d'écran de l'environnement APLUSIX.

Outre l'enseignement de l'algèbre, l'objectif d'APLUSIX est d'offrir une plateforme d'expérimentation didactique. A cette fin, la séquence des actions élémentaires effectuées par l'élève pour résoudre un exercice est enregistrée dans un fichier texte qui est appelé le *protocole brut*. Dans le cadre d'expérimentations effectuées dans les années 2003-2005 en France et au Brésil, l'équipe d'APLUSIX a collecté un grand nombre de traces d'élèves résolvant des problèmes. C'est ce matériau qui constitue le point d'entrée de notre approche.

## 2.2 Analyse des protocoles

Chaque ligne d'un fichier de protocole brut représente une *action élémentaire* effectuée par l'élève, correspondant à une frappe de touche, une utilisation de la souris, ou l'exécution d'une fonction dans un menu d'APLUSIX. Il conserve également d'autres informations, telles : l'expression en cours et la durée en secondes séparant deux actions. Si les protocoles bruts permettent d'analyser l'interaction d'un élève avec le système (temps de résolution, commande utilisée, ...), les informations qu'ils contiennent sont de trop bas niveau pour mettre aisément en évidence des *attitudes* caractéristiques des connaissances de l'élève. Inversement si l'on se limite à n'observer que les *étapes du raisonnement* (cf. 2.1), la granularité devient trop élevée car entre deux étapes successives l'élève peut effectuer de nombreuses opérations de nature différentes.

Afin de retrouver automatiquement<sup>1</sup> un bon niveau d'observation, l'équipe d'APLUSIX a développé un module, nommé ANAIS, d'interprétation symbolique des *protocoles bruts*. Ce module est composé de deux parties. D'une part, d'un ensemble de règles élémentaires de réécriture, définies par des didacticiens, permettant de caractériser le passage d'une expression algébrique à une autre. Ces règles représentent soit des transformations « correctes » (par exemple :  $(a+b)^2 \rightarrow a^2+2ab+b^2$ ), soit des transformations « erronées » lorsqu'il s'agit de représenter les erreurs des élèves (par exemple :  $x^n \rightarrow nx$ ). D'autre part, ANAIS contient un moteur de recherche travaillant en stratégie *meilleur d'abord*. Ce moteur permet de retrouver de manière heuristique, à partir du protocole brut, la plus courte séquence de règles dont l'application construit un chemin logique entre une étape de

<sup>1</sup> Nous ne demandons pas cette information directement aux élèves pour plusieurs raisons. Tout d'abord, comme nous l'avons dit APLUSIX travaille dans un paradigme *micro-monde* où l'élève est laissé autonome ; ensuite, ce dernier n'a pas forcément conscience des étapes intermédiaires qu'il produit ; enfin la tâche d'acquisition de ces données sur l'ensemble des exercices serait trop lourde ; dans le cas des expérimentations au Brésil, il y a 2700 élèves impliqués.

l'élève et l'étape suivante. Cette séquence correspond à une interprétation<sup>2</sup> du raisonnement de l'élève. Ainsi, dans la figure 3 ci-dessous, on voit que le logiciel ANAIS interprète la transformation de l'élève comme l'application successive de deux règles : R1 puis R2.

Ainsi, ANAIS fournit plusieurs informations qui vont nous être utiles dans ce travail. D'une part, il permet d'identifier les règles, et donc les *types d'opérations* (mouvement, factorisation, ...) implicitement utilisées par l'apprenti en indiquant si elles sont algébriquement *correctes ou non*. D'autre part, les étapes produites par l'élève peuvent contenir simultanément des transformations justes et erronées ; en introduisant des *étapes intermédiaires homogènes*, du point de vue de la (non)correction, le logiciel ANAIS permet de déceler plus précisément les (mé)connaissances de l'élève. De la sorte, les informations fournies par ANAIS constituent donc des étapes logiques élémentaires, qui sont soit produites directement par les élèves, soit engendrées par ANAIS par décomposition des étapes de résolution des élèves. Finalement, pour effectuer une modélisation globale de l'élève il est nécessaire d'agréger ces informations et de les interpréter.

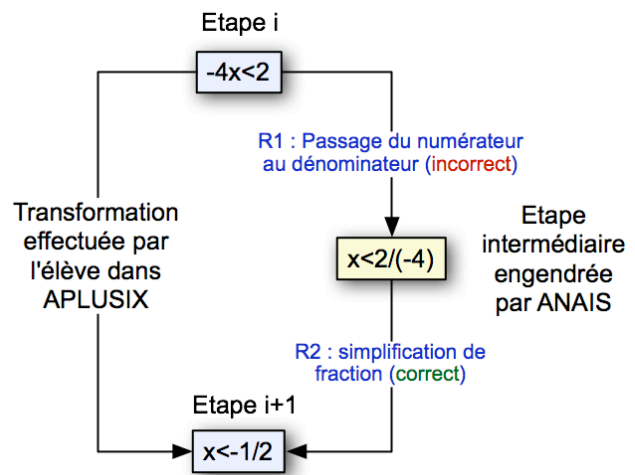


Figure 3 : analyse des protocoles par ANAIS

### 3 Modélisation de l'élève

#### 3.1 Etat de l'art

Généralement, les traces de résolution de problèmes qui sont enregistrées par un EIAH prennent la forme, comme dans le cas d'APLUSIX, de longues séquences de micro-comportements qu'il s'agit d'interpréter en changeant de niveau de granularité (Mostow, 2004). Plusieurs travaux ont déjà utilisé des techniques d'apprentissage automatique afin de découvrir, derrière une succession de clics de souris, les connaissances du domaine bien ou mal maîtrisées par l'élève. Dans tous les cas, l'extraction de ces régularités nécessite une redescription préalable des productions des élèves en termes de plus haut niveau, définis par des experts humains, et qui sont spécifiques du domaine d'application visé par l'EIAH. C'est également ce que nous faisons dans ce travail, en nous appuyant sur les interprétations produites par ANAIS.

Notons que beaucoup de travaux construisent les modèles d'élèves avec un apprentissage supervisé qui s'appuie sur des profils pré-établis par des experts du domaine. C'est le cas de Profile Extractor (Esposito *et al.*, 2004) qui utilise des données comportementales d'élèves, issues de leurs interactions avec un système de e-learning, pour construire un arbre de décisions capable d'assigner chaque élève à une classe prédéfinie. Nous nous distinguons de ces approches en ce sens que notre objectif est de découvrir ces profils.

*AnimalWatch* (Beck & Woolf, 2000) qui concerne le domaine de l'arithmétique est plus proche de nos considérations. Le système analyse les données comportementales d'un élève pour prédire s'il sera capable de résoudre un problème donné et le temps qu'il y consacra. Un tel modèle est très utile puisque, à partir de ses prédictions, on peut fournir à l'élève des problèmes adaptés à son niveau de compétences et qui lui permettront de progresser. Les descripteurs de haut niveau qui sont utilisés par *AnimalWatch* appartiennent à quatre catégories :

<sup>2</sup> Dans ce travail, nous n'analysons que les opérations de déplacement de termes dans les expressions (passage d'un terme du membre gauche vers le membre droite, ou inversement). Dans un tel contexte, plus de 80% des interprétations effectuées automatiquement par ANAIS ont été validées par les didacticiens travaillant dans ce projet.

- *Student* : le sexe, le score à différents tests ;
- *Topic* : le type d'opérateur (addition, soustraction, etc.) et le type d'opérande ;
- *Problem* : le degré de difficulté du problème, le nombre de prérequis nécessaires pour le résoudre (par exemple : addition de fractions, simplification de fractions, calcul du PPCM, etc.), et enfin le degré de difficulté des prérequis ;
- *Context* : le nombre d'erreurs antérieures, le niveau des conseils donnés par le système, etc.

Dans *AnimalWatch*, différents algorithmes ont été utilisés pour prédire la probabilité de succès de l'élève et le temps consacré, tels qu'un classifieur bayésien et un arbre de décisions, mais les auteurs se sont finalement rabattus sur une simple régression linéaire. Là encore nous nous distinguons de cette approche par le domaine d'application et surtout la finalité : nous cherchons à construire tout un profil cognitif et non à prédire la valeur de deux variables. Cependant, comme nous le verrons nous utilisons certains des descripteurs de *AnimalWatch*, notamment ceux de la catégorie *Topic*.

Dans le domaine de la modélisation d'élèves en mathématique, le système *Web-EasyMath* (Tsiriga et Virvou, 2004) évite d'assigner une classe prédéfinie à chaque nouvel élève en lui faisant passer un test préliminaire permettant d'extraire des descripteurs pertinents (auto-estimation de ses compétences dans le domaine et taux de réponses correctes aux quatre opérations de base). En appliquant un algorithme des *k plus proches voisins*, il le positionne par rapport aux autres élèves en calculant leurs similarités. Ceci permet d'initialiser le modèle de l'élève, représenté *in fine* par un ensemble de connaissances, accompagnées par la probabilité que l'élève les possède.

Sison & Shimura (1998) établissent un état de l'art des travaux qui appliquent des techniques d'apprentissage automatique dans les EIAH et proposent plusieurs axes pour classer les différentes manières d'apprendre un modèle d'élève à partir de ses « comportements », qui peuvent être des traces de résolution ou plus simplement des réponses à des questions :

- Complexité du comportement étudié : depuis une simple valeur jusqu'à des expressions élaborées. Dans notre cas, les comportements de l'élève sont relativement complexes puisqu'il s'agit de la manipulation d'expressions algébriques ;
- Multiplicité des comportements : certaines approches peuvent se satisfaire d'analyser un comportement unique pour inférer des connaissances sur l'élève, alors que d'autres, comme la nôtre, nécessitent un grand nombre d'observations de comportements ;
- Connaissances du domaine : elles peuvent être codées dans le système ou elles-mêmes apprises automatiquement ; dans notre cas, elles sont en partie définies automatiquement mais aussi représentées avec des descripteurs sélectionnés de manière *ad hoc* ;
- Utilisation de « misconceptions » (bibliothèque d'erreurs) : là encore, cet ensemble peut être établi à l'avance ou appris. Dans notre approche, les règles erronées sont apprises par le système. Toutefois, la notion de correction d'une étape de raisonnement de l'élève est fournie par ANAIS en comparant les « domaines solution » avant et après chaque étape, en outre celui-ci se sert de ses propres règles erronées pour subdiviser les étapes.
- Construction du modèle de l'élève : elle peut être analytique (en essayant de retrouver les étapes qu'a pu suivre l'élève) ou synthétique (par généralisation des comportements). La détection des étapes intermédiaires effectuée par ANAIS se situe dans la première modalité, et notre approche se place dans la seconde.

### 3.2 Représentation des données d'apprentissage

Les données sur lesquelles nous avons travaillé proviennent d'expérimentations effectuées au Brésil au printemps 2004. Le niveau des classes étudiées était équivalent à la 4<sup>ème</sup> et 3<sup>ème</sup> de nos collèges. Les expérimentations ont concerné 2700 élèves environ, chaque élève ayant traité en moyenne entre 3 et 10 problèmes (résolution d'équations et/ou inéquations).



Comme nous l'avons expliqué (§ 2.2) les profils de l'élève que nous construisons s'appuient sur les *observables* fournis par ANAIS, c'est-à-dire sur la décomposition des étapes de l'élève en *étapes élémentaires* correspondant à l'application d'une seule règle de réécriture. Nous modélisons ici les manipulations algébriques effectuées par l'élève, indépendamment de la séquence d'actions dans lesquelles elles apparaissent ; il ne s'agit donc pas de modéliser la stratégie de résolution de l'apprenant. En outre, dans la présente étude, nous ne nous intéressons qu'à une seule famille de règles : celles concernant le déplacement des termes d'un côté à l'autre d'une expression (équation ou inéquation), ce que nous appellerons par la suite un *mouvement*. En voici deux exemples :

$$\begin{cases} 2x - x < 2 \rightarrow 2x < 2 + x \\ -3x(x - 5) < 4 \rightarrow (x - 5) > -4/3x \end{cases}$$

Ce choix est motivé, pragmatiquement, par le fait que ANAIS contient un ensemble de règles de réécriture validées par des didacticiens pour ce type d'opérations (ce qui est un prérequis permettant de décomposer la trace de résolution en étapes élémentaires). Mais il convient de souligner également que la compréhension des mécanismes de déplacement de termes constitue une phase fondamentale dans l'apprentissage de l'algèbre en classes de 4<sup>ème</sup> et 3<sup>ème</sup>. De manière synthétique, chacun des *observables* est constitué par un couple d'expressions algébriques correspondant à une transition d'un *état initial* vers un *état final* ainsi que d'un *diagnostic* indiquant le type (ici c'est toujours un « mouvement ») et la justesse de cette transition :

$$Observable_i : Etat\ initial \xrightarrow{D_j} Etat\ final \quad \left\{ \begin{array}{l} D_j : \text{diagnostic indiquant le type de la transition} \\ \text{effectuée et sa justesse (correcte/incorrecte)} \end{array} \right.$$

Nous n'effectuons pas l'apprentissage directement avec cette représentation pour deux raisons. D'une part, il est délicat de travailler sur des expressions algébriques dont la sémantique est complexe : ainsi, deux équations syntaxiquement voisines peuvent être sémantiquement très différentes. D'autre part, il est nécessaire d'explicitier les actions effectuées par l'apprenant afin d'apprendre correctement les profils. Aussi, nous allons modéliser notre ensemble d'apprentissage en extrayant de chaque *observable* un triplet  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$ , que nous appellerons un *cas*, et qui est décrit ici par un ensemble de descripteurs (figure 4). La partie *contexte* d'un *cas* explicite les informations de l'*état initial* qui sont significatives vis-à-vis à la fois du type de transition effectuée ainsi que du but de la modélisation. De façon similaire, la partie *produit* caractérise les informations pertinentes associées à l'*état final*. La partie *action* décrit les modifications effectuées par l'élève pour passer de l'*état initial* à l'*état final* ; cette partie indique également si la transition effectuée est correcte ou non. Cette modélisation est voisine de celle proposée par Renaudie (2005), mais chez ce dernier les parties *action* et *produit* sont confondues ce qui introduit des ambiguïtés sur la signification des descripteurs.

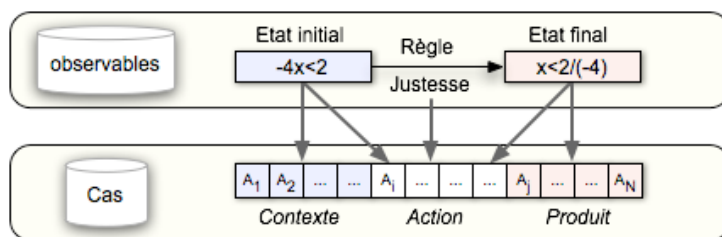


Figure 4 : Modélisation d'un observable sous le forme d'un cas  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$ .

Ainsi que nous l'avons évoqué dans l'introduction, grâce à cette représentation très générale  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$ , nous allons pouvoir mettre en œuvre un mécanisme d'apprentissage qui est dans une certaine mesure indépendant du domaine étudié et donc transposable à d'autres problèmes. Cet aspect sera discuté plus en détail dans la conclusion du rapport. Les modèles que nous voulons apprendre (cf. §3) visent à repérer des comportements « remarquables » de l'apprenant

que l'on appellera des *attitudes* et qui caractérisent l'état de ses connaissances. D'un point de vue didactique, deux catégories d'attitudes opposées sont intéressantes à rechercher :

- les « théorèmes en actes » (Vergnaud 2001), qui correspondent à des attitudes dans lesquels les élèves réalisent toujours le même type de transformation à partir d'un ou de plusieurs contextes différents. Ces théorèmes en actes peuvent soit correspondre à des connaissances correctes ou à la mise en œuvre de connaissances partiellement ou totalement erronées.
- les « irrégularités » (Renaudie 2005) qui correspondent à des attitudes dans lesquelles les élèves effectuent des transformations différentes dans des contextes sémantiquement identiques.

### 3.3 Descripteurs des triplets

#### 3.3.1 Description du contexte et du produit

Les descripteurs retenus pour décrire les parties *contexte* et *produit* d'un *cas* doivent caractériser à la fois la réalité mathématique de la modification effectuée sur l'expression (en termes d'opérandes et d'opérateurs mis en jeu), mais également essayer d'explicitier l'ensemble des « perceptions » de l'élève. En effet, un débutant en algèbre peut être perturbé par la présence (ou l'absence) de certains symboles qu'il maîtrise mal, même s'ils sont mathématiquement non significatifs vis-à-vis de l'opération effectuée. Par exemple, le terme «  $-1x$  » peut être perçu comme étant de nature différente de «  $-x$  » ; de même, la simple présence d'une racine carrée dans une équation peut avoir un effet perturbant pour l'élève. Nous en verrons des illustrations dans la section §5.

De ce fait, nous avons défini un ensemble assez large de descripteurs, dont la liste détaillée se trouve en Annexe A, afin de prendre en compte cette double dimension mathématique et perceptive. Certains descripteurs sont redondants (par exemples *argument.opérateur* et *argument.catégorie*) afin de subdiviser certains domaines de valeurs en différents intervalles ; cela permet d'exprimer des taxonomies de valeurs et d'avoir plusieurs niveaux de généralisation possibles. Soulignons bien ici qu'il ne s'agit pas de réaliser une description complète des expressions algébriques mais simplement de ne garder que les informations qui sont soit impliquées dans la transformation effectuée, soit susceptibles de changer le regard que porte l'élève sur l'expression. En ce qui concerne les descripteurs du *produit*, on n'a gardé que ceux qui sont pertinents pour la transformation ; ils constituent donc un sous-ensemble de ceux décrivant le *contexte*.

#### 3.3.2 Description des actions

Les *actions* explicitent les modifications effectuées par l'élève dans l'observable pour passer de *l'état initial* à *l'état final*. Dans notre représentation, ces modifications peuvent être souvent déduites automatiquement du *contexte* et du *produit* car elles expriment simplement le fait qu'il y a eu ou non une modification de la valeur de leurs descripteurs (table 1).

Contexte	Produit	Action déduite
contexte.arg.opérateur = +	produit.arg.opérateur = +	action.arg.opérateur = faux
contexte.arg.opérateur = +	produit.arg.opérateur = -	action.arg.opérateur = vrai

Table 1 : Exemples de déduction de la valeur de l'action.

Il est important de faire apparaître ces descripteurs car, dans une *attitude*, l'information pertinente peut résider non pas dans la valeur spécifique du descripteur mais dans le fait que l'élève ne la modifie pas. Le descripteur *action.argument.négatif* en est un bon exemple : s'il reste toujours faux lorsque l'élève déplace des termes d'un membre à l'autre, cela montre que celui-ci ne maîtrise pas la règle du changement de signe dans les mouvements (ex :  $a=b \rightarrow a-b=0$ ). Néanmoins, tous les descripteurs d'action ne sont pas déductibles du *contexte* et du *produit*, c'est le cas du descripteur *action.transformation.juste* dont la valeur est indiquée dans le *diagnostic* établi par ANAIS.

### 3.3.3 Portées des descripteurs

Dans la description des éléments du triplet, certains descripteurs ont été dédoublés afin de prendre en compte la *localisation* de l'information qu'ils véhiculent. Nous distinguons ainsi dans notre représentation trois localisations<sup>3</sup> dans une formule :

- Argument : la partie de l'expression directement affectée par le mouvement.
- Terme : l'argument plus l'ensemble des éléments qui lui sont liés par des  $\times$  ou  $\div$ .
- Expression : ensemble de l'expression algébrique.

$$\begin{array}{c}
 \text{Expression} \\
 \overbrace{\hspace{10em}} \\
 \begin{array}{c}
 \text{Terme} \\
 \underbrace{-5x + 4 - \frac{7}{3}}_{\text{Argument}} = 9x^2 - 10 \quad \rightarrow \quad x + 4 - \frac{7}{3} = 9x^2 - 10 + 5
 \end{array}
 \end{array}$$

Figure 5 : Exemple de mouvement (incorrect) avec la description des différentes localisations.

### 3.3.4 Exemple de description d'un cas

Après différentes expérimentations, nous avons retenu 26 descripteurs (cf. Annexe A), dont certains sont utilisés pour décrire plusieurs localisations différentes. Initialement, le triplet représentatif d'un cas :  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$  comprend 37 descripteurs : 25 pour le *contexte*, 6 pour l'*action* et 6 pour le *produit*. La majorité de ces descripteurs sont de type booléen. La table 2 montre un exemple de vecteur correspondant à la description du cas décrit dans la figure 5.

Contexte		Action		Produit	
argument.coté	<i>gauche</i>				
argument.position	<i>début</i>				
argument.polynôme	<i>faux</i>				
argument.coefficient	<i>vrai</i>				
argument.signelmplicite	<i>faux</i>				
argument.opérateur	$\times$	argument.opérateur	<i>vrai</i>	argument.opérateur	$+$
argument.catégorie	<i>mult</i>	argument.catégorie	<i>vrai</i>	argument.catégorie	<i>add</i>
argument.négatif	<i>vrai</i>	argument.négatif	<i>vrai</i>	argument.négatif	<i>faux</i>
terme.polynôme	<i>vrai</i>				
expression.type	<i>equation</i>	expression.type	<i>faux</i>	expression.type	<i>équation</i>
expression.polynôme	<i>vrai</i>				
		transformation.juste	<i>faux</i>		

Table 2 : Exemple de description d'un cas. Afin de simplifier le tableau les descripteurs non significatifs du contexte (ceux dont la valeur est *faux* ou ceux non modifiés par l'action effectuée) ont été omis.

## 4 Construction des modèles

### 4.1 Introduction

La détermination des *attitudes* est effectuée par un *apprentissage non-supervisé* car nous n'avons pas de concept cible à apprendre<sup>4</sup>. La classification s'effectue *élève par élève* sur l'ensemble des cas qui lui sont associés, c'est-à-dire l'ensemble des mouvements qu'il a pu faire dans les exercices

<sup>3</sup> Notons que ce découpage est assez général car, dans tous types de problèmes, pour caractériser le passage d'un état du système à un autre par une action, on peut toujours définir trois groupes d'éléments : ceux directement modifiés (ici l'argument), ceux en relation avec les éléments modifiés (ici le terme) et l'environnement global (ici l'expression).

<sup>4</sup> Certes, on pourrait construire un classifieur permettant de déterminer les contextes dans lesquels les élèves obtiennent un résultat correct ou non mais, le but de notre étude n'est pas de prévoir le succès ou l'échec.

qu'il a traités. En moyenne, comme nous l'avons dit, les élèves ont traité entre 3 et 10 exercices ce qui permet à ANAIS d'extraire entre 50 et 100 *cas* pour chaque élève car il y a évidemment plusieurs mouvements algébriques effectués dans chaque exercice. Expérimentalement, cela s'est avéré suffisant pour faire émerger des attitudes dans lesquelles on retrouve un motif stable de triplet  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$  caractérisant les « réflexes mathématiques » de l'apprenant. Notons que l'on pourrait également envisager d'effectuer ce traitement au niveau d'un ensemble d'élèves de même niveau pour faire émerger un modèle collectif comme dans Bisson *et al.* (2003) ; c'est ce que nous étudions dans la partie §5.3, mais en travaillant à partir des attitudes individuelles produites.

Les *attitudes* ne correspondent pas nécessairement à des comportements qui ont été vus en cours. Elles pourront être soit justes, soit fausses, soit partiellement correctes selon les situations. De surcroît, il est important de tenir compte du fait que les êtres humains peuvent faire des erreurs d'inattention, ou même avoir un comportement qui semble « aléatoire », dans le sens où les actions effectuées peuvent varier selon le moment où il travaille avec APLUSIX. Du point de vue de l'apprentissage du modèle de l'élève, cela signifie que les *cas* étudiés présenteront un certain « bruit ». Ceci nous a conduit à introduire une dimension statistique dans le mécanisme de classification afin de pouvoir décider à partir de quel niveau un comportement devient suffisamment significatif pour constituer une *attitude* stable.

## 4.2 Classification des cas

Pour construire les *attitudes*, nous allons utiliser une méthode de type CAH (Classification Ascendante Hiérarchique, voir entre autres : Berkhin 2002, Jain *et al.* 1999) qui consiste à construire initialement une matrice de distances entre toutes les observations puis à agréger itérativement les observations (ici les *cas*) de manière à construire un arbre binaire de *classes* (ici les *attitudes*) partiellement ordonnées par une relation de généralité. L'avantage de cette approche est qu'elle produit différents niveaux de généralisation des *cas*, ce qui rend possible la visualisation (cf. figure 6) du degré de généralité des *attitudes* pour sélectionner celles que l'on souhaite retenir. L'utilisation de la CAH nécessite la définition préalable de deux distances :

- Une distance entre les observations (ex : distance de Minkowski, ...)
- Une distance entre les classes (ex : saut minimal/maximal, distance de Ward, ...)

Toutefois, nous n'avons besoin ici que d'une seule distance qui porte à la fois sur les *cas* et les *attitudes* car nous utilisons une méthode d'agrégation des classes spécifiques.

### 4.2.1 Méthode d'agrégation

Dans l'algorithme de CAH « standard », les classes construites n'ont pas de description explicite<sup>5</sup>. À chaque itération, le système regroupe les 2 instances (I, J) les plus proches voisines (observations et/ou classes) en une nouvelle classe H, puis il met à jour la matrice des distances entre H et les classes restantes  $R_k$  en l'évaluant à partir des distances existantes entre les  $R_k$  et les instances I et J. Dans notre travail, *cas* et *attitudes* ont une représentation commune et lors de l'agrégation on va calculer explicitement la distance entre la nouvelle attitude et les autres entités de la matrice<sup>6</sup>.

Dans la suite, chaque élève est représenté par un vecteur dont chaque composante correspond à une modalité possible d'un des descripteurs du triplet  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$  et ayant comme valeur le nombre d'occurrences de cette modalité. Pour un *cas* ce nombre est forcément soit 0 soit 1, mais dans une *attitude* (qui agrège des informations issues de plusieurs *cas*) la valeur est proportionnelle aux nombres de *cas* qui la constituent. Ainsi, nous conservons dans la description des attitudes les informations statistiques permettant d'associer un *degré de confiance* aux valeurs de modalité

<sup>5</sup> On peut évidemment construire une description en intention de la classe mais ce n'est qu'à des fins explicatives.

<sup>6</sup> Avec cette approche il est possible que les valeurs successives de distances trouvées lors de la classification ne soient plus monotones croissantes ce qui induit des inversions de nœuds dans la hiérarchie. Ceci est peu gênant en pratique.

observées et nous prenons en compte la variabilité des comportements décrits dans l'introduction (§ 5.1). Lorsque deux cas/attitudes sont agrégés, la nouvelle attitude est calculée en faisant la somme des occurrences de chacune des modalités (figure 6) :

Descripteurs	Cas 12				+	Attitude 6				=	Attitude (6&12)			
	gauche	droite				gauche	droite				gauche	droite		
Init.arg.coté	0	1			1	3			1	4				
Init.arg.position	début	milieu	fin	isolé	début	milieu	fin	isolé	début	milieu	fin	isolé		
	0	1	0	0	0	1	3	0	0	2	3	0		
Init.arg.complexe	vrai	faux			vrai	faux			vrai	faux				
	1	0			1	1			2	1				
...	...	...			...	...			...	...				

Figure 6: Exemple d'agrégation de deux cas/attitudes avec la remise à jour du nombre d'occurrences.

#### 4.2.2 Distance entre deux attitudes

Chaque attitude  $A_i$  est composée de trois termes : le contexte  $ctx(A_i)$ , l'action  $act(A_i)$  et le produit  $prd(A_i)$ . Ici, nous avons considéré les deux derniers termes comme étant « liés » car ils décrivent complètement la *transformation* réalisée par l'élève c'est à dire l'action effectuée et le résultat obtenu. Aussi la distance globale  $D_{Att}$  entre deux attitudes (ou cas) est la suivante :

$$D_{Att}(A_i, A_j) = \alpha \times dist((ctx(A_i), ctx(A_j))) + (1 - \alpha) \times (dist(act(A_i), act(A_j)) + dist(prd(A_i), prd(A_j)))$$

En faisant varier le paramètre  $\alpha$  on donne une importance plus grande soit au contexte soit à la transformation. Si le coefficient  $\alpha$  est petit ( $\alpha \approx 0,20$ ) le terme dominant est le poids relatif de la distance entre transformations, ce qui favorise le regroupement d'actions voisines même si les contextes sont différents. Ceci permet de mettre en évidence des *théorèmes en actes* (cf. §4.1). Inversement si le coefficient  $\alpha$  est grand on favorise la mise en évidence d'attitudes ayant des contextes similaires, c'est-à-dire d'*irrégularités*. Pour calculer la distance  $dist(V_i, V_j)$  entre deux vecteurs quelconque de données (c-à-d  $ctx(A_i)$ ,  $act(A_i)$  ou  $prd(A_i)$ ), nous comparons, descripteur par descripteur ( $d$ ), le profil de répartition fréquentiel des modalités ( $m$ ).

$$dist(V_i, V_j) = \sum_d \left( p_d \times \sum_m \left| \frac{\#V_{i,d,m}}{\sum V_{i,d}} - \frac{\#V_{j,d,m}}{\sum V_{j,d}} \right| \right) \quad \text{avec} \quad \begin{cases} \#V_{i,d,m} : \text{nombre d'occurrences de la modalité } m \text{ du descripteur } d \\ \sum V_{i,d} : \text{somme des occurrences des modalités du descripteur } d \end{cases}$$

Dans cette formule, on observe que chaque descripteur peut-être pondéré par un coefficient  $p_d$ . Dans nos expérimentations ces coefficients ont été ajustés de la manière suivante :

- l'attribut *Transformation.juste* n'est pas pris en compte ( $p=0$ ) car, ainsi que nous l'avons déjà dit, nous ne cherchons pas à construire des attitudes basées sur le fait que l'élève a réussi ou échoué. Toutefois comme on le verra par la suite, dans la classification obtenue cette distinction apparaît parfois sous l'influence des autres descripteurs.
- on met un fort poids sur *Expression.type* car on ne souhaite pas mélanger dans les attitudes les équations et les inéquations qui représentent des problèmes assez différents.
- hormis les attributs ci-dessus, tous les attributs de la partie transformation ont le même poids. Il en est de même dans la partie contexte sauf pour les attributs *Argument.opérateur* et *Argument.négatif* dont le poids a été augmenté après les premières expérimentations de manière à maintenir séparées, lors de la classification, les opérations (et signes) entre elles.

## 5 Résultats obtenus

### 5.1 Exemples d'attitudes individuelles

Cette approche classificatoire a été mise en œuvre sur l'ensemble des élèves de l'expérimentation effectuée au Brésil. La figure 7 présente un exemple de hiérarchie *d'attitudes individuelles* (élève avec l'identifiant 1497) avec une valeur  $\alpha=0,80$  (cf. § 2.5.2), ce qui correspond donc à privilégier la détection d'irrégularités, ce qui n'empêche pas l'identification de théorèmes en acte comme nous le verrons. Cet élève avait produit cinquante *cas*, et nous n'avons représenté dans cette figure que les *attitudes* ayant une distance non nulle. Afin de visualiser rapidement la signification des classes, les valeurs de certains attributs pertinents sont associées à des codes visuels (Table 3) :

Attribut graphique	Descripteur associé			
Couleur de fond	action.transformation.juste	vrai : vert/clair	faux : rouge/foncé	mixte : orange/moyen
Couleur du contour	action.argument.négatif	vrai : bleu/foncé	faux : jaune/clair	mixte : vert/moyen
Couleur du nombre	contexte.argument.négatif	vrai : bleu/foncé	faux : jaune/clair	mixte : vert/moyen
Forme du noeud	contexte.argument.opérateur	+ : triangle   - : carré   × : pentagone   ÷ : cercle		
Nombre de contours	action.argument.opérateur	+ : 1   - : 2   × : 3   ÷ : 4		
Orientation de la forme	contexte.expression.type	Equation : 0°		Inéquation : 45°
Anomalie dans le diagnostic d'ANAIS	10 traits de contour autour du noeuds			

Table 3 : Codes visuels utilisés pour représenter les attitudes dans la hiérarchie.

Comme toujours en classification hiérarchique, se pose le problème de l'identification des nœuds à conserver, ou en d'autres termes la définition du *seuil de coupure* dans l'arbre. Il s'est avéré expérimentalement que la valeur pertinente du seuil était relativement stable pour l'ensemble des élèves et qu'une distance de 38 constituait un bon compromis. Dans la figure 7, ce seuil conduit à l'identification de 5 attitudes différentes dont les modalités sont données dans la table 4.

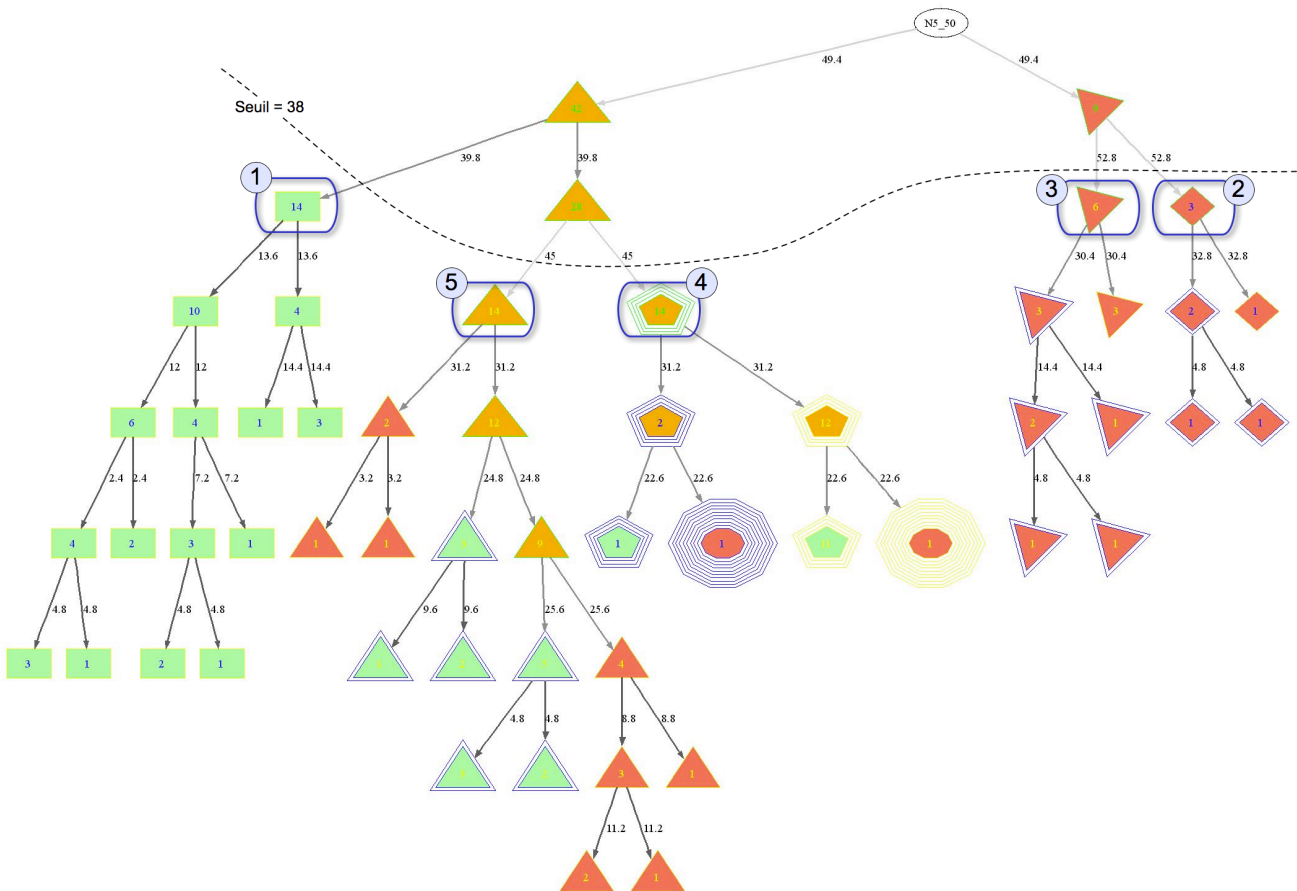


Figure 7: Classification des cas de l'élève id=1497 et numéro des cinq attitudes retenues. La représentation graphique est produite avec Graphviz (<http://www.graphviz.org/>). Les distances entre nœuds sont indiquées sur les arcs.

Dans cette table, les cases colorées correspondent à des descripteurs pour lesquels plusieurs modalités simultanées ont été repérées dans les attitudes (ce qui correspond à une généralisation), et afin de simplifier la notation nous n'y avons pas reporté le nombre d'occurrences associé à chaque modalité. Notons que la seconde attitude a des effectifs trop réduits (dernière ligne du tableau) pour qu'elle soit significative. Nous allons détailler la signification des attitudes 1 et 5.

L'attitude numéro 1 correspond à un comportement *correct*. Elle indique que cet élève maîtrise la résolution d'équations (*ctx.exp.type*) simples (*ctx.arg.racine*, *ctx.arg.puissance* et *ctx.arg.fraction* sont faux) dans lesquelles un argument négatif doit être déplacé (*ctx.arg.négatif*=vrai). Il s'agit par exemple de transformations comme :  $6x-3=2x+4 \rightarrow 6x=2x+4+3$ . L'élève déplace correctement cet argument de l'autre côté du signe égalité, quelle que soit sa position dans l'expression (*ctx.arg.côté* et *ctx.arg.position* sont généralisés) et la valeur du coefficient (*ctx.arg.coefficient*), en conservant la catégorie additive (*act.arg.catégorie*=faux) et en changeant le signe (*act.arg.signe*=vrai). C'est donc un *théorème en acte* : dans des contextes différents, l'élève applique toujours la même règle.

Descripteur	Attitude 1	Attitude 2	Attitude 3	Attitude 4	Attitude 5	
Contexte	Arg.côté	gauche, droit	gauche	gauche	gauche	gauche, droit
	Arg.position	début,fin,milieu,isolé	début,fin	fin	début	début,fin,milieu
	Arg.complexe	faux	faux	faux	faux	faux
	Arg.polynôme	vrai, faux	vrai, faux	faux	faux	vrai, faux
	Arg.racine	faux	faux	faux	faux	faux
	Arg.puissance	faux	faux	faux	faux	faux
	Arg.fraction	faux	faux	faux	faux	faux
	Arg.coefficient	vrai, faux	vrai	vrai	vrai	vrai, faux
	Arg.négatif	vrai	vrai	faux	vrai, faux	faux
	Arg.signeImplicite	faux	faux	faux	vrai, faux	vrai, faux
	Arg.opérateur	-	-	+	x	+
	Arg.catégorie	add	add	add	mult	add
	Terme.polynôme	vrai	vrai	vrai	vrai	vrai
	Terme.fraction	faux	faux	faux	faux	faux
	Terme.puissance	faux	faux	faux	faux	faux
	Terme.racine	faux	faux	faux	faux	faux
	Exp.type	équation	inéquation	inéquation	équation	équation
	Exp.sens	NA	gauche, droit	gauche, droit	NA	NA
	Exp.large	NA	faux	faux	NA	NA
	Exp.Polynôme	vrai	vrai	vrai	vrai	vrai
	Exp.Fraction	faux	faux	faux	faux	faux
	Exp.Puissance	faux	faux	faux	faux	faux
	Exp.Racine	faux	faux	faux	faux	faux
Exp.nulGauche	faux	faux	faux	faux	vrai, faux	
Exp.nulDroite	faux	faux	faux	faux	faux	
Action	Arg.Opérateur	vrai	vrai, faux	vrai, faux	vrai, faux	vrai, faux
	Arg.Catégorie	faux	faux	faux	faux	faux
	Arg.Signe	vrai	vrai, faux	vrai, faux	faux	vrai, faux
	Exp.Type	vrai	vrai	vrai	faux	faux
	Exp.Sens	NA	NA	NA	NA	NA
	Transformation.juste	faux	vrai	vrai	vrai, faux	vrai, faux
Produit	Arg.opérateur	+	+, -	+, -	÷	+,-
	Arg.catégorie	add	add	add	mult	add
	Arg.négatif	faux	vrai, faux	vrai, faux	vrai, faux	vrai, faux
	Exp.type	équation	équation	équation	équation	équation
	Exp.sens	NA	NA	NA	NA	NA
	Exp.large	NA	NA	NA	NA	NA
Effectifs	14	3	6	14	14	

Table 4 : Attitudes produites pour l'élève id=1497

L'attitude numéro 5 est en revanche *incohérente* car dans des contextes similaires, mais avec un argument positif (*ctx.arg.négatif*=faux), l'élève a un comportement moins systématique et il échoue parfois à changer le signe de l'argument. En pratique, il y a deux sous-attitudes qui ont été regroupées : l'une correspondant à un comportement *correct*, l'autre à un comportement *incorrect*, et c'est justement l'intérêt de la méthode que d'avoir explicité le fait que cet élève n'a pas un comportement identique face à des situations similaires. Il est toutefois possible comme nous le

verrons dans la section suivante § 6.2 de redescendre dans l'arbre jusqu'à la bifurcation entre une attitude généralisée correcte et une attitude généralisée incorrecte, et d'isoler les quelques descripteurs qui peuvent être la cause de ces comportements différents.

## 5.2 Exploitation des attitudes

Les *attitudes individuelles* ainsi générées peuvent avoir une double utilisation dans un EIAH. Tout d'abord, on peut les utiliser pour rappeler à l'élève quelle est la règle *correcte* qu'il aurait dû appliquer dans ce contexte, mais on peut surtout exploiter les valeurs des descripteurs afin d'engendrer des exercices permettant à l'élève de corriger une attitude *incorrecte* et/ou *incohérente*. Par exemple, dans l'attitude 5 ci-dessus, on utiliserait les valeurs des descripteurs pour construire un exercice contenant un argument positif à déplacer de l'autre côté de l'équation, tel :  $7x+4=11$ .

La seconde application de cette approche est à destination des enseignants ou des didacticiens qui sont intéressés à connaître les comportements récurrents des élèves. Il est clair que la représentation des attitudes sous la forme d'un vecteur de valeurs n'est pas directement lisible, aussi nous avons conçu un module qui convertit automatiquement ces informations en langue naturelle.

La méthode repose sur une simple concaténation de morceaux de textes instanciés par les valeurs des descripteurs. Lorsque les descripteurs ont été généralisés, c'est-à-dire qu'ils possèdent plusieurs modalités, le morceau de texte correspondant est omis. De surcroît, deux exemples génériques sont également engendrés pour illustrer l'attitude. Enfin, dans le cas d'une attitude représentant à la fois des modifications correctes et incorrectes, nous redescendons dans l'arbre comme indiqué précédemment afin d'indiquer les descripteurs qui discriminent les comportements *corrects* de ceux qui sont *incorrects*. Voici un exemple de texte correspondant à l'attitude 5 présentée plus haut. Les textes correspondant aux trois autres attitudes significatives : 1, 3 et 4 se trouvent dans l'annexe B).

```
Attitude numero 5, effectif 14.  
Correcte : 8 Incorrecte : 6.
```

```
Diagnostic :
```

```
Attitude incoherente consistant en un mouvement dans une equation d'un terme,  
initialement positif en position additive, ce mouvement s'effectuant avec ou  
sans changement de signe, la position finale etant additive.
```

```
Exemples :
```

```
r - v = nx -----> r - v - nx = 0  
v - r = n -----> v - r + n = 0
```

```
Explication :
```

```
L'eleve ne semble pas avoir une attitude coherente face a ce type de contexte,  
ce qui peut traduire une incomprehension plus profonde.  
On peut neanmoins chercher certains descripteurs de contexte qui pourraient  
permettre d'expliquer le comportement de l'eleve.
```

```
Les causes possibles de l'erreur sont :
```

```
-> l'argument se trouve du cote droit  
-> l'argument comporte une partie polynomiale  
-> l'argument comporte une partie numerique implicite  
-> le signe de l'argument est explicite  
-> l'argument initial est en derniere position  
-> le membre de gauche est nul
```

## 5.3 Recherche d'attitudes collectives

Dans les données expérimentales issues du Brésil, environ 1507 élèves ont traité suffisamment d'exercices pour que l'on puisse engendrer des *attitudes collectives* statistiquement significatives les concernant. Aussi, nous avons étudié la répartition globale de ces *attitudes individuelles* afin d'examiner si certaines d'entre elles étaient partagées par plusieurs élèves et, si oui, lesquelles.



Pour effectuer cette étude, il est impossible d'établir directement un histogramme de fréquences de chaque attitude. En effet, celles-ci sont produites par induction à partir des traces de comportement des élèves, dès lors, même si deux attitudes expriment globalement les mêmes informations, on va observer de légères variations dans le nombre d'occurrence des modalités ce qui fait que les attitudes vont, presque toutes, être deux à deux différentes. Il est donc nécessaire de procéder à une phase initiale d'agrégation des attitudes individuelles en utilisant, comme précédemment, un mécanisme de classification ascendante hiérarchique. Cette opération est facilement réalisable puisque les cas et les attitudes partagent la même représentation (cf. § 4.2.1). Pour regrouper les 11026 attitudes engendrées par les 1507 élèves nous avons utilisé les paramètres suivants :

- La *pondération des attributs* est identique à celle utilisée pour classifier les cas (cf. § 4.2.2).
- Le *paramètre  $\alpha$*  a pour valeur 0,20 à la fois lors de la construction des attitudes individuelles et lors de leur agrégation. Nous avons en effet recherché ici à faire apparaître de manière privilégiée des *théorèmes en actes*, c'est à dire des comportements dans lesquels les élèves effectuent des transformations similaires dans contextes différents.
- Le *seuil de coupure* dans l'arbre est fixé à 10 alors qu'il était égal à 38 pour la construction des attitudes individuelles. Ce seuil est relativement bas car le but ici est de « lisser » les différences entre des attitudes individuelles ; en les généralisant trop on obtiendrait des attitudes qui ne pourraient plus être associée à un comportement précis.

Les principales *attitudes collectives* produites sont présentées dans la figure 7. Nous n'avons conservé que les attitudes qui regroupent un minimum de 100 cas initiaux. La description de toutes les attitudes discutées ci-dessous se trouve dans l'annexe C de ce rapport.

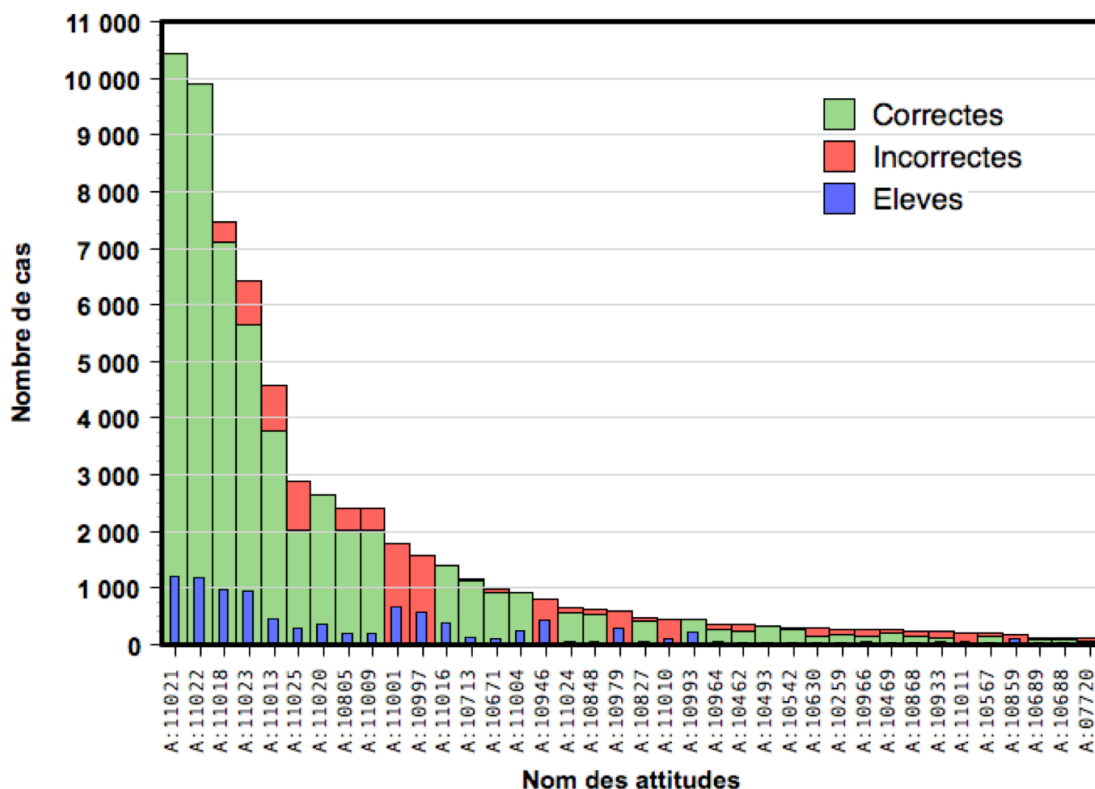


Figure 7: Histogramme des 38 attitudes les plus fréquentes engendrées avec la base issue des expérimentations effectuées au Brésil. Pour chaque attitude sont indiqués le nombre de cas corrects et incorrects qui la composent ainsi que le nombre d'élèves chez lesquels cette attitude a été observée.

Les deux attitudes les plus fréquentes, qui sont partagées par plus de 80% des élèves, correspondent à une application toujours *correcte* des règles de mouvements dans les équations lorsque l'argument est positif (*attitude 11021* - 1217 élèves) ou négatif (*attitude 11022* - 1188 élèves).

Inversement des attitudes systématiquement *incorrectes* apparaissent également. Par exemple, lorsque l'argument à déplacer est dans une inéquation (*attitude 10997* - 565 élèves), beaucoup d'élèves ne changent pas le signe de l'argument. De manière plus générale, on observe que les mouvements dans les inéquations sont globalement moins bien maîtrisés que dans les équations, ce qui conduit à la création d'attitudes mixtes (*attitude 11023* - 955 élèves) ou les élèves changent bien le signe mais également parfois le sens de l'inéquation ce qui serait révélateur d'une confusion avec les règles de mouvement portant sur les produits et quotients.

Certaines des attitudes incorrectes produites (*attitude 11001* - 678 élèves) montrent cependant une limite de notre approche. L'attitude *11001* est la version duale de l'attitude correcte *11021* décrite ci-dessus, aussi on s'attendrait à ce que les effectifs en termes d'élèves soient complémentaires, ce qui n'est pas le cas. En effet, 388 d'entre eux ont simultanément une attitude *toujours correcte* et une attitude *toujours incorrecte*, ce qui est peu logique. La cause de ce problème provient de la faible valeur  $\alpha$  que nous avons utilisée lors de la classification et qui a conduit le système à généraliser faiblement les *transformations* effectuées et donc à former 2 attitudes distinctes. Augmenter la valeur  $\alpha$  réglerait ce problème au risque de faire apparaître des attitudes dont la conclusion serait trop générale pour être interprétable. Une amélioration de notre méthode passe sans doute par une détermination plus automatique du paramètre  $\alpha$  et/ou la détection et l'élimination de ces attitudes contradictoires.

## 6 Conclusion

Dans ce rapport, nous avons présenté une méthode permettant d'analyser automatiquement des comportements d'élèves, sous la forme d'un triplet  $\langle \text{contexte}, \text{action}, \text{produit} \rangle$  caractérisé par un ensemble de descripteurs, afin de construire des profils exprimés sous la forme d'*attitudes*. Dans son développement actuel, cette approche offre plusieurs propriétés intéressantes :

- Le résultat est directement interprétable par un expert et apporte aux didacticiens une vision globale et synthétique de l'état des connaissances d'un élève.
- Le système sait prendre en compte les connaissances contradictoires, représentées sous la forme d'irrégularités, et permet de faire la distinction entre le *régulier* et l'*occasionnel*.
- Les diagnostics effectués dans le domaine de l'algèbre peuvent être facilement utilisés pour paramétrer un générateur d'exercices destiné à entraîner les élèves.
- La méthode proposée permet de regrouper facilement les *attitudes individuelles* afin de construire un diagnostic général du groupe d'élèves étudié via les *attitudes collectives*.

L'un des avantages de ce modèle de l'élève réside dans sa dimension cognitive car il représente la logique propre à l'élève et non la justesse mathématique de ses actions. Ceci rejoint les avis de didacticiens (Dillenbourg *et al.*, 1992, Balacheff 1992), qui préfèrent la vision d'une utilisation rationnelle, mais non appropriée, de processus cognitifs à la notion d'« erreur de connaissance ». Ainsi, les théorèmes-en-acte de statut mixte sont le signe de connaissances qui peuvent être erronées, mais qui ont par ailleurs un domaine de validité : utiles dans certaines situations particulières, ils peuvent se révéler inadaptés dans un autre contexte. D'un point de vue qualitatif, les modèles produits mettent en évidence des compétences et des erreurs plutôt classiques, que les enseignants connaissent bien. Ceci dit, il n'en reste pas moins intéressant qu'un processus automatique conduise à des conclusions congruentes avec celles de l'expert.

Le domaine d'application étudié est limité ici aux *mouvements algébriques* dans les équations et inéquations, mais nous envisageons, à très court terme, l'extension de ces possibilités. En effet, l'équipe Did@tic du laboratoire Leibniz-IMAG a récemment développé un nouvel ensemble de règles pour ANAIS permettant l'identification et l'interprétation automatiques des étapes de *développements* et de *réductions* d'expressions. Comme pour les mouvements, ces règles semblent caractériser de manière satisfaisantes plus de 80% des productions d'élèves. Pour analyser ces

nouveaux cas il est bien sûr nécessaire d'introduire de nouveaux descripteurs dans notre triplet <contexte, action, produit>. Toutefois, une étude préliminaire semble montrer que cette tâche est réalisable, les descripteurs supplémentaires étant qualitativement et quantitativement très semblables à ceux qui sont actuellement utilisés.

Plus fondamentalement, l'une des limites de notre approche réside dans le fait que les comportements d'élève sont considérés indépendamment du temps. Il serait donc intéressant d'essayer de modéliser la *trajectoire d'apprentissage* entière de l'élève depuis ses premiers exercices, en examinant comment les différentes attitudes apparaissent, disparaissent et se stabilisent. Une piste pour faire ce travail serait d'utiliser le mécanisme de classification incrémental COBWEB (Fisher, 1987) auquel on soumettrait une séquence chronologique de cas venant d'un élève donné. En étudiant l'évolution de la hiérarchie d'attitudes produites on pourrait établir un *chronogramme d'attitudes* dans lequel apparaîtraient et disparaîtraient des *théorèmes en actes* et des *irrégularités* au fur et à mesure que l'élève affine ses connaissances.

Enfin, on peut s'interroger sur la possibilité d'appliquer notre méthode de génération de profils à d'autres domaines d'application, ce qui n'est pas toujours une chose facile dans les EIAH. On peut souligner que nous avons introduit relativement peu de mécanismes *ad hoc*. Certes les descripteurs utilisés sont spécifiques au domaine (comment ne pas l'être d'ailleurs !), cependant les concepts généraux utilisés ici, tels : la notion d'*observable* correspondant à une étape élémentaire du traitement effectué par l'élève, la représentation des *cas* par des triplets <contexte, action, produit>, le mécanisme de classification et la production d'attitudes (et leur explication) restent indépendants du domaine traité. On peut donc considérer que la portabilité de notre approche est conditionnée à la définition de plusieurs critères.

D'une part, il est nécessaire que l'on puisse segmenter la trace des comportements d'élèves sous la forme d'une série de transition entre un *état initial* et un *état final* et que l'on puisse associer à chacune de ses transitions un *diagnostic* indiquant le type de l'opération effectuée ainsi que sa correction vis-à-vis du domaine. D'autre part, il est nécessaire que l'on puisse encoder ces informations sous la forme de triplets <contexte, action, produit> pour construire les modèles.

Enfin, il est important de noter que les comportements enregistrés par l'EIAH doivent posséder une granularité permettant d'effectuer leur analyse. Ainsi, il faut une certaine variabilité dans les observations pour engendrer suffisamment de descripteurs : un EIAH dans lequel l'élève rentrerait directement des entiers en réponse à des questions de calcul ne conviendrait pas. Mais inversement, un EIAH qui ne fournirait comme trace de comportement que des positions de clics de souris ne permettrait pas d'engendrer un diagnostic pertinent.

## **Remerciements**

Nous remercions chaleureusement tous les membres de l'équipe Did@tic du laboratoire Leibniz-IMAG travaillant sur le projet APLUSIX qui nous ont permis de réaliser ce travail en nous fournissant les données et surtout en nous apportant l'expertise didactique nécessaire. Ce travail a été partiellement financé via l'ACI « Cognitive » du Ministère de la Recherche. Nous remercions également David Renaudie dont la thèse a servi de base pour effectuer ce travail.

## 7 Références

- BISSON G., BRONNER A., GORDON M.B., NICAUD J.F., RENAUDIE D. (2003) : *Analyse statistique de comportements d'élèves en algèbre*. Actes de la conférence EIAH'2003 (Strasbourg), pp. 67-78.
- BALACHEFF N. (2002) : Contribution à la réflexion sur la recherche sur les EIAH. Les Technologies en éducation : perspectives de recherche et questions vives, Baron G.-L. & Bruillard E. Eds, Paris.
- BECK, J.E., WOOLF, B.P (2000). High-level Student Modeling with Machine Learning, Proceedings of the Intelligent Tutoring Systems Conference, 584-593.
- BERKHIN, P. (2002). Survey of clustering data mining techniques. Tech. rep., Accrue Software, San Jose, CA. <http://citeseer.nj.nec.com/berkhin02survey.html>
- DILLENBOURG P., SELF J.A. (1992) : A framework for learner modelling. Interactive Learning Environments Vol 2, pp 111-137.
- ESPOSITO, F., LICCHELLI O., SEMERARO, G. (2004). Discovering Student Models in e-learning Systems. Journal of Universal Computer Science 10(1), 47-57.
- FISHER, D.H. (1987), Knowledge Acquisition via Incremental Conceptual Clustering, Machine Learning 2:139-172
- JAIN A.K., MURTY M.N., FLYNN P.J. (1999). Data Clustering: A Review. ACM Computing Surveys, vol. 31, pages 265-322, September 1999
- NICAUD J-F, BOUHINEAU D., HUGUET T. (2002). « The Aplusix-Editor: A New Kind of Software for the Learning of Algebra » . Proceedings of the 6th International Conference on Intelligent Tutoring Systems. Lecture Notes In Computer Science, Springer Vol. 2363. Biarritz, France and San Sebastian, Spain, June 2-7, 2002 Pages: 178 – 187
- MOSTOW, J. (2004). Some useful design tactics for mining ITS data, Proceedings of the ITS2004 Workshop on Analyzing Student-Tutor Interaction Logs to Improve Educational Outcomes, August, 2004.
- RENAUDIE D (2005). Méthodes d'apprentissage automatique pour la modélisation de l'élève en algèbre. Thèse de l'INPG soutenue le 14 janvier 2005.  
<http://www-leibniz.imag.fr/Apprentissage/Publications/Renaudie-2005-These.pdf>
- SISON R., SHIMURA, M. (1998). Student Modeling and Machine Learning. International Journal of AI and Education 9, 128-158.
- TSIRIGA V., VIRVOU, M. (2002). Initializing the Student Model using Stereotypes and Machine Learning, In Proceedings of the 2002 IEEE International Conference on Systems, Man and Cybernetics.
- VERGNAUD G. (2001) : *Forme opératoire et forme prédicative de la connaissance*. Actes du colloque GDM 2001, Portugais J. Eds, pp 6-27.

**Annexe A** : Liste des attributs utilisés pour représenter les « mouvements ».

Cette table indique quels sont les descripteurs qui sont utilisés (●) ou non (-) pour la caractérisation du *contexte* (Ctx), de l'*action* (Act) et du *produit* (Prd) effectuée. Les attributs qui apparaissent dans la partie *action* sont tous de type booléens et, à l'exception de *action.transformation.juste*, sont calculés à partir de la valeur des descripteurs communs au *contexte* et au *produit*.

Attribut	Domaine (Ctx et Prd)	Ctx	Act	Prd	Description
<i>Description de l'argument sur lequel porte le mouvement</i>					
Argument.coté	{gauche, droite}	●	-	-	Place de l'argument dans l'expression
Argument.position	{début, milieu, fin, isolé}	●	-	-	Sa position dans le membre, la valeur « isolé » signifiant que le membre ne contient que lui.
Argument.complexe	{vrai, faux}	●	-	-	Présence d'un opérateur + ou -, ex : • $2x$ (Argument.complexe = faux) • $(3x+8)$ (Argument.complexe = vrai)
Argument.polynôme	{vrai, faux}	●	-	-	Présence d'une variable, ex : $2x$
Argument.racine	{vrai, faux}	●	-	-	Présence d'une racine, ex : $\sqrt{3x}$
Argument.puissance	{vrai, faux}	●	-	-	Présence d'un exposant, ex : $x^2$
Argument.fraction	{vrai, faux}	●	-	-	Présence d'une fraction, ex : $-3/4$
Argument.coefficient	{vrai, faux}	●	-	-	Coefficient numérique explicite, ex : $3x$
Argument.négatif	{vrai, faux}	●	●	●	Argument de signe négatif, ex : $-4$
Argument.signeImplicite	{vrai, faux}	●	-	-	Signe de l'argument implicite, ex : $(+)4x$
Argument.opérateur	{+, -, ×, ÷, ^, √}	●	●	●	Opérateur de priorité max. dans l'argument
Argument.catégorie	{add, mult, puiss}	●	●	●	Catégorie de cet opérateur : • add={+, -} ; mult={×, ÷} ; puiss={^, √}
<i>Description du terme impliqué par le mouvement.</i>					
Terme.polynôme	{vrai, faux}	●	-	-	Même signification que pour l'argument
Terme.fraction	{vrai, faux}	●	-	-	idem
Terme.puissance	{vrai, faux}	●	-	-	idem
Terme.racine	{vrai, faux}	●	-	-	idem
<i>Description de l'expression</i>					
Expression.type	{équation, inéquation}	●	●	●	Type de l'expression
Expression.sens	{gauche, droite, NA}	●	●	●	Orientation de l'inéquation (NA si équation)
Expression.large	{vrai, faux, NA}	●	-	●	Inégalité large ou stricte (NA si équation)
Expression.Polynôme	{vrai, faux}	●	-	-	Même signification que pour l'argument
Expression.Fraction	{vrai, faux}	●	-	-	idem
Expression.Puissance	{vrai, faux}	●	-	-	idem
Expression.Racine	{vrai, faux}	●	-	-	idem
Expression.nulGauche	{vrai, faux}	●	-	-	Indique si le membre gauche = 0
Expression.nulDroite	{vrai, faux}	●	-	-	Indique si le membre droit = 0
<i>Information fournie par ANAIS</i>					
Transformation.juste	{vrai, faux}	-	●	-	indique la validité du mouvement

## Annexe B : Exemples de diagnostics automatiques individuels

Attitude numero 1, effectif 14.  
Correcte : 14 Incorrecte : 0.

Diagnostic :  
Attitude correcte consistant en un mouvement dans une equation d'un terme, initialement negatif en position additive ce mouvement s'effectuant avec changement de signe , la position finale etant additive.

Exemples :  
 $r - v = - nx \text{ -----} \rightarrow r - v + nx = 0$   
 $v - r = - n \text{ -----} \rightarrow v - r + n = 0$

Explication :  
L'eleve a adopte une attitude a la fois correcte et coherente pour ce type de contexte, ce qui correspond a la connaissance et l'application d'une regle juste.

Attitude numero 3, effectif 6.  
Correcte : 0 Incorrecte : 6.

Diagnostic :  
Attitude incorrecte consistant en un mouvement dans une inequation au sens stricte d'un terme , initialement positif en position additive ce mouvement s'effectuant avec changement du type de l'equation avec ou sans changement de signe , la position finale etant additive.

Exemples :  
 $n \leq t - w \text{ -----} \rightarrow 0 = t - w - n$   
 $n > w - t \text{ -----} \rightarrow 0 = w - t + n$

Explication :  
L'eleve a adopte une attitude incorrecte, mais neanmoins coherente, ce qui correspond a l'application rigoureuse d'une regle fausse.

Attitude numero 4, effectif 14.  
Correcte : 12 Incorrecte : 2.

Diagnostic :  
Attitude incoherente consistant en un mouvement dans une equation d'un terme , initialement positif ou negatif en position multiplicative ce mouvement s'effectuant sans changement de signe , la position finale etant divisive.

Exemples :  
 $r - v = n*x \text{ -----} \rightarrow (r - v) / (n) = x$   
 $v - r = - n*x \text{ -----} \rightarrow (v - r) / (n) = x$

Explication :  
Le rapport des effectifs corrects-incorrecs etant tres disequilibre, on peut considerer cette attitude comme etant coherente.

## Annexe C : Exemples de diagnostics automatiques collectifs

Attitude numero 11022, effectif 9888.  
Correcte : 9888 Incorrecte : 0.

Diagnostic :

Attitude correcte consistant en un mouvement dans une equation d'un terme ,  
initialement negatif en position additive ce mouvement s'effectuant avec  
changement de signe , la position finale etant additive.

Exemples :

- nx = t - w -----> 0 = t - w + nx  
v - r = - n -----> v - r + n = 0

Explication :

L'eleve a adopté une attitude a la fois correcte et coherente pour ce type de  
contexte, ce qui correspond a la connaissance et l'application d'une regle  
juste.

Attitude numero 11021, effectif 10444.  
Correcte : 10444 Incorrecte : 0.

Diagnostic :

Attitude correcte consistant en un mouvement dans une equation d'un terme ,  
initialement positif en position additive ce mouvement s'effectuant avec  
changement de signe , la position finale etant additive.

Exemples :

r - v = nx -----> r - v - nx = 0  
v - r = n -----> v - r - n = 0

Explication :

L'eleve a adopté une attitude a la fois correcte et coherente pour ce type de  
contexte, ce qui correspond a la connaissance et l'application d'une regle  
juste.

Attitude numero 11001, effectif 1795 (678).  
Correcte : 0 Incorrecte : 1795.

Diagnostic :

Attitude incorrecte consistant en un mouvement dans une equation d'un terme ,  
initialement positif en position additive ce mouvement s'effectuant sans  
changement de signe , la position finale etant additive.

Exemples :

nx = t - w -----> 0 = t - w + nx  
v - r = n -----> v - r + n = 0

Explication :

L'eleve a adopte une attitude incorrecte, mais neanmoins coherente, ce qui  
correspond a l'application rigoureuse d'une regle fausse.

Attitude numero 10997, effectif 1566 (565).  
Correcte : 0 Incorrecte : 1566.

Diagnostic :

Attitude incorrecte consistant en un mouvement dans une inequation d'un terme , initialement positif en position additive ce mouvement s'effectuant avec ou sans changement de sens de l'inequation sans changement de signe , la position finale etant additive.

Exemples :

$nx < t - w \text{ -----} > 0 < t - w + nx$   
 $n \geq w - t \text{ -----} > 0 \geq w - t + n$

Explication :

L'eleve a adopte une attitude incorrecte, mais neanmoins coherente, ce qui correspond a l'application rigoureuse d'une regle fausse.

Attitude numero 11023, effectif 6423.  
Correcte : 5645 Incorrecte : 778.

Diagnostic :

Attitude incoherente consistant en un mouvement dans une inequation d'un terme , initialement negatif en position additive ce mouvement s'effectuant avec ou sans changement de sens de l'inequation avec changement de signe , la position finale etant additive.

Exemples :

-  $nx < t - w \text{ -----} > 0 < t - w + nx$   
-  $n \geq w - t \text{ -----} > 0 \geq w - t + n$

Explication :

L'eleve ne semble pas avoir une attitude coherente face a ce type de contexte, ce qui peut traduire une incomprehension plus profonde.



**L**e laboratoire Leibniz est fortement pluridisciplinaire. Son activité scientifique couvre un large domaine qui comprend aussi bien des thèmes fondamentaux que des thèmes très liés aux applications, aussi bien en mathématiques qu'en informatique.

Les recherches sur les Environnements Informatiques d'Apprentissage Humain et la didactique des mathématiques ouvrent cette pluridisciplinarité sur les sciences humaines, elles jouent un rôle particulier en favorisant les coopérations entre différentes composantes du laboratoire.

- \* mathématiques discrètes et recherche opérationnelle
- \* logique et mathématique pour l'informatique
- \* informatique de la connaissance
- \* EIAH et didactique des mathématiques

*Les cahiers du laboratoire Leibniz* ont pour vocation la diffusion des rapports de recherche, des séminaires ou des projets de publication réalisés par des membres du laboratoire. Au-delà, Les cahiers peuvent accueillir des textes de chercheurs qui ne sont pas membres du laboratoire Leibniz mais qui travaillent sur des thèmes proches et ne disposent pas de tels supports de publication. Dans ce dernier cas, les textes proposés sont l'objet d'une évaluation par deux membres du Comité de Rédaction.

**Comité de rédaction :** Nicolas Balacheff, Mirta Gordon, Denise Grenier, Yves Demazeau, Andras Sebo

Contact Gestion & Réalisation : Jacky Coutin  
Directeur de la publication : Nicolas Balacheff  
ISSN : 1298-020X - © laboratoire Leibniz