HAL
open science

Colloque Intelligence Artificielle, 16-18 septembre 1997

Jacques Pitrat, Gérard Tisseau, Régis Moneret, Arnaud Fredon, Jacques Duma, Hélène Giroire, Françoise Le Calvez, Marie Urtasun, Michel Masson

To cite this version:

Jacques Pitrat, Gérard Tisseau, Régis Moneret, Arnaud Fredon, Jacques Duma, et al.. Colloque Intelligence Artificielle, 16-18 septembre 1997. Kornman,Sylvie. Colloque Intelligence Artificielle, Sep 1997, Ile de Berder, France. lip6.1998.007, LIP6, 1998. hal-02549576

HAL Id: hal-02549576

https://hal.science/hal-02549576v1

Submitted on 21 Apr 2020

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.

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Résumé

Ce rapport contient les textes de quelques unes des interventions faites au colloque Intelligence Artificielle à l'île de Berder du 16 au 18 septembre 1997.

Mots-clés

Intelligence artificielle, emotion, EIAO, raisonnement temporaire

Abstract

These proceedings include 6 papers selected from the workshop on Artificial Intelligence that was held on Berder island, on September 16-18, 1997.

Keywords

Artificial intelligence, emotion, EIAO, temporal raisoning

Actes réunis par Sylvie Kornman, LAMSADE, Université Paris 9 Dauphine

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Sommaire

Présentation

Jacques Pitrat 1

Qu'est-ce qu'une émotion ?

Gérard Tisseau 3

Les Emotions Mécaniques

Régis Moneret 15

Les émotions : quels intérêts pour l'homme ?

Arnaud Fredon 25

Reconnaître les émotions des êtres humains

Jacques Pitrat 33

Les "machines à construire" : des interfaces dans le projet COMBIEN?

Groupe COMBIEN 47

Connaissances Temporelles et Raisonnement

Michel Masson 77

L'équipe Métaconnaissance du LIP6 ainsi que plusieurs chercheurs d'EDF se sont réunis dans l'île de Berder du 16 au 18 septembre 1997. Ce rapport contient les textes de quelques unes des interventions qui y ont été faites.

Ce rapport commence par quatre papiers qui sont une réflexion et un état de l'art sur les émotions dans les systèmes d'intelligence artificielle. Les psychologues sont devenus conscients de l'importance de l'interaction des émotions avec la cognition pure. Elles se révèlent en particulier avoir une très grande influence sur les performances des sujets humains en résolution de problèmes. Si leur effet peut se révéler négatif dans certaines circonstances, il peut être aussi bénéfique, notamment pour mieux mobiliser le potentiel d'un humain qui fait face à une crise grave. Il pourrait donc être intéressant qu'un système informatique puisse "ressentir" des émotions, en particulier s'il s'agit d'un système travaillant dans des conditions critiques. Il peut être également intéressant pour un système d'analyser et de comprendre les émotions ressenties par les agents — notamment humains — avec lesquels il entre en interaction.

Le papier suivant décrit les travaux du groupe *Combien?* Son but est de construire un système destiné à l'enseignement, focalisé sur les problèmes de dénombrement en mathématiques. Un tel système doit allier des possibilités en résolution de problèmes et en pédagogie, et proposer aux élèves et aux professeurs une interface pédagogique.

Nous avons enfin une synthèse sur un problème aussi important que difficile à résoudre : comment lier le raisonnement et les connaissances temporelles. Les contraintes liées au temps interviennent dans un grand nombre de problèmes ; malheureusement, il n'est pas facile de les prendre en compte de façon satisfaisante.

Nous remercions Christophe Turle et Régis Moneret pour s'être si bien occupés de l'organisation matérielle de ce colloque qui a pu avoir lieu grâce à un contrat EDF. Merci aussi à Sylvie Kornman qui a parfaitement pris en main l'édition de ce rapport.

Jacques Pitrat

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Qu'est-ce qu'une émotion ?
Etude des tentatives de définition

Gérard Tisseau

LIP6
Université Pierre et Marie CURIE

Résumé : Le concept d'émotion a fait l'objet de nombreuses tentatives de définition dans des disciplines diverses. Les méthodes utilisées pour aborder ce concept peuvent consister par exemple à décrire et mesurer des phénomènes physiologiques liés aux émotions, à étudier des textes parlant des émotions, à observer les corrélations entre les émotions et des fonctions mentales, et à identifier des zones du cerveau liées aux émotions. L'article passe rapidement en revue différentes études concernant ces aspects.

Mots-clés : émotion, définition des émotions

Le but de cet article est de faire le point sur les difficultés que présente toute tentative de définition du concept d'émotion et de décrire certaines de ces tentatives. La plupart des informations qui figurent ici sont extraites des sources suivantes : [Dantzer 94], [Goleman 97], [Damasio 95], qui ne seront pas citées systématiquement à chaque occurrence.

1. Qu'est-ce qu'une émotion ?

On manque de définition précise et communément acceptée du concept d'émotion. Une des raisons de ce manque de consensus est que ce concept a été étudié avec des méthodes diverses par des disciplines ayant des préoccupations différentes : la philosophie, la biologie, la psychologie, la linguistique. De plus, comme il s'agit de phénomènes ressentis personnellement par chacun d'entre nous, nous sommes tous persuadés que nous savons ce qu'est une émotion. Le langage courant n'a pas attendu les études systématiques pour introduire de nombreux termes de sens commun liés aux émotions, sans nécessairement se soucier de justifications théoriques ou expérimentales.

Devant cette diversité, il est difficile de proposer ici une définition de référence. On peut quand même remarquer un lien étroit entre le terme “émotion” et le terme “ressentir”, lien qui semble être pris en compte par toutes les approches. Plus précisément, on peut énoncer un critère d’élimination pour délimiter le concept :

“ Si ça ne se ressent pas, ce n’est pas une émotion ”

Encore faut-il préciser que l’acte de ressentir n’est pas nécessairement entièrement conscient. La réciproque du critère est moins assurée : tout ce qui se ressent n’est pas forcément considéré comme une émotion.
Ce lien permet de préciser la catégorie conceptuelle dans laquelle on peut ranger le concept d’émotion : il s’agit d’un phénomène se produisant au sein d’un système (l’être humain en l’occurrence, mais aussi plus généralement un être vivant et pourquoi pas un système artificiel).

1.1. Une émotion est un phénomène physique et mental

Une émotion est un phénomène ayant à la fois des aspects physiques (coeur battant, mains moites) et des aspects mentaux (j’aime, je n’aime pas). Relativement à ce dernier aspect, il est qualifié de phénomène *affectif* (en opposition à *cognitif*). On distingue habituellement différents types de phénomènes affectifs, l’un d’entre eux étant les émotions et d’autres étant par exemple les “sentiments” et les “passions”. Les définitions des différents types sont floues et controversées car elles proviennent d’approches assez différentes, comme la philosophie, la biologie et la psychologie.

Le phénomène peut être étudié d’un point de vue statique ou d’un point de vue dynamique. Dans le premier cas, une émotion est vue comme un *état* (je suis en colère) et dans le second elle est vue comme un *processus* (cela me met en colère, ma colère retombe).

1.2. L'opposition émotion / raison

Les émotions sont souvent opposées à la raison, comme l’illustrent les couples antinomiques suivants : affectif / cognitif, impulsif / réfléchi, le coeur / la tête, ressentir / penser.
Certains faits biologiques semblent soutenir cette vision : l’étude du cerveau et de son évolution suggèrent l’existence de trois parties différentes, associées respectivement aux instincts, aux émotions et à la raison. Cette vision est contestée comme étant trop schématique quant au rôle attribué à chaque partie, mais on reconnaît généralement qu’il y a bien trois sous-systèmes distincts interagissant. Le sous-système “rationnel” est apparu bien après le sous-système “émotionnel” dans l'évolution et l’a prolongé sans le remplacer.

2. Phénomènes observables liés aux émotions

Une première approche pour essayer d'identifier les émotions est d'étudier les phénomènes corrélés, les plus faciles à aborder étant les phénomènes physiologiques. Avant de relier ces phénomènes aux émotions, la première difficulté est de les mesurer et de les décrire.

2.1. Expressions faciales

Il est bien connu que le visage peut refléter fidèlement les émotions. C'est une des bases du jeu des acteurs par exemple. On a essayé de caractériser l'expression faciale (statique) à l'aide de différents critères : l'état des muscles faciaux, la position de la tête et des yeux. On a également essayé de décrire l'aspect dynamique (changements d'apparence). Ces tentatives ont introduit des systèmes de codage (Facs, Max, Affex) fondés sur des critères précis et aussi objectifs que possible. Une expression est codée par un vecteur d'attributs (de trente à quarante). Le codage est effectué par un observateur humain et la méthode semble assez objective puisque la concordance entre observateurs entraînés est de l'ordre de 80%.

2.2. Indices vocaux

La voix peut exprimer des émotions par l'intermédiaire de paramètres acoustiques comme la fréquence, l'intensité, le tempo, la tonalité. Ce mode d'expression caractérise mieux certaines émotions que d'autres : la tristesse et la colère sont bien reconnaissables (au moins 90% d'accord lors des expériences), alors que la peur et la joie le sont moins (50% d'accord seulement).

2.3. Manifestations viscérales

D'autres phénomènes liés aux émotions sont moins immédiatement perceptibles de l'extérieur : il s'agit des manifestations viscérales. Le mot "viscéral" est d'ailleurs associé à quelque chose qu'on ressent de l'intérieur de manière immédiate, irraisonnée. Différents critères sont utilisés (le "détecteur de mensonges" utilise ce genre de mesures) : la fréquence cardiaque, la répartition du sang (rouge de colère, blanc de peur), la conductance et la température de la peau, le rythme de la respiration, la tension des muscles.
Par exemple, on demande à des acteurs de mimer une expression et on mesure certains paramètres. Le bonheur, le dégoût et la surprise correspondent à une fréquence cardiaque faible, la peur et la tristesse à une fréquence cardiaque élevée et à une température cutanée faible, la colère à une fréquence cardiaque élevée et à une température cutanée élevée (on s'échauffe).

En ce qui concerne le mode respiratoire, le rire de la joie consiste en une inspiration suivie de saccades expiratoires, les pleurs de la tristesse en saccades inspiratoires suivie d'un soupir. La colère se manifeste par une augmentation de fréquence et d'amplitude de la respiration, la tendresse par une basse fréquence avec des pauses expiratoires, la peur par une respiration irrégulière de faible amplitude, de haute fréquence, avec des apnées.

2.4. Manifestations hormonales

A un niveau encore moins perceptible et plus interne, on trouve des manifestations hormonales, concernant essentiellement trois types d'hormones :
— les catécholamines (adrénaline, noradrénaline), qui sont liées à la mobilisation en cas d'urgence ;
— les glucocorticoïdes (ACTH), contrôlant la réaction d'adaptation au stress ;
— les opioïdes endogènes (endorphine), associées à la sensibilité à la douleur et à la régulation thermique.
Ces manifestations sont d'ailleurs liées aux autres manifestations viscérales, sur lesquelles elles ont des effets déclencheurs.
On a tenté d'établir des correspondances directes entre hormones et émotions, mais cela s'est révélé assez réducteur. Une hormone donnée ne semble pas spécialisée dans une émotion donnée.

3. L'identification des différentes émotions

Le mot "émotion" désigne un phénomène générique, qui s'instancie en diverses émotions particulières, comme la colère ou la peur. Comment distinguer et définir ces instances ? Ont-elles d'ailleurs une réalité psychologique ?

3.1. Méthodes expérimentales en psychologie

En psychologie expérimentale, lorsqu'on veut étudier une émotion particulière, on peut essayer de la provoquer chez un sujet pour pouvoir l'observer. Mais cela suppose qu'on a déjà une idée des situations qui peuvent déclencher cette émotion, et que ces situations vont nécessairement déclencher l'émotion voulue dans une forme "pure". C'est rarement le cas, et les expériences fondées sur ce principe ne fournissent pas de conclusions très assurées, d'autant plus qu'elles sont surtout effectuées sur des animaux.

On peut également étudier les réactions face à diverses situations, mais sans préjuger d'une émotion particulière. Des mesures comme celles qu'on a décrites précédemment peuvent ainsi être obtenues et comparées pour voir si des comportements caractéristiques apparaissent.
Un autre procédé, plus indirect, consiste à demander à des sujets d'identifier des émotions dans le comportement d'autres personnes (sur des photos ou des films par exemple). Plutôt que d'identifier les émotions, il s'agit alors d'identifier les concepts relatifs aux émotions, ou même les émotions relatives aux émotions. Un accord entre les sujets ne prouve pas vraiment qu'il existe un *phénomène* objectif appelé "peur", mais prouve qu'il existe un *concept* de peur partagé par les sujets.

3.2. Méthodes linguistiques

Les méthodes linguistiques étudient l'utilisation des mots désignant les émotions dans des textes en différentes langues. Le but est d'essayer de dégager des concepts universels relatifs aux émotions, du moins tels qu'ils se manifestent dans le langage. Nous verrons plus loin les travaux d'Anna Wierzbicka utilisant cette méthode.

3.3. Certaines émotions semblent "universelles"

Les résultats des différentes tentatives d'identification des émotions particulières confirment l'existence d'émotions de base identifiables, comme la joie, la surprise, la tristesse, la colère, le dégoût, la peur. Les chercheurs ne se sont pas mis d'accord sur une liste standard, mais les exemples précédents font l'objet d'un accord assez vaste.
D'autres émotions sont également citées, comme le mépris, la honte, la culpabilité, l'espoir, le désespoir, l'amour, le soulagement.

4. Emotions et autres phénomènes mentaux

Les émotions ne sont pas des phénomènes isolés du reste des phénomènes mentaux. Nous étudions ici les rapports qui peuvent exister entre les émotions et d'autres aspects, comme le jugement, la mémoire, l'apprentissage, le raisonnement.

4.1. Corrélations entre émotions et fonctions mentales

On a identifié des régions du cerveau impliquées simultanément dans les processus suivants :

— ressentir et exprimer des émotions ;

— adopter un comportement socialement approprié ;
— adopter un comportement avantageux pour soi ;
— prendre une décision (personnelle ou sociale) en situation d'incertitude ;
— prévoir un avenir incertain (personnel ou social) et planifier en conséquence ;
— retenir présente à l'esprit l'image d'un objet en absence de ce dernier.

Des résultats concordants ont été obtenus par différents procédés, comme l'étude de patients ayant subi des dommages dans une région du cerveau ou l'observation de l'activité électromagnétique du cerveau.

4.2. Emotion et jugement

Les corrélations précédentes ont donné lieu à des interprétations concernant le rôle des émotions dans des tâches de jugement. En particulier, A. Damasio [Damasio 95] interprète une émotion comme un qualificatif qu'on peut appliquer à une situation ou à une pensée. C'est un rapport d'évaluation condensé : bon / mauvais. En ressentant une émotion, on perçoit à la fois une situation ou une pensée (le qualifié) et un état du corps (le qualificatif), et on associe ces perceptions. Les émotions facilitent ainsi la formation d'opinions, de jugements, de décisions. Le cerveau tient compte du jugement émotionnel pour règler la "tonalité cognitive", c'est-à-dire par exemple la rapidité du raisonnement et l'abondance et la diversité des images mentales, des associations d'idées, des déductions. Par exemple, lorsqu'on est dans un état dépressif, le rythme du cerveau est lent et les images évoquées sont peu nombreuses et peu variées.

On distingue souvent les émotions négatives (désagréables, associées à un jugement défavorable) et les émotions positives. Les émotions négatives sont plus variées que les positives. Cela peut s'expliquer par leur rôle dans la sélection naturelle : il faut d'abord survivre et pour cela reconnaître les dangers potentiels, qui sont nombreux. Les deux types d'émotions semblent d'ailleurs pris en compte par des systèmes neurologiques différents.

4.3. Emotion, mémoire et apprentissage

Les émotions sont liées également à la mémoire et à l'apprentissage.

Il semble y avoir deux systèmes de mémoire : l'un pour l'information brute, l'autre pour la charge affective : "L'hippocampe vous permet de reconnaître votre cousine. Mais l'amygdale ajoute qu'elle est antipathique" (LeDoux). L'hippocampe et l'amygdale sont deux parties du cerveau impliquées dans ces processus, l'amygdale jouant un rôle important dans le traitement des émotions.

En cas d'émotion intense, l'ordre est donné de mémoriser la situation et de l'associer à l'émotion. On apprend ainsi des règles du type : situation → émotion. Une situation nouvelle peut correspondre à plusieurs règles et provoquer une combinaison d'émotions, qui peut être mémorisée comme une nouvelle émotion. Cela peut ainsi conduire à l'apprentissage d'émotions nuancées. On a d'ailleurs remarqué que les enfants ont au départ un répertoire limité d'émotions et qu'ils arrivent à en acquérir de nouvelles par l'expérience (dans laquelle les contacts sociaux ont une grande importance).

4.4. Raisonner sur les émotions

Il est possible d'effectuer des raisonnements à propos des émotions, et cela peut intervenir dans les prises de décisions. On donne ci-dessous des exemples exprimés sous formes de règles.

Prévoir pour éviter :

si X provoque la peur
et si Y implique X,
alors éviter Y.

Si les lions me font peur et qu'ils viennent boire le soir à la rivière, j'évite de me trouver le soir à la rivière.

Généraliser pour éviter :

Si X provoque la peur
et si X est un Z
alors éviter les Z

Si les lions me font peur, sachant qu'il s'agit de carnivores, j'évite les carnivores. Cela peut conduire à une attitude exagérément prudente et me donner la phobie des chats par exemple.

Observer pour affronter :

Si X provoque la peur
mais que X peut être annihilé par l'action A
alors, lors de la rencontre avec X :
diminuer la peur
et augmenter la focalisation sur l'action A

Achille me fait peur mais je sais qu'il est faible du talon, donc je prends la résolution de l'affronter en visant son talon.

5. Caractéristiques de l'esprit émotionnel

Daniel Goleman [Goleman 97] a résumé ce qu'étaient selon différents chercheurs les caractéristiques de l'"esprit émotionnel", c'est-à-dire des mécanismes mentaux concernant les émotions. Les traits principaux sont les suivants.

5.1. Rapidité et simplicité

La pensée est automatique, inconsciente, involontaire, accompagnée d'un sentiment de certitude fort. La perception est simplifiée et saisit les choses d'un seul coup, dans leur ensemble. Le jugement est intuitif, instantané et parfois trompeur. La pensée est catégorique, sans nuances, exagérée : tout, toujours, jamais. La précision est sacrifiée à la vitesse.

5.2. Retour au passé

L'esprit réagit au présent comme s'il était le passé, en essayant de reconnaître dans la situation présente des situations passées (consciemment ou non) et d'appliquer les réactions qui ont réussi alors.

5.3. Logique associative

Les idées et images s'enchaînent par évocations et associations plus que par une démarche logique. Le mode de pensée est celui des symboles, des métaphores, des rites, de la poésie, du mythe et du rêve. Les évocations sont souvent superficielles, fondées sur des détails ou des traits de surface.
Le déclenchement d'une émotion particulière mobilise un catalogue privilégié d'actions, de pensées et de souvenirs associé à cette émotion, ce qui permet de préparer l'individu à une réaction rapide et appropriée.

5.4. Conviction personnelle

La pensée est personnalisée, autocentrée et autojustificatrice : elle ne considère que son propre point de vue, trouve toujours de bonnes raisons pour se justifier et écarte les objections.

6. Les travaux linguistiques de A. Wierzbicka

Anna Wierzbicka a essayé de définir les émotions, ou plus précisément les concepts de sens commun relatifs aux émotions (ce que les gens disent à propos des émotions), en utilisant une approche linguistique [Wierzbicka 92].

6.1. Primitives sémantiques universelles

Pour arriver à définir la sémantique d'un terme d'un langage, il faut disposer d'un métalangage. Anna Wierzbicka a donc cherché à définir d'abord un ensemble de "primitives sémantiques" qui soient universelles, c'est-à-dire communes à toutes les langues et à toutes les cultures (un "alphabet des pensées humaines"). Cette recherche, extrêmement approfondie, dure depuis plus de vingt ans à travers l'étude de nombreuses langues de la planète. Les primitives qu'elle distingue actuellement sont les suivantes, classées par catégories :

"substantifs" :	je, tu, quelqu'un, quelque chose, les gens.
"déterminants et quantificateurs" :	ceci, le même, un autre, un, deux, beaucoup, tous.
"prédicats mentaux" :	penser (à), dire, savoir, connaître, sentir, vouloir.
"actions et événements" :	faire, arriver (*happen*).
"évaluatifs" :	bon, mauvais.
"descriptifs" :	grand, petit.
"temps et lieu" :	quand, où, avant, après, sur, sous.
"méta-prédicats" :	non, parce que, si (conditionnel), pouvoir (*can/may*).
"intensificateur" :	très.
"taxonomie, partonomie" :	sorte de, partie de.
"prototype" :	comme.

Cette liste peut sembler banale, mais c'est justement parce qu'elle est formée de concepts universels et primitifs. Elle a été testée extensivement pour vérifier que chaque langue possède un terme pour chaque concept, que ces termes sont effectivement capables de servir de blocs de base pour définir d'autres concepts, que chaque terme est facilement compris par tout le monde et que les concepts primitifs ne peuvent pas être définis les uns par les autres. Indépendamment de toute recherche sur les émotions, une telle liste peut être extrêmement utile en IA, notamment pour le traitement des langues naturelles, le dialogue homme-machine et l'apprentissage.

6.2. Définition des émotions par situations prototypiques

A partir des primitives universelles, Anna Wierzbicka a essayé de définir les termes anglais exprimant des émotions. Les concepts d'émotion, eux, ne sont pas universels et il faut préciser à quelle langue et à quelle culture ils se rapportent. C'est pourquoi, dans les exemples qui suivent, nous utiliserons les termes anglais : "happy" en anglais n'est pas l'équivalent exact de "heureux" en français.
Une définition se présente sous la forme d'un scénario prototypique : pour chaque émotion, on décrit une situation et des pensées qui accompagnent cette situation. Ce scénario ne prétend pas être une caractérisation ou une explication de l'émotion, mais il donne seulement un exemple en disant : imaginez que vous êtes dans cette situation et que vous pensez ce qui est décrit, alors ce que vous ressentiriez s'appelle … (de la peur ou de la surprise, etc.). La possibilité de définir un concept de cette façon repose d'abord sur le fait qu'il est exprimé à l'aide des primitives universelles supposées compréhensibles par tout le monde et ensuite sur nos capacités d'empathie et d'anticipation (nous sommes capables d'imaginer ce que nous ressentirions dans une situation donnée).

Anna Wierzbicka distingue différents thèmes intervenant dans une classification des émotions : les choses qui arrivent aux gens, les choses que les gens font, les pensées des gens sur les gens, les attitudes émotionnelles des gens envers les gens. Les primitives les plus souvent utilisées sont : bon, mauvais, arriver, faire, vouloir, non vouloir, je, quelqu'un, tous les gens. Voici des exemples de termes anglais dans différents thèmes :

— les mauvaises choses qui arrivent aux gens
sad, unhappy, distressed, upset, sorrow, sorry, grief, despair, depressed

— les bonnes choses qui arrivent aux gens
joy, contented, pleased, delighted, excited, happy

— les mauvaises choses faites par les gens
anger, indignation, shocked, appalled, hurt

— penser à soi-même
remorse, guilt, ashamed, humiliated, embarrassed, pride, triumph

6.3. Exemples de définitions

Donnons maintenant des exemples de définitions.

Quelquefois une personne pense quelque chose comme :

j'ai fait quelque chose

à cause de cela, quelque chose de mal s'est produit

à cause de cela, je ne peux pas m'empêcher de penser du mal de moi

A cause de cela, cette personne ressent quelque chose de mauvais.

Quelqu'un qui ressent quelque chose comme cela ressent de la culpabilité (*guilt*).

La définition a été rédigée pour former des phrases lisibles, mais il ne faut pas oublier qu'elle peut être en fait codée formellement uniquement à l'aide des primitives sémantiques. La définition a été écrite en français car elle n'utilise que les primitives universelles. Seul le terme défini (*guilt*) est associé à une langue et une culture.

L'intérêt d'une telle forme de définition apparaît lorsqu'on cherche à comparer des concepts entre eux. Modifions par exemple la définition précédente en remplaçant les pensées de la personne par :

J'ai fait quelque chose de mal

je savais que c'était mal

je ne peux pas m'empêcher de penser : c'était mal

j'aimerais ne pas avoir fait cela

Cette fois, la personne était consciente que ce qu'elle faisait était mal et contrairement au scénario précédent elle aimerait ne pas l'avoir fait. Le terme anglais ainsi défini est *remorse*.

On peut également mettre en évidence des définitions duales, comme "soulagement" et "déception" :

Quelquefois une personne pense quelque chose comme :

" quelque chose de mauvais va arriver

je ne veux pas cela"

A cause de cela, cette personne ressent quelque chose de mauvais.

Après cela, cette personne pense quelque chose comme :

" je sais maintenant que cela n'arrivera pas"

A cause de cela, cette personne ressent quelque chose de bon.

Quelqu'un qui ressent quelque chose comme cela ressent du *soulagement*.

Si on remplace dans cette définition “mauvais” par “bon”, “bon” par “mauvais” et “je ne veux pas” par “je veux”, on obtient une définition de la *déception*..

Références bibliographiques

[Damasio 95] Damasio A.R., L’erreur de Descartes, la raison des émotions, Odile Jacob, Paris, 1995.

[Dantzer 94] Dantzer, Les émotions, collection “Que sais-je”, PUF, Paris, 1994.

[Goleman 97] Goleman D., L’intelligence émotionnelle, comment transformer ses émotions en intelligence, Robert Laffont, Paris, 1997.

[Wierzbicka 92] Wierzbicka A., Defining Emotion Concepts, Cognitive Science 16, 539-581, 1992.

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Les Emotions Mécaniques

Régis Moneret

LIP6
Université Pierre et Marie CURIE
Regis.Moneret@poleia.lip6.fr

Résumé: Dans cet article, nous nous interrogeons sur l'intérêt que peut présenter pour un système informatique en relation avec des utilisateurs humains, le fait de simuler des émotions. Nous présenterons ensuite des systèmes ayant implémenté cette approche. Enfin, nous nous poserons la question de savoir si, dans certains domaines, où la machine obtient des performances très inférieures à celles de l'être humain, il ne serait pas nécessaire de lui intégrer également des mécanismes comparables aux émotions.

Mots-clefs : Emotions, Simulation, Intelligence Emotionnelle.

1 Introduction

Comme l'a montré [Pitrat 1997], un système informatique interagissant avec des êtres humains peut avoir besoin de connaître leur état émotif. Cela lui permet en effet de s'adapter spécifiquement à ses interlocuteurs. Toutefois, il peut arriver que les émotions nuisent précisément à l'interaction harmonieuse entre l'homme et la machine. C'est par exemple le cas lorsqu'un élève se servant d'un logiciel d'EIAO (Enseignement Intelligent Assisté par Ordinateur) est découragé par des échecs répétés. Pour atteindre son objectif (enseigner une matière à l'élève), le système doit par conséquence commencer par modifier son état émotionnel, après l'avoir identifié. Or, empiriquement, on constate que ce phénomène " d'influence " passe très souvent par l'expression d'émotions (telles qu'une attitude de soutien dans le cas précédent). Les programmeurs sont donc naturellement amenés, à tenter de faire simuler des émotions par leurs systèmes, afin de les rendre plus " proches " de leurs utilisateurs humains, plus " conviviaux ".

Cependant, toutes les émotions ne sont pas néfastes, loin de là. Elles permettent aux êtres humains de briller dans des domaines où la machine malgré sa puissance de calcul obtient des résultats comparables -au mieux- à ceux d'un enfant (compréhension des langues naturelles par exemple). C'est pourquoi, on peut à juste titre se demander si les émotions humaines ne pourraient pas servir de modèles à la machine. Allons encore plus loin : le seul exemple d'être intelligent que nous connaissions à l'heure actuelle possède justement des d'émotions. Peut-il donc exister une Intelligence totalement dépourvue d'émotions ? Ne devons-nous pas tenter d'abstraire ce qui fait la force des émotions humaines afin de les coder dans la machine ? En d'autres termes, la machine ne doit-elle pas réellement ressentir des " émotions " à l'instar des êtres humains si nous voulons lui faire accomplir certaines tâches (génération ou traduction automatique d'œuvres littéraires...) ? Voici quelques unes des questions auxquelles nous essayerons de répondre tout au long de cet article.

Dans le premier paragraphe, nous reviendrons sur les avantages qu'une machine peut avoir à simuler des émotions humaines, puis nous détaillerons quelques systèmes existants qui ont implémenté cette approche. Enfin, nous passerons en revue les avantages que les émotions peuvent présenter pour un être intelligent, avant de conclure sur le rôle des émotions pour l'Intelligence Artificielle.

2. Applications possibles de systèmes simulant des émotions

2.1 En EIAO

Comme nous l'avions déjà mentionné dans l'introduction, un système d'aide à la décision " intelligent " ne devrait pas seulement être capable de reconnaître le découragement, la fatigue, l'ennui ou le manque d'attention de ses utilisateurs, mais devrait en outre être capable d'y porter remède en se montrant chaleureux par exemple, ou au contraire, en se mettant en colère. Le but étant ici d'apporter un soutien moral, de susciter une plus grande motivation chez l'utilisateur, en plus du seul enseignement des connaissances.

2.2 Pour des Interfaces Intelligentes

Les émotions permettent de communiquer rapidement les états internes entre agents : ainsi, par exemple, si je sais que l'ingénieur système n'est pas de bonne humeur, je remettrai peut-être à

plus tard les questions que je souhaitais lui poser à propos de la configuration de tel ou tel logiciel. Et cela, sans avoir explicitement communiqué avec lui. Si l'ordinateur était capable de transmettre de manière analogue ses " émotions " (par exemple sa charge de travail au moyen d'un code couleur), les utilisateurs gagneraient un temps précieux. Ceci est tout particulièrement vrai dans le cas de certaines situations (pilotage d'avion...) où il faut communiquer un maximum d'informations en un minimum de temps. De même, des ersatz d'émotions pourraient être utilisés dans des systèmes multi-agents pour accélérer les communications entre machines (robots, processeurs...)

2.3 En Traitement des Langues Naturelles

Dans ce domaine tout particulièrement, la simulation des états affectifs revêt un intérêt crucial. En effet, dès que l'on sort des textes techniques, il est nécessaire d'introduire des biais émotionnels pour adapter le discours en fonction de son interlocuteur. Ainsi, par exemple, on n'emploiera pas le même vocabulaire selon qu'on cherche à convaincre quelqu'un ou au contraire, à se montrer objectif, selon qu'on cherche à se faire apprécier, ou que l'on veut mettre quelqu'un en colère.

2.4 Pour des Systèmes d'aide à la décision

Même ici, l'ordinateur devra à plus ou moins long terme utiliser les émotions pour vaincre les doutes et les réticences de ses utilisateurs. Rares sont en effet les personnes qui acceptent une décision engageant leur avenir sans connaître les raisons qui la fondent. La machine devra expliquer les motifs de son choix, et convaincre ses utilisateurs, sans doute comme nous-mêmes êtres humains le faisons : à l'aide de biais affectifs, de techniques de vente, qui permettent " d'inspirer confiance ". Peut-être même devra-t-elle faire preuve de charisme pour emporter l'adhésion.

2.5 En Science Cognitive / Vie Artificielle

La simulation étant un moyen de vérification des modèles sur les émotions, son intérêt dans ce domaine n'en est que plus évident encore.

3. La Simulation Emotionnelle

Il est toujours difficile de parler des “ émotions ” d’une machine sans tomber dans un anthropomorphisme primaire consistant à lui attribuer des sentiments que NOUS ressentirions si nous étions “ à sa place ”. Ainsi, par exemple, Kasparov lors du match qui l’opposa à Deep Blue [Guterl 1996] oublia momentanément l’aspect mécanique de son adversaire pour évoquer sa “ confusion ” lors du dernier jeu, avis d’ailleurs partagé par la plupart des observateurs. Ceci prouve si besoin était, que même le programme le plus inhumain peut être crédité de sentiments pourvu que son comportement s’insère un tant soit peu dans nos manières de penser. C’est précisément sur cet anthropomorphisme que vont se baser la plupart des systèmes de simulation, en comptant sur l’empathie des utilisateurs pour “ combler leurs lacunes ”.

3.1 Problématique de la simulation émotionnelle

[Pitrat 1997] a déjà évoqué le problème lié à la représentation des émotions en machine. Bien que parlant des systèmes de détection d’émotions, ses propos s’appliquent tout naturellement aux systèmes de simulation. Néanmoins, la simulation possède également des particularités spécifiques, car il faut non seulement savoir traduire une émotion par des comportements “ physiques ” (voix, gestes, choix du vocabulaire...), ce qui représente en quelque sorte l’étape inverse de la reconnaissance des émotions, mais en outre, il faut être également en mesure d’intégrer harmonieusement ces différents modes d’expressions (problème dont un analyseur n’a pas à se soucier).

3.2 L’Expression “ Physique ” des Emotions

Les émotions sont intimement liées avec nos attitudes corporelles : ainsi, lorsque nous sommes en colère, notre corps (et en particulier notre visage), notre voix, nos actions même reflètent notre état mental. Il ne s’agit pas simplement d’un but ou d’une action qui traduirait ponctuellement notre colère, mais d’une série de gestes et de paroles qui vont petit à petit composer une peinture “ impressionniste ” de nos sentiments. [Hovy 1990] l’a particulièrement bien montré en cherchant à analyser comment un locuteur parvenait à donner un certain niveau de politesse à son discours : en effet, aucun mot à lui seul ne peut rendre poli un discours “ neutre ”. C’est au contraire tout un ensemble de choix syntaxiques (complexité des phrases...) et sémantiques (choix des mots...) qui rend ou non un discours formel. De même, une saccade oculaire, un clignement de paupières n’a pas de signification en lui-même, mais en

relation avec l'ensemble du discours (et ceci, de manière différente en fonction des interlocuteurs).

3.3 L'Intégration des différents Traits Emotionnels

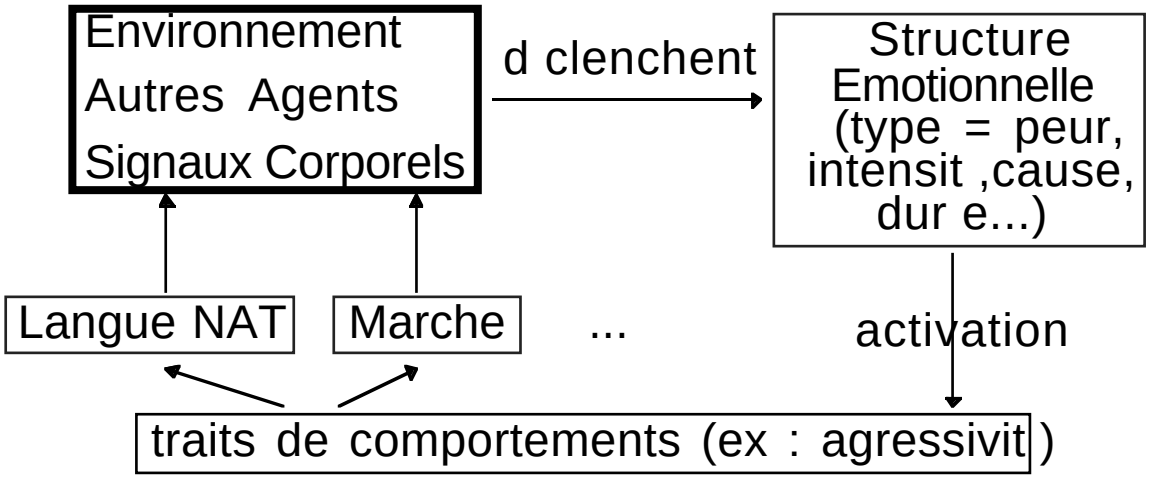
Dans cette partie, nous présenterons trois architectures qui ont été développées afin de permettre l'intégration de différents traits émotionnels au discours :

Une Architecture par planification " mixte " [Hovy 1990] :
L'idée est de combiner les avantages de la planification traditionnelle " top-down " avec des modules de choix stratégiques fonctionnant de manière " bottom-up " destinés à tirer parti des opportunités qui se présentent au niveau sémantique. Ainsi, on construit un plan " classique " pour satisfaire des buts ponctuels (arrangement des idées, des paragraphes...) alors que les buts pragmatiques, souvent contradictoires entre eux (être détaillé tout en restant bref...) sont gérés localement par les modules de choix de bas niveau (tournures, vocabulaire...) qui vont permettre de réduire ainsi la combinatoire. Des variables globales servent à comptabiliser la fréquence avec laquelle les buts pragmatiques ont été satisfaits tout au long du texte (a-t-on été assez formel ? assez poli ? ou faut-il rajouter quelques tournures littéraires pour parfaire le tout ?).

Une Architecture par règles empiriques a différents niveaux d'abstraction [Pelachaud 1996] :
Ce système tente d'utiliser des mouvements corporels (mouvement des yeux, des lèvres, inclinaison de la tête, position du corps,...) pour transmettre le mieux possible un texte écrit à l'oral. Pour ce faire, l'auteur fournit au programme un ensemble de règles qui décrivent les émotions en termes d'activations de groupements musculaires (dans un formalisme analogue avec ce qui se fait actuellement dans le monde des effets spéciaux au cinéma). Le problème étant que ces règles ne sont pas activées au même moment : en effet, alors que les mouvements des lèvres doivent être " réactualisés " après chaque phonème prononcé, les saccades oculaires suivent un rythme plus lent (en gros, le rythme des mots). Quant aux postures et aux mouvements de la tête, ils ne varient qu'au bout d'une phrase en moyenne. Pour rendre la simulation encore plus réaliste, certaines règles décrivent des mouvements corporels liés à des besoins biologiques (cligner des paupières pour humidifier les yeux). Le système procède alors en commençant par activer les règles qui suivent le rythme le plus rapide. A partir des faits inférés, il remonte ensuite vers les règles qui ont un déclenchement plus long. Finalement, un plan d'activation des différents groupes musculaire est ainsi produit pour " déclamer " le texte initial.

Une Architecture " broad but swallow " [Oz 1997] :

Un groupe de recherche à l'université Carnégie-Mellon travaille actuellement sur la mise au point d'un environnement virtuel dans lequel des êtres humains pourraient interagir de manière " crédible " avec des agents dotés d'émotions. Ce projet, à la fois artistique (puisqu'il s'agit de créer un nouveau médium, à mi-chemin entre le roman, le film et le jeu sur ordinateur) et scientifique (puisque l'équipe se propose, à travers la création de tels agents, de mettre à jour les mécanismes liés à notre perception des émotions) est symptomatique des efforts menés à l'heure actuelle pour réaliser des agents communicants (" ChatterBots " de 'CHATTER' bavarder et de 'roBOTS' robots) de plus en plus robustes à l'interaction humaine. Il faut bien comprendre que le but n'est plus ici de faire des agents qui soient **intelligents**, ni même **réalistes** mais bien des agents **crédibles**, **uniques entre eux, et dotés d'une forte personnalité**. Ce projet s'éloigne donc quelque peu de l'Intelligence Artificielle traditionnelle au sens où les agents en IA sont censés être intelligents, se ressembler tous plus ou moins, sans véritable personnalité propre. Il s'éloigne également des travaux en Vie Artificielle où le but est de créer des agents le plus conforme possible à la réalité. Les chercheurs de Carnégie-Mellon s'inspirent en cela de sondages réalisés auprès d'artistes qui ont en effet montré que les personnages de films ou de romans n'avaient pas nécessairement besoin d'être très réalistes pour être crédibles (parfois même, un trop grand réalisme pouvait susciter de mouvement de méfiance de la part des spectateurs qui pouvaient se sentir mal à l'aise à cause de cette ressemblance proche-mais-pas-parfaite avec la réalité). Au contraire, les personnes qui lisent des romans ou vont au cinéma voir des films ont naturellement tendance à mettre en veille leur sens critique. Ainsi, une caricature de la réalité suffit-elle à créer l'illusion de la vie. Ce qui ressort également de cette enquête, c'est que les personnages de fiction, pour être crédibles, doivent exprimer leurs émotions de manière aussi visible que possible, afin qu'elles soient perçues par tous les publics. Cet argument a poussé les membres de l'équipe à développer une architecture " broad but swallow ", dans laquelle l'expression des émotions passe par l'ensemble des actions de l'agent, que ce soit ses paroles, ses actes ou encore sa physionomie. La Figure 1 illustre cette architecture : au départ, l'environnement, les autres agents, ou le corps de l'agent lui-même déclenche la création d'une structure émotionnelle caractérisée par son type d'émotion, son intensité, sa cause, la rapidité avec laquelle cette émotion va disparaître... Une fonction définie spécifiquement pour chaque agent, active alors un trait de comportement (ceci permet de moduler la réponse en fonction de l'agent : en effet, bien que nous ressentions tous les mêmes émotions, nous avons chacun une manière propre de les exprimer, certains étant plus expansifs que d'autres). Le nombre de traits de comportement (agression, affection,...) est fixe, et sert à paramètrer les différentes fonctions du système. Ainsi, selon les traits déclenchés, et donc, selon les émotions de l'agent, le système de reconnaissance et de génération de langue naturelle (à base de 'templates'), le système de déplacement... réagiront d'une manière différente. A noter que cette approche est très modulaire, et qu'on peut aisément rajouter/enlever/remplacer des modules sans pour autant être obligé de toucher au reste du système.



4. Les Emotions Artificielles

Dans cette partie nous nous intéresserons au problème de savoir s'il serait utile de doter une machine de mécanismes comparables à ce que sont les émotions chez l'être humain. Comme il n'existe pas à ma connaissance d'exemple d'un tel système, nous ne pourrons malheureusement que présenter au lecteur des pistes qui nous semblent intéressantes d'explorer, en lui laissant le soin de juger de la pertinence d'une telle approche.

4.1 Les émotions comme " accident " (au sens philosophique) de la complexité

Pour certains psychologues tels [Sloman], les émotions seraient des propriétés émergentes de tout système suffisamment complexe disposant de ressources limitées, un peu à la manière des " deadlocks " en Architecture Réseaux où il existe un nombre précis de stations pouvant être connectées à une unité centrale, au-delà duquel le temps de réponse de chaque station augmente brutalement, et ceci, sans qu'on puisse trouver de paramètre physique dans le système qui soit directement responsable de ce nombre. En d'autres termes, les mécanismes du cerveau chargés de gérer les difficultés d'un environnement complexe pourraient naturellement se retrouver dans des états " perturbés " (que Sloman appelle " émotionnels ") sans que rien dans l'architecture du cerveau ne soit directement lié à l'apparition de tels états.
Si tel est le cas, des systèmes informatiques complexes pourraient également manifester de tels " troubles " sans que leurs auteurs les aient explicitement codés.

4.2 Existence d'une Intelligence sans Emotions ?

[Damasio 1994] a montré que des personnes qui ont subi des lésions cérébrales ayant endommagé leurs centres nerveux émotionnels perdent également la capacité de prendre des décisions, et, d'une manière générale, à gérer leur vie**, et cela, alors que leurs capacités intellectuelles** mesurées au moyen de tests psychométriques **soient demeurées intactes**. Il semble donc bien, au moins pour l'être humain, que les émotions jouent un rôle non négligeable dans la manière que nous avons d'appréhender le Réel.
Partant de ce constat, bon nombre de chercheurs en IA, en Science Cognitive ou en Psychologie ([Damasio 1994], [PNL 1988], ...) ont essayé de recenser les différentes fonctions des émotions pour l'homme. Parmi les plus citées :

- Sortir de bouclages (ennui)
- Ajout/Suppression de buts à notre "agenda" (amour, haine, ...)
- Modification des priorités/intérêts de nos buts
- Mobilisation de ressources (peur, angoisse, ...)
- Jugements (goût, esthétisme...)[1]
- Moteur de la créativité (frustration...)

Certaines de ces fonctions sont déjà prises en compte par de nombreux systèmes informatiques (modification des priorités de différents plans, interruptions...), d'autres ont fait l'objet de recherche poussées (sortir de bouclages [Kornman 1993]). Par contre, les problèmes liés aux choix subjectifs ainsi qu'à la créativité, malgré quelques travaux sur le sujet ([Lenat 1983], [Hofstadter 1995],..) demeurent encore hors d'atteinte de nos systèmes.
Prenons par exemple les programmes cherchant à générer des concepts nouveaux. Il est généralement assez facile d'écrire un algorithme qui combine des concepts déjà existant. Par contre, ce qui pose évidemment problème, c'est de trouver un moyen d'évaluer les concepts qui ont été ainsi produits (ce n'est d'ailleurs pas un hasard si Lenat procédait lui-même à l'évaluation des résultats produits par EURISKO, voir à ce sujet [Boden 1997]) : en effet, s'il est déjà très difficile d'expliquer d'une manière accessible à un ordinateur les raisons pour lesquelles nous apprécions une fugue de Bach ou encore une toile de Monet, ceci devient une tâche presque impossible lorsqu'il s'agit de définir a priori des critères d'évaluation pour des

---

[1] Selon [Minsky 1985], nos jugements en matière de style et d'esthétisme sont des moyens utilisés par notre cerveau pour réduire la complexité du monde en nous épargnant un travail mental pénible pour distinguer entre deux alternatives proches (et donc, difficiles à distinguer) alors que de toute façon, n'importe laquelle convient.

œuvres non encore créées (comment définir ce qu'est un roman **intéressant** ?, un théorème **utile** ? une musique **entraînante** ?...).

5. Conclusion

La Simulation des Emotions est à mon avis appelée à se développer de plus en plus dans les années à venir (réalité virtuelles, IHM...). De plus, la reconnaissance et la simulation d'émotions étant deux faces d'une même pièce, les progrès des chercheurs en Sciences Cognitives sur les mécanismes humains devraient bientôt trouver leur pendant dans le domaine de la simulation, voire de la mise au point " d'émotions " véritablement propres à la machine.

6. Références Bibliographiques

[Boden 1997] Boden M.A., *Creativity and Artificial Intelligence*, IJCAI, pp1563-1566, 1997

[Comp-Go 1996] archives de la liste " computer-go@anu.edu.au " sur la programmation du jeu de Go disponible via FTP à ftp://bsdserver.ucsf.edu/Go/comp/compgo-mail-8.Z

[Damasio 1994] Damasio A., *L'Erreur de Descartes : la Raison des Emotions*, Odile Jacob, 1994.

[Dehn 1981] Dehn N., *Story-Generation after TALE-SPIN*, Proceedings of the seventh IJCAI, pp16-18,1981.

[Emotions 1997] Page Web sur les Emotions
http://emotion.ccs.brandeis.edu/Emotion/EmoRes/CompAI/CompAI.htm

[Goleman 1995] Goleman D., *Emotional Intelligence*, Bantam Books, 1995

[Guterl 1996] Guterl F., *Silicon Gambit*, archives du *Discover Magazine* disponible au
http://www.dc.enews.com/magazines/discover/magtxt/060196-toc.html

[Hofstadter 1995] Hofstadter D.R., *Fluid Concepts and Creative Analogies : Computer Models of the Fundamental Mechanisms of Thought*, Basic Books, New York, 1995.

[Hovy 1990] Hovy E.H., *Pragmatic and Natural Language Generation*, Artificial Intelligence n°43, pp 153-197, 1990.

[Kornman 1993] Kornman S., *SADE : Un Système Réflexif de Surveillance à base de Connaissances*, Thèse de Doctorat de PARIS VI, 1993.

[Lenat 1983] Lenat D.B., *The Role of Heuristics in Learning by Discovery : Three Case Studies* in Machine Learning, pp243-306, 1983

[Minsky 1985] Minsky M., *The Society of Mind*, Touchstone Book, 1985.

[Oz 1997] Site Web du Projet Oz dirigé par W.Scott Neal REILLY
http://www.cs.cmu.edu/afs/cs/project/oz/web/oz.html

[Pelachaud 1996] Pelachaud C. et al, *Generating Facial Expressions for Speech*, Cognitive Science 20 vol1, p 1-46, 1996.
[Pitrat 1997], Pitrat J., *Reconnaître les Emotions des Etres Humains*, Actes de Berder, 1997.
[PNL, 1988] Cudicio C., *Maîtriser l'Art de la Programmation Neuro-Linguistique*, Les Editions d'Organisation, 1988.
[Sloman] voir [Emotions 1997]

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Les Émotions : quels intérêts pour l'homme ?

Arnaud Fredon

LIP6
Université Pierre et Marie CURIE
fredon@poleia.lip6.fr

Résumé : Si les aspects négatifs des émotions sont connus de tous, il semble que leurs avantages pourtant nombreux aient mis plus longtemps à être répertoriés. Nous présentons donc une vue partiale présentant les aspects positifs des émotions. Ces avantages concernent la survie de l'individu et de l'espèce, la capacité à communiquer avec ses semblables, et enfin la capacité à raisonner correctement.

Mots-clés : émotions, sciences cognitives, raisonnement humain.

1 - Introduction

L'une des principales difficultés rencontrées dans l'étude des émotions provient du fait que les travaux menés sur ce sujet dans le cadre de différentes disciplines (éthologie, biologie, psychologie cognitive, psychologie sociale, sociologie, …) restent empreints d'une approche plus philosophique que scientifique. Le terme "émotion" lui-même voit sa définition varier considérablement d'un auteur à l'autre, certains refusant même de le définir.
Il faut dire que l'histoire des disciplines cognitives explique le manque considérable d'études effectuées sur le sujet. Descartes, en premier lieu, sépare la raison de l'émotion (la *res cognitans* opposée à la *res existencia*), plaçant cette dernière dans le cadre encore actuel d'un processus tout au plus gênant. Plus tard, les behavioristes, s'interdisant toute interprétation des comportements observés, laissent également de côté l'aspect émotionnel. Les cognitivistes enfin délaissent ce secteur d'étude qu'ils jugent pour la plupart comme étant en dehors de leur champ d'étude.
L'enchaînement de ces courants de pensée majeurs explique donc le manque d'approche rigoureuse sur la matière. Aussi, notre article présentera pour l'essentiel des hypothèses et non des certitudes.

2 - L'émotion et la survie : avantages adaptatifs pour l'individu et l'espèce

En 1814 déjà, le chevalier de Lamarck [LAM 14], dans sa classification des espèces, utilisait dans ses critères de classification la capacité à ressentir des émotions. Cette capacité distinguerait selon lui les trois quarts supérieurs des espèces. Le critère suivant de sa hiérarchie était l'intelligence. Ces idées trouvent leur aboutissement une cinquantaine d'années plus tard chez Charles Darwin. Celui-ci, dans "l'expression des émotions chez l'homme et les animaux" [DAR 77], juge en effet que la capacité à ressentir des émotions est une forme supérieure de l'adaptation. On peut également extrapoler les travaux de Lamarck pour supposer que celui-ci faisait de l'émotion un préambule implicite à l'intelligence.

L'avantage le plus évident des émotions vient de leurs manifestations physiologiques. Ces mécanismes ont permis, chez l'homme notamment, d'augmenter les chances de survie en permettant des réactions rapides et appropriées. La peur, par exemple, s'accompagne d'une décharge importante de noradrénaline, ainsi que d'une mobilisation rapide du sang vers les muscles de la locomotion permettant ainsi la fuite (cette mobilisation du sang vers les muscles du grand squelette explique d'ailleurs la pâleur du visage). De plus, le mécanisme qui maintient l'individu figé pendant une fraction de seconde permettrait d'envisager la possibilité de se cacher au lieu de fuir [GOL 95]. De façon similaire, la colère provoque une mobilisation du sang vers les membres supérieurs permettant ainsi une utilisation plus précise et plus puissante des bras et des mains. L'intérêt de ces mécanismes physiologiques liés aux émotions primaires que sont la peur, la colère et la joie est évident et a été largement étudié.

Cependant, un autre aspect intéressant de ces manifestations émotionnelles a été exprimé en 1928 par Janet [JAN 28], puis repris en 1995 par David Goleman [GOL 95] : les émotions constituent un véritable système d'interruption des tâches. Lorsqu'une émotion importante se déclenche, la "mémoire de travail" de l'individu est vidée et les comportements complexes sont remplacés par des comportements plus simples et plus adaptés, telle que la fuite. Goleman va plus loin en parlant de "détournement émotionnel" : l'amygdale, une des parties du cerveau limbique qui participe au mécanisme des émotions, dispose d'une circuiterie spéciale qui lui permet de récupérer une partie des informations fournies par le système sensoriel. Ce système est approximatif, car l'information récupérée est partielle, mais il permet une communication beaucoup plus rapide que celle existant entre les organes sensoriels et le néocortex. L'amygdale peut alors estimer la nécessité d'une réaction immédiate, grâce à un système de "pattern matching" approximatif. Cette mauvaise qualité d'estimation n'est pas gênante car elle entraîne un surplus de prudence au lieu de l'inverse, ce qui est favorable à la survie de l'individu tant que cela ne devient pas trop excessif. Si l'amygdale conclut à une possibilité de danger, elle peut alors envoyer immédiatement des signaux appropriés aux différents organes concernés, ce qui permet une réaction avant même que le néocortex n'ait reçu le signal initial. Le néocortex aura ensuite l'occasion de tempérer ou de supprimer la réaction produite. On retrouve dans ce

mécanisme la mémoire inductive des formes que pressentait Konrad Lorentz lorsqu'il évoquait la mémoire émotionnelle [LOR 70].
Immelmann [IMM 82], de son côté, met en évidence le fait que ces réactions très rapides sont exploitées par des phobies instinctives. Ainsi, l'être humain naît avec une phobie naturelle des serpents et des araignées. Ce mécanisme est indispensable pour contrecarrer le réflexe de curiosité des nouveau-nés, qui auraient sans cela une fâcheuse tendance à multiplier les explorations hasardeuses. Par contre, grâce à l'apprentissage, l'adulte peut ensuite contrôler peu à peu ces phobies instinctives. Ainsi, l'être humain dispose d'un mécanisme de sécurité lui permettant de survivre aux choses inconnues jusqu'à ce que celles-ci soient connues.
Ainsi, les émotions ont un avantage évident en terme de survie. Les réactions physiologiques adaptées qu'elles provoquent sont déclenchées sans faire appel à des mécanismes de raisonnement qui seraient trop lents ou qui pourraient conclure à des buts incompatibles à la survie de l'individu faute d'informations suffisantes.

3 - L'émotion et la communication

Il serait cependant trop simpliste de ne voir dans les émotions qu'un mécanisme adaptatif seulement destiné à la survie de l'individu. Darwin lui-même pressent le rôle des émotions dans les situations communicationnelles intervenant pour les membres d'une espèce ayant une organisation sociale. Il formule l'hypothèse de la sociabilité qui marquera le début de nombreuses études sur la relation existant entre comportement social et émotions. Selon cette hypothèse, l'émotion permet à l'individu, qu'il soit humain ou animal, d'exprimer à ses congénères son désir présent de sociabilité. Les extrêmes de cette échelle continue se manifestent d'un côté par la joie, dans le cas où l'individu souhaite la présence d'autres membres de son espèce, et de l'autre côté par la colère, l'individu manifestant alors à ses congénères son désir d'être laissé seul. Cependant, cette hypothèse est insatisfaisante tout d'abord car elle ne fournit une explication que pour certaines émotions simples, et d'autre part car elle n'est pas exhaustive, la colère ou la joie pouvant se manifester en dehors d'un besoin d'expression de la sociabilité. Les cas évoqués par Darwin sont expliqués d'une façon pertinente par l'exploitation des émotions, évoquée par plusieurs auteurs, dont par exemple Marvin Minsky [MIN 85] : pour accomplir une tâche particulière, nous sommes capables de trouver de la motivation en faisant appel à des émotions comme l'anticipation d'une déception en cas d'échec, ou la joie d'un succès. Minsky présente par exemple le cas d'un chercheur capable de repousser le sommeil en stimulant sa colère à l'idée que son concurrent puisse aboutir avant dans ses recherches. Dans les exemples cités par Darwin, la sociabilité exploiterait la joie et la colère, au lieu de les provoquer.

Il faut attendre la publication des travaux de Wallon [WAL 34] pour avoir enfin une approche expérimentale sur le sujet. Wallon travaille avec des mères de jeunes enfants sur les colères exprimées par ceux-ci. Il distingue deux types de colères : les colères explosives qui semblent se manifester sans but et sans raison, et les colères orientées qui prennent pour cible un objet ou une personne (le plus souvent la mère). Son étude montre l'évolution de la répartition de ces colères en fonction de l'âge de l'enfant : tandis que les nouveau-nés manifestent presque uniquement des colères explosives, celles-ci laissent avec l'âge peu à peu la place à des colères orientées. C'est ce que Wallon baptisa la dramatisation des conduites. Cette idée est complétée par la suite par Château [CHA 83] : l'enfant est d'abord soumis à des colères explosives dues à des états internes insatisfaisants. Peu à peu, celui-ci apprend à orienter cette colère soit vers la source matérielle de son insatisfaction, soit vers une personne pouvant remédier à son état. Ce faisant, il fait peu à peu de la colère un moyen de communication. De plus, ne disposant pas d'autre moyen de communication, le jeune enfant doit pour sa survie manifester des signes clairs et excessifs, et trouve alors dans la colère une solution adéquate [MIN 85].

4 - L'émotion et le raisonnement

"La question n'est pas de savoir si des machines intelligentes peuvent avoir des émotions, mais si des machines peuvent être intelligentes sans avoir des émotions"

Marvin Minsky, *The Society of Mind*

Comme nous l'avons évoqué dans notre introduction, une dichotomie entre raisonnement et émotion perdure depuis longtemps et reste encore vivace pour la majorité des gens, comme en témoignent des expressions comme "garder son sang-froid" ou "réfléchir à tête reposée". Beaucoup considèrent encore nos processus émotionnels comme une perturbation du comportement, une nuisance à contrôler. Pendant des années, la querelle entre partisans de la théorie de James-Lange et ceux de la théorie de Cannon porte sur l'origine des émotions mais pas sur leur utilité. Comme nous l'avons déjà évoqué, certains auteurs s'intéressent tout de même à la relation entre intelligence et émotions. Château estime que l'intelligence se développe sur un fond de conscience affective, c'est-à-dire que l'émotion procède de l'intelligence, de même que le système néocortical, auquel on attribue préférentiellement les activités supérieures de l'intelligence, prend racine dans le système limbique, qui prend en charge les mécanismes émotionnels. C'est la prise de conscience de ses émotions d'origine purement physiologique qui permettrait l'apparition de l'intelligence.

Janet, quant à lui, estime que l'émotion est douée d'un pouvoir d'analyse. Ainsi, lorsque nous voyons un ours, nous montrons de la peur lorsque celui-ci est en liberté dans un forêt, et de l'amusement (ou de l'indifférence) lorsque le même animal se trouve dans un zoo.

Ces émotions sont présentées conjointement à la perception de l'animal et seul le contexte permet de distinguer les deux situations. Le système émotionnel serait responsable de cette distinction. On retrouve là le rôle analytique supposé exister dans l'amygdale, ou encore la mémoire inductive des formes de Lorenz.

Mais l'idée qui nous paraît la plus importante trouve son origine dans les travaux de Marvin Minsky. Celui-ci montre le rôle directif de l'émotion en prenant pour hypothèse l'existence d'un système rationnel, doué de logique, mais dépourvu d'émotions. Un tel système produirait alors des assertions de la forme :

A implique A ou A ou A
P ou non P
si 4 est 5 alors les cochons volent

Ces productions sont bien sûr tout à fait valables sur un plan logique, et pourtant nous les trouvons spontanément inintéressantes, voire absurdes. C'est le mécanisme des émotions qui permet un tel jugement. Dépourvue d'émotion, la logique erre sans fin dans les méandres de l'aberration. Ce rôle directif de l'émotion est donc très important pour mener une attitude rationnelle constructive. La dichotomie raison et émotion n'a alors plus lieu d'être, puisque les deux doivent travailler de concert pour produire un résultat satisfaisant. Minsky montre également le rôle de l'émotion pour la préservation des buts. En effet, aucun but ne peut être viable à long terme s'il ne dispose pas d'un mécanisme de préservation qui l'empêche d'être sans cesse remplacé par de nouveaux buts. L'émotion, ou plutôt la motivation, qui lui est associée en psychologie, permet cette préservation à long terme.

Mais les travaux de Minsky sont montrés d'une façon encore plus décisive par les travaux en neurologie d'Antonio Damasio. Au cours de sa carrière, Damasio a rencontré un individu qu'il a appelé le cas Elliot [DAM 95]. Elliot est un cas comparable à Phineas Gage qui fut célèbre au siècle dernier pour avoir vécu avec une partie du cerveau arrachée dans un accident. Elliot, lui, a perdu à cause d'un méningiome une partie de ses tissus cérébraux appartenant au cortex ventro-médian du lobe frontal. La lésion est faible, et Elliot semble disposer de toutes ses facultés intellectuelles, mémorielles, etc., comme l'a montré la série de tests qui lui ont été présentés. Pourtant, suite à cette maladie, sa vie se dérègle : il semble incapable de prendre une bonne décision, alors qu'il en est toujours capable dans des conditions de laboratoire. Damasio finira par se rendre compte qu'il ne manque à Elliot qu'une seule de ses facultés : la capacité de ressentir des émotions. Cette perte des émotions entraîne une indifférence du sujet face aux différents choix qui se présentent à lui dans la vie, ce que Damasio appelle la myopie de l'avenir.

Les travaux de Damasio l'amène parallèlement à formuler l'hypothèse des marqueurs somatiques, selon laquelle les processus émotionnels interviendraient pour effectuer un filtrage

de nos pensées afin que les processus de prise de décision soient facilités. Cette hypothèse novatrice reste encore très critiquée, mais elle présente l'avantage d'une explication simple d'une partie de notre fonctionnement mental.

5 - Conclusion

Ainsi, nous avons pu voir que l'émotion présentait un avantage adaptatif important, du moins chez l'homme car il est impossible d'estimer l'activité émotionnelle d'un animal de façon objective, car l'on prête une émotion à un animal manifestant un comportant attribuable à une émotion. Cette définition cyclique a conduit les éthologues à ne pas étudier cette question.
De plus, les émotions sont un instrument de la communication. Elles permettent une forme de communication par l'exploitation de ses propres émotions. Les expressions faciales liées aux émotions sont aisément reconnues par l'interlocuteur et assurent une information parallèle à celles fournies par le discours. L'émotion est donc le moyen de communication unique des jeunes enfants, mais reste un des moyens de communication chez l'adulte.
Enfin, il semble qu'au regard des travaux effectués au cours de ce siècle, il faille abandonner la vision classique des émotions. Celles-ci semblent en effet indispensables à nos processus intellectuels, essentiellement dans notre vie courante mais également pour des activités de plus haut niveau comme les mathématiques. Le raisonnement est intimement lié à la capacité à ressentir les émotions et la dichotomie de Descartes ne semble plus devoir perdurer.

6 - Bibliographie

[CHA 83] Château, Jean, "L'intelligence ou les intelligences ?", P.Mardaga, Bruxelles, 1983

[DAM 95] Damasio, Antonio R., "L'erreur de Descartes : la raison des émotions",Odile Jacob, Paris, 1995, trad.

[DAR 72] Darwin, Charles, "L'expression des émotions chez l'homme et les animaux", Paris, C.Reinwald, 1877, The University of Chicago Press, 1965, rééd.

[GOL 95] Goleman, David, "Emotional Intelligence"

[IMM 82] Immelmann, Klaus, Ruwet Anne, "Dictionnaire de l'éthologie", P. Mardaga, Bruxelles, 1982

[JAN 28] Janet, Pierre, "de l'angoisse à l'extase", Alcan, Paris, 1928

[LAM 14] Lamarck, Jean Baptiste Pierre Antoine de Monet de, "Connoissances Humaines", œuvre originale de 1814 rééd. : "inédits de Lamarck" d'après les manuscrits conservés à la bibliothèque centrale du Muséum National d'Histoire Naturelle de Paris

[LOR 70] Lorentz, Konrad, "Essais sur le comportement animal et humain : les leçons de l'évolution de la théorie du comportement", Editions du Seuil, Paris, 1970

[MIN 85] Minsky, Marvin, "The Society of Mind", Simon and Schuster, New York, 1985, "La société de l'esprit", InterEditions, Paris, 1988, trad.

[WAL 34] Wallon, Henri, "les origines du caractère chez l'enfant", Boivin, Paris, 1934, Presses Universitaires de France, Paris, 1949, rééd.

7 - Ouvrages complémentaires

Fraisse P. & Piaget J., "Traité de Psychologie Expérimentale" vol. 5, Presses Universitaires de France, Paris, 1963

Guyomarc'h Jean-Charles, "Abrégé d'éthologie", Masson, Paris, 1980

Harlow ,"love in Infant Monkeys"

James W., "What is emotion ?",Mind, 1884, 9, 188-205

Que Sais-Je "Les émotions" n° 2380

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Reconnaître les émotions des êtres humains

Jacques Pitrat
CNRS–LIP6
Université Pierre et Marie Curie

Résumé : Un système informatique qui interagit avec des êtres humains peut avoir besoin de connaître l'état émotif de son interlocuteur. Cela lui permet de l'aider en s'adaptant à ses émotions. Nous commençons par montrer les applications possibles de tels systèmes. Nous verrons ensuite que deux méthodes permettent d'évaluer un état émotif : en étudiant le comportement de l'interlocuteur ou en prédisant quelles doivent être ses émotions d'après les événements auxquels il a dû faire face. Cette dernière possibilité demande au système d'avoir une théorie des émotions humaines. Nous concluons en examinant les problèmes éthiques présentés par des systèmes capables de détecter les émotions humaines.

Mots-clés : émotions, EIAO, compréhension des langues naturelles.

1. Introduction

Un système qui interagit avec un utilisateur ou qui traite de documents où figurent des êtres humains doit avoir une connaissance de l'existence des émotions. Celles-ci ont en effet une grande importance dans notre comportement ; un système qui l'ignorerait serait totalement incapable de comprendre ou de prédire les réactions des êtres humains. Il est nécessaire de comprendre les raisons d'une action faite par un humain pour expliquer les décisions qu'il a prises. Cela est également utile pour prédire son comportement futur afin de l'incorporer dans le modèle du monde qui permet de planifier : on agit en fonction du comportement probable des humains avec lesquels on sera en contact. Un système doit enfin pouvoir aider les êtres humains en sachant éviter que leurs émotions aient des répercussions défavorables pour eux ; par ailleurs, un système peut se servir de leurs émotions pour qu'ils réussissent mieux les tâches qu'ils souhaitent accomplir. Un tel système évitera de mettre en colère son élève ou calmera sa colère, il saura l'encourager quand il est démoralisé. Ce système aidera mieux son utilisateur à apprendre que s'il ne tenait aucun compte des émotions ressenties par l'élève.

Nous commencerons par indiquer des applications possibles de systèmes capables de reconnaître les émotions des humains. Nous verrons ensuite comment on peut représenter les émotions. Nous examinerons les deux méthodes permettant de trouver les émotions d'un être humain : observer son comportement ou prédire les émotions grâce à une connaissance du sujet et des événements qui se sont produits. Nous parlerons enfin des problèmes d'éthique qui se présenteront quand les systèmes seront assez performants pour découvrir les émotions des êtres humains.

2. Quelques applications possibles de systèmes découvrant les émotions humaines

Les réalisations concrètes sur la découverte et l'utilisation des émotions humaines sont encore très rares. Beaucoup de chercheurs étudient la modélisation des utilisateurs, par exemple il y a déjà eu six conférences UM (User Modeling) sur ce sujet. Mais ils s'intéressent aux connaissances, aux préférences, aux comportements typiques des utilisateurs plutôt qu'à leurs émotions. Toutefois, depuis peu, quelques projets liés aux émotions sont en cours. Nous allons examiner quelques applications potentielles de ces recherches.

2.1. Communication homme-machine

Un système devrait avoir une idée des émotions que ressent son utilisateur pour qu'il agisse en conséquence. Par exemple, un système d'exploitation ne rend pas toujours les services que l'on attend de lui. S'il se rendait compte d'une certaine impatience de la part de l'utilisateur, il serait plus agréable pour celui-ci qu'il commence par s'excuser. Il passerait ensuite la main à une procédure capable de remédier aux inconvénients qui ont entraîné cette insatisfaction. Nous évaluons constamment l'humeur de ceux que nous rencontrons et nous adaptons notre comportement à ce que nous décelons. Par exemple, nous évitons ce qui pourrait heurter quelqu'un qui est triste. Pourquoi les ordinateurs n'en feraient-ils pas autant avec nous ?
Cela est particulièrement important pour les systèmes d'EIAO. L'intérêt de l'élève pour ce qu'on lui propose est un facteur crucial pour qu'il apprenne. Si l'on se rend compte que celui-ci s'ennuye ou manque d'enthousiasme, il faudrait reviser la méthode d'apprentissage choisie. Il faudrait aussi deviner les frustrations du sujet, qui peuvent se manifester par la colère. Une fois qu'une émotion a été détectée, il s'agit ensuite d'en trouver la cause. Une émotion traduit souvent une difficulté de l'élève qu'il est important de déceler.

2.2. Monitorer en temps réel les êtres humains

Monitorer les êtres humains en temps réel est une application potentielle qui n'existe encore qu'à l'état de projet. Dans la section précédente, un utilisateur se connecte avec un ordinateur dans un but bien précis. Ici, l'humain serait connecté en permanence avec un ordinateur qui l'aiderait à passer les moments difficiles. L'ordinateur s'en rendrait compte en détectant les émotions ressenties par son partenaire humain. Par exemple, il lui jouerait une musique calmante quand celui-ci aurait tendance à être excité ou une musique joyeuse quand celui-ci serait triste. Il pourrait également dialoguer avec lui pour l'aider à surmonter ses problèmes. L'ordinateur serait dans cette optique une sorte d'ange gardien qui serait toujours à l'écoute ; il serait prêt à agir en cas de difficulté. Ceci est un des buts du projet Affective Computing du MIT [Affective Computing 1997].

2.3. Mieux connaître un être humain

On veut ici découvrir le caractère d'un humain, et non plus son état émotif instantané. Pour cela, le système devrait trouver des patterns dans son comportement émotif. Il pourrait voir qu'il se décourage facilement, qu'il se met souvent en colère, qu'il s'intéresse à peu de choses…A partir de cela, il pourrait prédire ou conseiller en tenant compte des forces et des faiblesses du caractère de son sujet. Il serait également possible de prévoir les difficultés qui pourraient se présenter. Comme dans la section précédente, il s'agit seulement de projets et il n'y a pas à ma connaissance de réalisation dans cette voie.

2.4. Compréhension de textes exprimés dans une langue naturelle

Un système qui doit comprendre un texte a besoin de comprendre les raisons des actions des personnages qui y figurent. Considérons l'histoire suivante :
Pierre aimait la fille du Roi. Un jour, un dragon enleva la princesse. Pierre prit son cheval et la délivra du dragon. Le Roi lui donna la main de sa fille.
Nous voyons apparaître implicitement un très grand nombre d'émotions dans cette histoire :
— l'amour entre un homme et une femme
— l'amour d'un père pour sa fille
— la peur ressentie par la princesse
— la joie de la princesse quand Pierre vient la délivrer
— la tristesse du Roi et de Pierre quand ils ont appris l'enlèvement de la princesse
— l'anxiété du Roi qui attend la délivrance de sa fille
— la fierté de Pierre quand il a réussi à arracher la princesse au dragon
— l'admiration de la princesse pour Pierre
— le soulagement du Roi quand il a appris que sa fille était délivrée

— la reconnaissance du Roi pour ce qu'a fait Pierre

……

Une seule de ces émotions apparaît explicitement dans le texte : l'amour de Pierre pour la princesse. Pourtant, nous ne pouvons dire que l'on a compris ce texte si l'on n'y trouve aucune autre des émotions précédentes.

Dans cet exemple, la connaissance des émotions humaines est utile pour :

— compléter ce qui manque pour créer la représentation interne du sens du texte. On va ainsi inférer que Pierre a épousé la princesse bien que cela ne soit pas dit explicitement. Cela est en accord avec ce que l'on sait des émotions probablement présentes et des comportements qui y sont associés.

— lever les ambiguïtés. A qui le Roi donna-t-il la main de sa fille ? L'élément le plus proche qui soit du même genre et nombre que le pronom "lui" est le dragon. Cette référence doit être rejetée, car elle est totalement en désaccord avec les émotions probables des personnages qui ne doivent ressentir que de la haine pour le dragon.

— maîtriser la génération des inférences. Engendrer toutes les inférences possibles est un phénomène explosif, les inférences faites conduisant à de nouvelles inférences. La connaissance sur les émotions des humains peuvent écarter des inférences parce que trop peu probables. On ne va par exemple pas inférer que le Roi donne la main de sa fille à Pierre parce qu'il estime qu'il est utile d'avoir dans sa famille un tueur de dragons. Cette inférence est possible, mais il y en a une plus plausible basée sur la reconnaissance.

De tels travaux ont été développés par Schank et ses élèves [Schank 1977]. Ils sont un peu délaissés actuellement : les chercheurs travaillant sur les langues naturelles préfèrent s'intéresser aux textes techniques où les émotions n'apparaissent guère.

3. Représenter les émotions

Deux valeurs sont liées à toute émotion humaine. La première est la valeur instantanée d'une émotion pour un individu, par exemple il est en colère. La deuxième est la valeur d'une prédisposition permanente d'un individu pour une émotion : il est colérique.

3.1. Représentation de la valeur instantanée d'une émotion

Nous pouvons représenter soit l'état actuel (il est en colère), soit l'existence d'un changement d'état (il se met en colère).

On représente souvent un individu par un ensemble de couples (attribut – valeur de l'attribut). Il suffit dans ce cas d'ajouter des couples supplémentaires : le premier élément décrit la nature de l'émotion et le deuxième élément indique la valeur de cette émotion pour l'individu en question.

Par exemple, à côté de (nom : Dupont, taille : 175cm), nous pouvons avoir (colère : élevée, joie : faible). C'est ainsi que Schank [Schank 1975], dans la théorie de la dépendance conceptuelle propose de représenter les émotions suivantes avec les valeurs associées qui sont ordonnées :

peur : terrified, scared, anxious, calm.

colère : furious, enraged, angry, irked, upset, calm.

état mental : catatonic, depressed, upset, sad, OK, pleased, happy, ecstatic.

dégoût : nauseated, revolted, disgusted, bothered.

surprise : surprised, amazed, astounded.

Il est impossible de traduire les valeurs de ces émotions, la gamme n'étant pas la même d'une langue à une autre. La valeur de ces émotions est représentée de la même façon que la faim (qui va de "starving" à " satiated") ou la santé (qui va de "dead" à "perfect health").

La dépendance conceptuelle permet aussi de représenter des changements d'état. Par exemple, la représentation de "to want" indique que la réalisation de l'action liée au complément direct de ce verbe va faire passer la joie du sujet du verbe d'une valeur x à une valeur x+2. C'est ainsi que "John wants a book" va être représenté en indiquant que le transfert de la propriété d'un livre à John va augmenter la joie de John. De même, "to annoy" (John annoyed Mary) indique une augmentation de la valeur de l'émotion "colère" chez l'humain complément direct (Mary) qui est due à l'objet sujet (John).

3.2. Représentation d'une tendance émotionnelle

Certains individus ont tendance à manifester une émotion particulière alors que d'autres ne présentent qu'exceptionnellement cette même émotion. Un colérique peut ne pas être en colère à un instant donné, mais il se met souvent en colère. Il est intéressant de représenter ces traits de caractère de façon à prédire le comportement d'un individu. Je ne connais pas de systèmes qui représentent ainsi les traits de caractères, mais cela peut aussi se faire à l'aide de couples (émotion–tendance à présenter cette émotion) qui figureraient aussi dans la description de l'individu.

Le système GRUNDY [Rich 1979] détermine les traits de caractère d'une personne qui souhaite emprunter un livre dans une bibliothèque. C'est ainsi qu'il évalue son éducation, son sérieux, sa piété, son intérêt pour la romance, pour le féminisme, sa tolérance vis-à-vis du sexe et de la violence, sa gentillesse… Curieusement, il n'y figure aucune émotion de façon explicite, mais il serait facile de compléter cette description par des attributs qui mentionneraient sa tendance à manifester diverses émotions. La représentation utilise des quadruplets :

— nom de l'attribut

— valeur de l'attribut

— confiance que l'on a dans l'exactitude de cette valeur. Si la confiance est de 1000, on est certain que l'attribut a cette valeur, si elle est de 100, c'est extrêmement douteux.

— justification de la détermination de cette valeur. On pense avec une assez grande certitude que Madame Dupont est pieuse parce que c'est une femme. Par contre, on pense que Madame Durand n'est pas pieuse parce que c'est une intellectuelle féministe, mais la certitude est moyenne, parce qu'elle est quand même une femme.

La représentation des traits de caractère liés aux émotions ne présente pas de difficulté. Le problème vient de la difficulté à évaluer ces traits de caractère, bien plus difficiles à déterminer que des valeurs instantanées : il faut observer le sujet sur une longue période pour trouver des valeurs permanentes.

4. Découvrir par observation les émotions d'un individu

Par observation d'un individu, nous sommes capables de découvrir qu'il ressent des émotions. S'il devient tout rouge, s'il utilise un vocabulaire grossier, s'il parle fort, nous jugeons qu'il est en colère. Il est possible de donner à un système informatique des informations sur l'utilisateur qui lui donnent la possibilité de découvrir que celui-ci a une certaine émotion. La difficulté est que ce système soit capable d'interpréter ces informations. Nous allons commencer par examiner quels types d'informations liées aux émotions peuvent être fournies à un ordinateur.

Le Prototype Sensing System du MIT [Affective Computing 1997] est connecté à un individu et fait en permanence les mesures suivantes :

— la réponse électro-dermale de la peau. La résistance électrique de la peau dépend de la transpiration qui est elle-même augmentée ou diminuée pour certaines émotions. Le détecteur de mensonges utilise depuis longtemps cette propriété.

— l'électro-myogramme mesure l'activité électrique qui accompagne les contractions musculaires.

— la pression sanguine.

— la fréquence respiratoire.

— la température du corps.

Il serait possible d'utiliser bien d'autres éléments qui donnent des renseignements sur l'état émotif :

— l'expression du visage. Mais les sujets sont souvent habitués à contrôler leur traits.

— les gestes. Les sujets contrôlent beaucoup moins les mouvements de leurs mains ou de leurs pieds.

— la modulation de la voix.

— le langage. Il arrive qu'un utilisateur ne soit pas content des prestations d'un système informatique censé dialoguer en utilisant une langue naturelle. L'expérience montre que cet utilisateur manifeste vivement son mécontentement par les termes qu'il utilise.

— les manifestations hormonales qui accompagnent de nombreuses émotions.

On n'utilise pas ces données parce que l'on ne sait pas encore les déceler avec précision. Mais cela peut changer dans un proche avenir. Bientôt, nous aurons des systèmes qui pourront analyser les expressions d'un visage. Une réalisation récente, dont le but était tout à fait différent, montre qu'il sera aussi possible d'étudier les gestes avec précision. En effet, Holden, Owens et Roy [Holden 1996] ont réalisé un système qui interprète le langage des mains des sourds-muets. Il faut pour cela être capable de percevoir la position exacte des différents doigts. Il faut aussi observer les mouvement très rapides qu'ils font quand ils passent d'une position à une autre. En effet, ces mouvements, par exemple un frétillement rapide de l'index, apportent de l'information. Observer les mouvements des mains est donc une tâche que l'on commence à savoir faire.

Comme nous l'avons remarqué plus haut, la principale difficulté est de relier automatiquement une mesure avec une émotion. Les expériences faites avec le Prototype Sensing System permettront sans doute de trouver des méthodes qui permettront de le faire sans erreur.

Le système précédent n'est porté par le sujet que quand il participe à une expérience. Un autre projet, celui du "Wearable Computer" [Affective Computing 1997] est beaucoup plus ambitieux. Il s'agit d'un ordinateur que nous porterions en permanence ; son but est d'augmenter l'efficacité et la qualité de la vie de celui qui le porte. Ce n'est pas un ordinateur portable, mais une partie de l'habillement. Cet ordinateur doit analyser en temps réel les capteurs portés par son hôte qui mène une vie normale. Dans sa version actuelle, l'ordinateur mesure 14x14x8 cm^3. Il comporte des capteurs affectifs, comme ceux que nous avons vus pour le système précédent, mais aussi des moyens plus normaux de communication. Il reçoit des informations et des ordres par l'intermédiaire d'un microphone et d'un "twiddler", petit clavier manipulable par une seule main. Les caractères sont produits par une combinaison de touches ; un sujet entraîné peut entrer jusqu'à 50 mots par minute. Il répond et donne des informations à son porteur par l'intermédiaire d'un écouteur et d'un "private eye". Ce dernier est positionné en face d'un seul œil. Il comporte une seule ligne de diodes, mais un miroir tournant donne l'illusion de voir tout un écran. Avec un tel système, l'homme sera accompagné d'un ange gardien qui détectera toutes ses humeurs. Il l'apaisera quand nécessaire par des paroles réconfortantes ou de la musique douce, il lui remontera le moral, il le conseillera. Bien sûr, tout cela n'en est qu'à ses débuts, mais cela montre l'ambition et l'intérêt que présenteront de tels systèmes le jour où l'on saura effectivement reconnaître les émotions humaines.

5. Supposer quelles sont les émotions d'un individu

Alors que dans la section précédente on mesure réellement les émotions d'un individu, on doit seulement ici deviner ce qu'elles peuvent être quand on sait qu'il a eu à faire face à certains événements. Si nous apprenons que quelqu'un a raté un examen auquel il tenait beaucoup, nous

savons, sans avoir besoin de le voir, qu'il est triste, qu'il n'a pas le moral. Il faut pour cela avoir une théorie générale des émotions et une connaissance du caractère du sujet. Nous avons vu que la détermination du caractère des individus n'est pas encore faite de façon satisfaisante par un système automatique. Aussi prend-on un caractère par défaut, celui de l'individu moyen, qui est en général défini de façon implicite. Plusieurs travaux ont pour but de donner la possibilité de prédire les émotions d'un sujet.

5.1. Des règles pour prédire l'état émotionnel d'un individu

Sachant qu'un individu a dû faire face à un certain événement, une règle peut donner la valeur probable d'une de ses émotions. Par exemple, le système de B.Roger [Roger 1987] apprenait à un utilisateur à jouer à la guerre des étoiles. Un paramètre "joie" est associé à l'utilisateur. Elle augmente à chaque événement heureux, par exemple la capture d'un vaisseau klingon. Elle diminue chaque fois que l'utilisateur se rend compte qu'il a commis une erreur, par exemple une erreur de navigation. Le système se servira ensuite de la valeur de ce paramètre pour aider l'utilisateur. Si cette joie atteint un niveau trop faible, il le félicitera dès qu'il aura réussi quelque chose :

> Bravo, vous commencez à bien savoir que lorsqu'il y a plusieurs klingons, il vaut mieux utiliser les torpilles.

Le système de Colby [Colby 1971] simule le comportement d'un paranoïaque. Son modèle contient la valeur de deux émotions : la crainte et la colère. Si, à l'instant i, le paramètre donnant la valeur associée à une de ces émotions est N_i, la valeur après un échange verbal susceptible de l'augmenter est :

$$N_{i+1} = N_i * (20-N_{i-1})$$

Après un échange normal, la valeur des deux paramètres est diminuée de 0.3.

Pfeifer [Pfeifer 1985] a utilisé des règles pour prédire le comportement d'un individu. Par exemple :

> Si l'état est négatif pour Soi
>
> Si l'état a été causé par Autre
>
> Si Autre était en contrôle
>
> Alors engendrer la colère envers Autre.

Pour sa part, Worden [Worden 1996] modélise le comportement émotionnel des primates. Quand un jeune singe voit une menace potentielle, comme un gros oiseau au-dessus de lui, il regarde un singe adulte et expérimenté et calque son comportement sur celui-ci. En effet, l'évolution n'a pas donné aux jeunes singes une connaissance innée pour distinguer entre un gros oiseau dangereux comme un aigle et un qui n'est pas dangereux comme un vautour. Cela donne les deux règles suivantes :

Si une scène peut présenter un danger

Si Autre, plus expérimenté, a peur

Alors Soi a peur.

Si une scène peut présenter un danger

Si Autre, plus expérimenté, n'a pas peur

Alors Soi n'a pas peur.

Remarquons que ces deux dernières règles supposent que le jeune singe sait reconnaître les émotions d'un autre individu.

En appliquant de telles règles, on peut prédire le comportement émotif d'un individu dans une situation donnée. Les règles précédentes ne contiennent pas de prémisses liées au caractère d'un individu particulier. Celle du paranoïaque ne sont valables que pour les individus qui ont cette maladie et les autres s'appliquent à un individu moyen. Il serait possible d'adapter ces règles à des familles resteintes d'individus en ajoutant des prémisses du type "Si le sujet est craintif".

5.2. Des règles pour changer l'état émotif d'un individu

Dans la section précédente, les règles servent à prédire les émotions. Ici, nous voulons donner au système la possibilité de modifier les émotions de l'utilisateur. Il y a eu très peu d'essais dans cette voie. Nous avons vu le travail de B.Roger [Roger 1987] qui évalue la joie probable de son utilisateur. Si elle est trop basse, le système le félicite dès qu'il a réussi quelque chose.

Le système Pauline [Hovy 1990] crée des textes devant représenter en anglais un ensemble d'informations. Ces textes peuvent avoir différents buts ; l'un de ceux-ci est de mettre son interlocuteur en colère. Parmi les paramètres que le système évalue figure la formalité. S'il doit mettre son interlocuteur en colère, il prend la valeur opposée de la formalité qu'il devrait normalement utiliser. En adaptant cela au français, s'il s'adresse à une personne qu'il devrait respecter, il la tutoie et lui dit "mon vieux". Inversement, si un mari au cours d'une scène de ménage se met à vouvoyer sa femme et lui donne du "chère madame", cela ne va certainement pas calmer le jeu.

5.3. Une définition formelle des émotions

Une autre solution est de définir formellement les émotions. On peut ainsi mesurer si une certaine émotion est présente quand le sujet est soumis à certains événements. Wierzbicka [Wierzbicka 1992] a ainsi défini un grand nombre d'émotions. Nous reprenons ici sa définition pour "ashamed" :

Tout le monde peut connaître quelque chose de mauvais à mon sujet.

A cause de cela, tout le monde pourra penser quelque chose de mauvais à mon sujet.

Je ne le veux pas.

Comme je ne le veux pas, je voudrais faire quelque chose.

Je ne sais pas ce que je pourrais faire.

Je voudrais que personne ne soit au courant de tout cela.

En raison de ce qui précède, je ressens quelque chose de mauvais.

Remarquons que nous ne sommes pas toujours responsables de ce quoi nous avons honte. Nous pouvons avoir honte de nos origines ou de nos parents.

5.4. La théorie de Schank

Il est impossible de comprendre correctement un texte où des êtres humains apparaissent si l'on ne dispose pas d'une théorie des émotions. Aussi R. Schank [Schank 1977] a-t-il proposé une théorie pour évaluer les émotions des personnages et les conséquences qu'elles ont sur leur comportement.

Un des éléments de base de sa théorie est le thème. Un thème représente une prédisposition d'un acteur. Ils permettent d'inférerer que cet acteur aura un certain but si un thème est présent.

Il existe trois sortes de thèmes :

— les thèmes liés à un rôle : docteur, éboueur, boulanger, policier…

— les thèmes entre personnes : père-fils, amoureux, rivaux, collègues, époux, patron-employé…

— les thèmes généraux de la vie qui peuvent être présents chez tout humain : être riche, faire quelque chose d'utile, vivre luxueusement, être honnête…

Si un thème est présent et si certains événements se sont produits, alors on peut en déduire qu'il est probable que les personnages auront certaines réactions et en particulier qu'ils auront certaines émotions. Remarquons qu'il s'agit de comprendre un texte ; il est parfaitement possible que certains individus se comportent de façon atypique. Mais alors, l'auteur doit explicitement indiquer que quelqu'un a eu un comportement anormal. Les thèmes permettent de faire des inférences naturelles, qui sont des sortes de valeurs par défaut. Nous avons le droit de penser que quelqu'un qui est ruiné va être triste, car cela conduit à la disparition du thème général "être riche". L'auteur n'a pas besoin de le dire, nous le savons tous. Par contre, s'il se trouve qu'un personnage est content d'être ruiné, l'auteur doit le dire. La connaissance sur le caractère normal d'un individu permet de s'exprimer de façon concise.

La connaissance des thèmes permet d'inférer certains buts. Par exemple :

Si deux individus sont liés émotionnellement de façon positive

Alors la reconnaissance d'un changement négatif chez l'un d'eux entraîne chez l'autre le but d'empêcher ce changement.

C'est ainsi que nous n'avons pas de peine à comprendre pourquoi Pierre prend son cheval quand il apprend que la princesse a été enlevée par un dragon.

6. Problèmes liés à l'étude des émotions par des systèmes informatiques

Les contraintes qui doivent porter sur les systèmes informatiques traitant des émotions sont de deux natures. D'une part, pour mettre au point les systèmes qui analyseront les paramètres mesurés sur un individu, il faudra réaliser des expériences où des sujets seront mis dans divers états émotifs. D'autre part, les systèmes auront des informations très personnelles sur les sujets qu'ils observeront de façon très indiscrète ; il s'agit de protéger ces informations. Pour ces deux raisons, le projet Affective Computing [Affective Computing 1997] insiste sur les contraintes qu'il s'agit de respecter et que nous allons résumer ici.

6.1. Contraintes sur les expériences

Pour voir les réactions d'un sujet quand il est dans un certain état émotif, il faut s'arranger pour qu'il ressente cette émotion. Doit-on alors lui faire peur, le mettre en colère, le rendre triste ? Cela pose évidemment un problème ; il faut manifestement prendre des précautions. Le groupe de recherche Affective Computing [Affective Computing 1997] a pris les engagements suivants :

— Le sujet a tout pouvoir sur la divulgation d'informations qui pourraient porter atteinte à son anonymat.

— Comme certaines études peuvent délibérément induire des émotions, les participants doivent savoir qu'ils peuvent retirer à tout moment leur consentement et interrompre l'expérience.

— Chaque participant a l'option d'être mis au courant de toute information qu'une expérience pourrait révéler sur lui, que l'expérimentateur soit ou non capable d'interpréter cette information.

Naturellement, il va sans dire que le sujet doit savoir qu'il participe à une expérience avant qu'elle ne commence. Nous devons éviter les situations qui se produisent parfois dans les émissions de caméra invisible : nous sommes gênés parce que le sujet passe par des états où il n'est pas à l'aise avant d'apprendre que ce n'était qu'un coup monté pour les besoins de l'émission.

6.2. Contraintes sur le système

Le principe de base est que l'information sur les émotions appartient uniquement à celui qui les ressent. Aussi est-il indispensable de la protéger :
— un firewall doit être établi en utilisant les ressources matérielles et logicielles autour de chaque système. Cela doit empêcher l'accès à toute information sur la vie émotionnelle du sujet sans son accord.
— toutes les données liées aux émotions d'un sujet doivent être cryptées.
— les procédures de communication doivent garantir que personne ne pourra intercepter les informations émotionnelles sur un sujet.

A première vue, certains pourraient penser que toutes ces précautions relèvent de la paranoïa. Pourtant quand on réfléchit à l'importance que les émotions ont dans notre vie, il est évident qu'il s'agit là de données extrêmement sensibles. Certes, les systèmes actuels ne sont pas encore capables de bonnes performances quand il s'agit d'analyser les émotions humaines. Cela ne doit pas nous empêcher de prendre de bonnes habitudes.

7. Conclusion

Les recherches dans le domaine de la prédiction ou de la mesure des états émotionnels humains sont encore embryonnaires. Elles seront pourtant appelées à un grand avenir quand elles aboutiront parce que :
— ce sera un plaisir d'être en contact avec des systèmes informatiques attentifs à nos humeurs.
— certains systèmes "anges gardiens" pourront aider leur patient à surmonter des passes difficiles et de lui rendre en permanence la vie plus agréable. La question est de savoir si ce sera un bon substitut au Prozac.
— un système de compréhension des langues naturelles doit disposer d'une théorie des émotions humaines s'il traite des textes qui ne sont pas techniques.

Références

[Affective Computing 1997] Site Web du Affective Computing Research Group dirigé par Rosalind Picard, http://vismod.www.media.mit.edu/vismod/demos/affect/ .
[Colby 1971] Colby K., Weber S. and Hilf F., Artificial Paranoia, *Artificial Intelligence 2*, 1-25, 1971.

[Holden 1996] Holden E., Owens R. and Roy G., Hand Movement Classification Using an Adaptive Fuzzy Expert System, *International Journal of Expert Systems 9*, 465-480, 1996.
[Hovy 1990] Hovy E., Pragmatic and Natural Language Generation, *Artificial Intelligence 43*, 153-197, 1990.
[Pfeifer 1985] Pfeifer R. and Nicholas D., Toward Computational Models of Emotions, *Progress in Artificial Intelligence*, Steels and Campbell Ed., 184-192, Ellis Horwood, 1985.
[Rich 1979] Rich E., User Modeling via Stereotypes, *Cognitive Science 3*, 329-354, 1979.
[Roger 1987] Roger B., *Un système de dialogue intelligent avec un interlocuteur à la découverte du monde simulé par un logiciel quelconque*, Thèse de l'Université Paris 6, 1987.
[Schank 1975] Schank R., *Conceptual Information Processing*, North-Holland, 1975.
[Schank 1977] Schank R., *Scripts, Plans, Goals and Understanding*, Lawrence Erlbaum, 1977.
[Wierzbicka 1992] Wierzbicka A., Defining Emotion Concepts, *Cognitive Science 16*, 539-581, 1992.
[Worden 1996] Worden R., Primate Social Intelligence, *Cognitive Science 20*, 579-616, 1996.

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Les "machines à construire" : des interfaces dans le projet COMBIEN? Conception, spécification et réalisation

Groupe COMBIEN

Gérard Tisseau[1], Jacques Duma[2], Hélène Giroire[1],
Françoise Le Calvez[3], Marie Urtasun[3]

Résumé : Dans le cadre du projet COMBIEN?, nous présentons un Environnement Interactif d'Apprentissage avec Ordinateur (EIAO) dans le domaine des dénombrements. Les interfaces de ce logiciel se présentent sous la forme de "machines à construire des configurations combinatoires". A l'aide d'une telle interface, un élève peut modéliser un problème de dénombrement et construire une configuration solution, et cette activité est ensuite exploitée comme support pour l'apprentissage des dénombrements. Nous donnons d'abord des exemples de machines et de leur utilisation. Puis nous étudions les concepts et les outils nécessaires à la conception, la spécification et la réalisation de telles machines, en les replaçant dans le cadre plus général des logiciels interactifs. Nous introduisons un mécanisme original de gestion de la disponibilité des actions offertes à l'utilisateur, fondé sur les concepts de variables, actions, contextes et modes de déclenchement.

Mots-clés : EIAO, logiciel interactif, interface homme-machine, spécification d'interfaces.

1.Introduction

Le projet COMBIEN? a pour but la réalisation d'un Environnement Interactif d'Apprentissage avec Ordinateur (EIAO) dans le domaine des dénombrements. Les problèmes de ce domaine se

[1] LIP6 Pôle IA, Université Pierre et Marie Curie, Paris.
[2] Lycée Jacquard, Paris.
[3] Jeune équipe Informatique et Communication, Université René Descartes.

ramènent tous à : “Combien y a-t-il d'éléments d'un univers U, vérifiant certaines contraintes ?”. Une des difficultés de ces problèmes est de définir l'*univers* U et les *contraintes*, ce qui nécessite de modéliser les situations étudiées. Considérons l'énoncé : “Combien y a-t-il de mains de cinq cartes contenant deux piques et trois coeurs ?”. L'univers U est l'ensemble des mains de cinq cartes, il est construit à partir de l'ensemble des cartes que nous appelons le *référentiel* du problème. Nous avons défini une méthode constructive de résolution qui consiste à énoncer un algorithme d’énumération des éléments de l’ensemble à dénombrer et à raisonner sur cet algorithme sans l’exécuter [Tisseau et al. 96].
Le système que nous construisons dans le projet COMBIEN? doit répondre à certaines exigences. Il doit d’abord être capable non seulement de proposer à l’élève des problèmes mémorisés dans une base mais aussi d’en recevoir de nouveaux, saisis par l’élève. Il doit ensuite être lui-même capable de résoudre ces problèmes par la méthode constructive et d'expliciter sa solution. Enfin, il doit être capable de recevoir une proposition de solution constructive pour déterminer sa validité. Nous avons mis au point un système d'IA répondant en partie à ces exigences, et comprenant actuellement un résolveur de problèmes, un évaluateur de solutions constructives et des interfaces. Ce résolveur [Guin 95] est fondé sur une classification des problèmes du domaine.
Le prototype réalisé est destiné à servir de noyau au système d'EIAO que nous réalisons. La progression pédagogique que nous proposons est dans une première phase la découverte de la méthode constructive. Nous avons créé pour cet enseignement des “machines à construire des objets” permettant aux élèves de se forger des représentations adaptées et de se familiariser avec l’utilisation de cette méthode [Le Calvez et al. 97].

Dans cet article, nous présentons dans une première partie certaines de ces machines (actuellement sous forme de prototypes) du point de vue de leur utilisation et de leur exploitation pédagogique. Puis, dans une deuxième partie, nous étudions les concepts et les outils nécessaires à la conception, la spécification et la réalisation de telles machines, en les replaçant dans le cadre plus général des logiciels interactifs. Nous discutons différentes approches que nous comparons à la nôtre, en faisant ressortir les spécificités d’un EIAO. Nous introduisons un mécanisme original de gestion de la disponibilité des actions offertes à l’utilisateur.

2. Les machines à construire des configurations combinatoires

L'utilisation de notre système d’IA suppose déjà une certaine familiarité avec les concepts mathématiques et avec la méthode constructive de dénombrement. Pour aider l'élève à acquérir préalablement cette familiarité, nous lui proposons de s'entraîner en utilisant des “machines à construire”, qui matérialisent certains schémas de constructions associés aux classes de

problèmes courantes. Nous décrivons ces machines dans ce qui suit. Nous les avons conçues pour placer l'élève dans des situations d'interaction où il peut observer et manipuler les objets intervenant dans la méthode constructive. Le modèle conceptuel de cette méthode n'est pas présenté de manière théorique, mais sous une forme concrète (une machine), ce qui introduit plusieurs niveaux de modélisation (comme dans [Bernat et al. 95]). Certaines modalités d'interaction contraignent l'élève à affronter les difficultés propres à la méthode, ce qui est courant en EIAO [Dubourg et al. 95].

2.1 Description générale

Ces “machines” se présentent à l'écran sous forme de fiches interactives, qui correspondent chacune à une classe de problèmes de dénombrement. Elles fournissent des outils pour construire certains objets mathématiques structurés : relations, applications, ensembles et listes. L'élève utilise ces machines pour produire un objet vérifiant des spécifications données, par exemple un mot de cinq lettres contenant exactement deux a.
Le contrôle de l’activité est partagé entre le système et l’élève. Le système détermine à chaque étape quelles possibilités d’action laisser à l’élève (quels boutons et menus sont activables), quelles informations afficher (état des variables, listes de choix). Pour cela, il tient à jour de manière interne des variables indiquant l’état d’avancement de la construction et vérifie la cohérence des actions et des choix de l’élève. Les activités conjointes du système et de l’élève concourent à exécuter un algorithme de construction, chaque machine étant associée à un certain algorithme qui est inscrit dans sa structure et son comportement. L'élève suit une progression que nous avons voulu contrôlée, car il est dans une situation d'apprentissage (nous discuterons les particularités d’une telle situation après les exemples de machines).

2.2 Utilisation

Nous allons suivre la progression d'un élève sur une machine pour en montrer l'utilisation. Nous avons choisi pour cela un exemple sur la quinzaine de machines conçues : la machine "Construction d'ensembles". Elle correspond à une classe de problèmes dans lequel l'élément de l'ensemble à dénombrer est un ensemble satisfaisant certaines contraintes. Par exemple : Dans un jeu de 32 cartes (*le référentiel*), combien y a-t-il de mains de 5 cartes (l'*univers* est l'ensemble des mains de 5 cartes) contenant deux piques et deux coeurs (*les contraintes*) ? La machine permet de construire une main, c’est-à-dire un élément de *l'ensemble à dénombrer* : l’ensemble des mains de 5 cartes contenant 2 piques et 2 coeurs.
La progression comporte quatre étapes : choix de l’exercice, paramétrage du problème, paramétrage et tirage d’objets, diagnostic.

Première étape : choix de l'exercice

Au départ, la machine présentée à l'élève lui permet seulement de choisir dans la liste des exercices celui qu'il veut résoudre. L'énoncé choisi est affiché et la deuxième partie de la machine devient accessible.

Deuxième étape : paramétrage du problème

L'élève choisit parmi les *référentiels* proposés, celui qui selon lui correspond à l'exercice. Il définit ensuite les paramètres de l'*univers,* ici le paramètre taille (taille = 5 pour une main de 5 cartes). Le bouton *valider* lui permet de fixer ses choix et d'accéder à l'étape suivante (voir la figure 1).

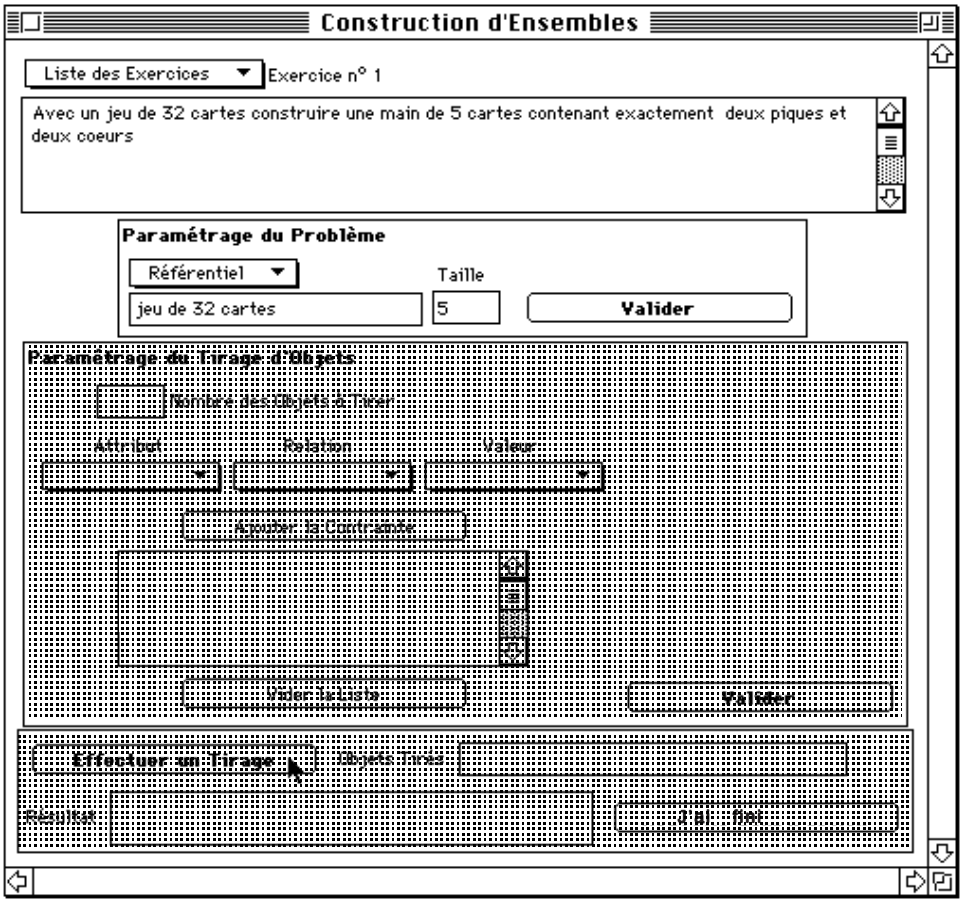


Figure 1

Troisième étape : Paramétrage et tirage d'objets

Cette étape consiste en la création d'un élément de l'ensemble à dénombrer qui sera visualisé dans la case *résultat*.. Ici cet élément sera composé de cinq cartes (les objets tirés) qui satisfont aux contraintes “deux piques, deux coeurs et une non-pique, non-coeur”. Les différentes zones dans cette étape deviennent disponibles au fur et à mesure. Le bouton *valider* va rendre accessible le bouton *effectuer un tirage*.

Dans cet exemple, l'élève définit et effectue trois tirages, le premier lui a permis d'obtenir 2 cartes de couleur pique, le second deux cartes de couleur coeur, et le troisième une carte de couleur différente de pique et de coeur. Le résultat de ces différents tirages sont cumulés dans

la case *résultat*.. La figure 2 correspond à ce que voit l'élève après avoir effectué le troisième tirage.

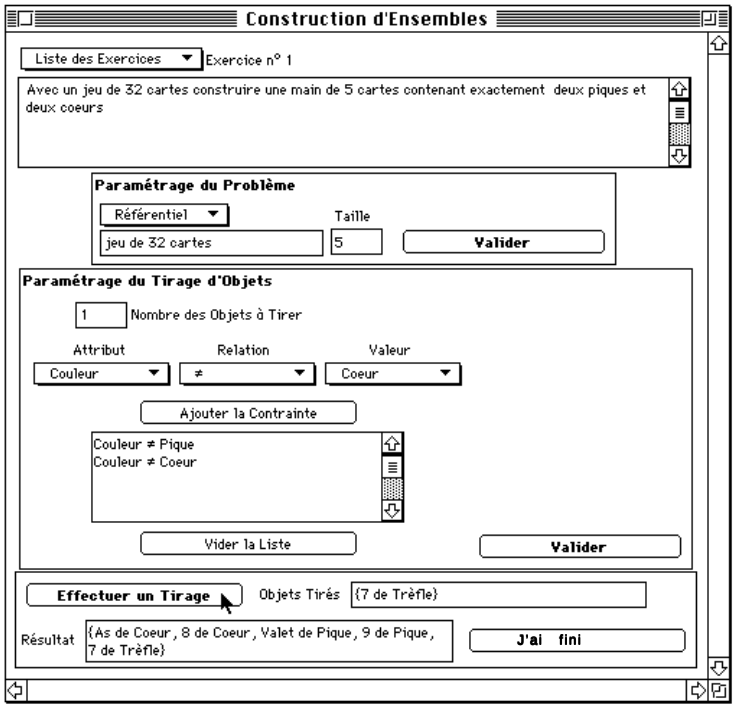


Figure 2

Quatrième étape : diagnostic

L'élève soumet sa construction au système en appuyant sur le bouton “J'ai fini”. Celui-ci lui renvoie une fenêtre visualisant l'algorithme de la méthode constructive qu'il a employée et lui rendant un diagnostic (cf. figure 3).

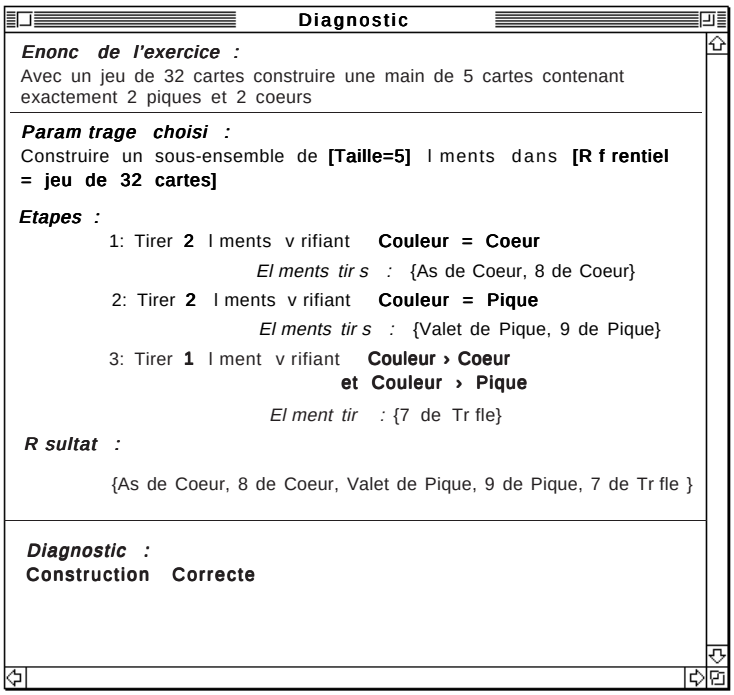


Figure 3

La notion de “diagnostic”, c’est-à-dire essentiellement de détection d’erreurs, peut prendre plusieurs significations suivant ce qu’on considère être une erreur ou non et suivant les types d’erreurs qu’on distingue. Nous discutons plus loin des différents types d'erreurs dans ce contexte d'apprentissage. Ici, dans le cas d'une mauvaise modélisation du problème, (taille de l'élément à former = 6), le système le détecte, mais laisse l'élève faire une construction et ce n'est qu'au niveau de “J'ai fini” qu'il obtiendra le diagnostic. Si cette construction est cohérente la fenêtre obtenue est la suivante (figure 4):

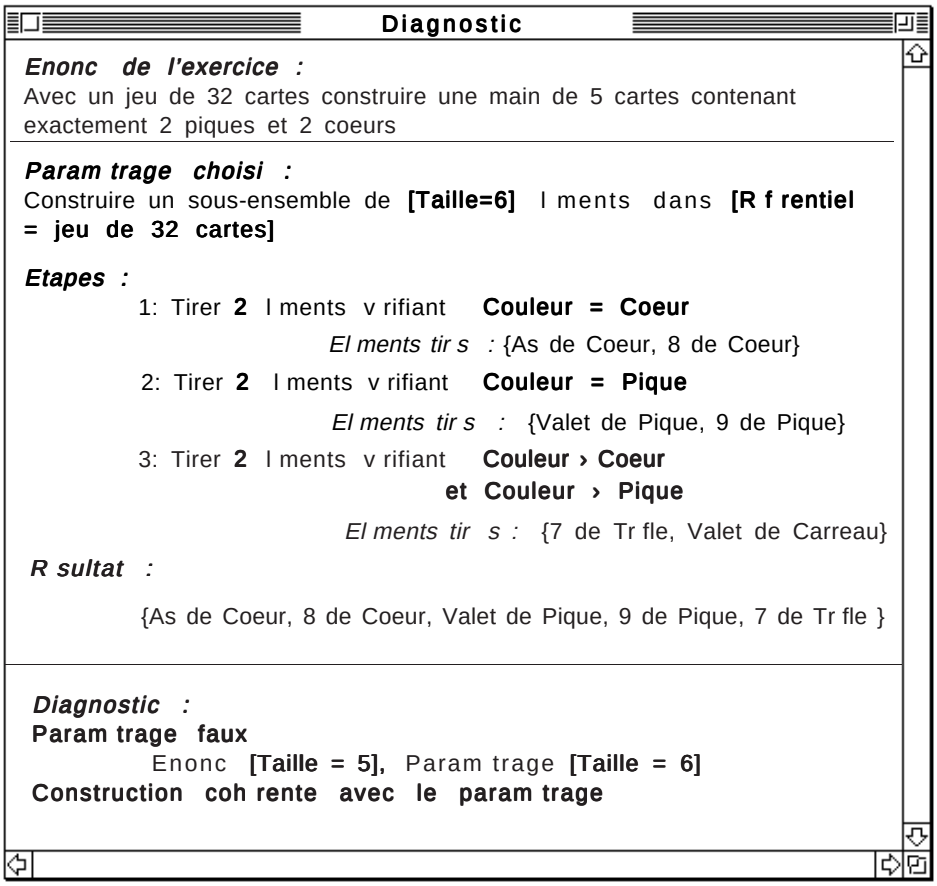


Figure 4

Dans ce cas, pour que l'élève énonce une réponse juste, il lui faudra recommencer à la deuxième étape. Dans d'autres cas, l'élève considère qu'il a terminé alors que sa construction est incomplète, le système détecte qu'il manque une partie de l'algorithme et le lui indique. L'élève peut alors compléter sa construction. C'est le diagnostic présenté dans la figure 5, l'élève n' a tiré que les deux coeurs et les deux piques et pense avoir terminé.

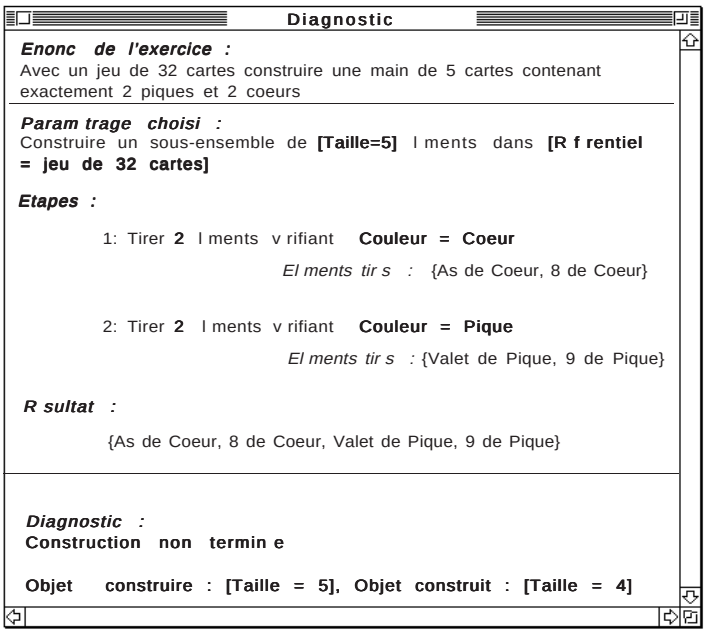


Figure 5

Dans le cas où l'algorithme est correct, l'élève se voit alors présenter une fenêtre (figure 6) reprenant son algorithme et lui permettant à chaque étape de faire compter par le système le nombre de tirages possibles, ce qui permet de résoudre le problème de dénombrement associé à l'exercice.

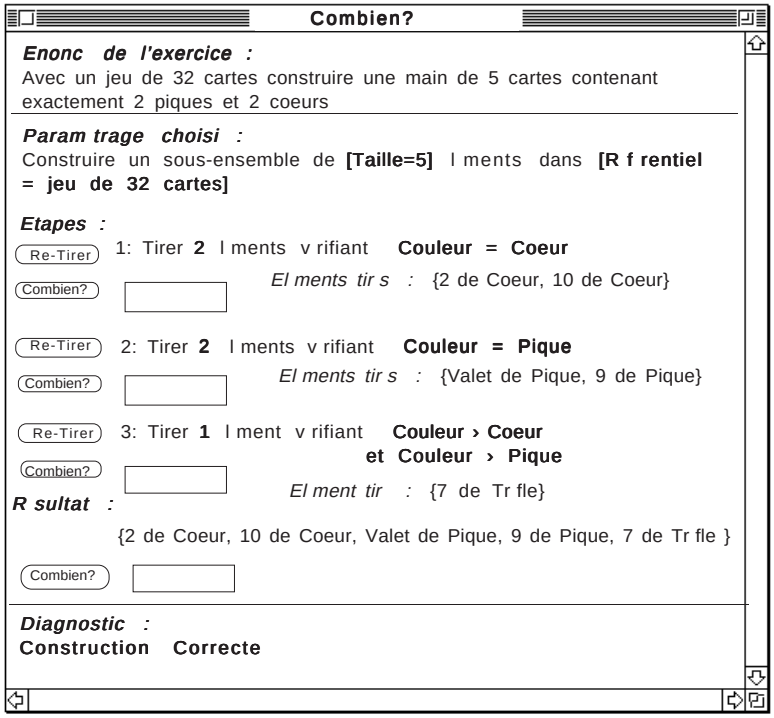


Figure 6

2.3 Objectifs pédagogiques

Notre objectif est de permettre à un élève d'une classe de terminale scientifique de se familiariser avec certains types de représentation et d'utiliser la méthode particulière de résolution que nous avons formalisée et que nous appelons "méthode constructive". L'élève devrait constater de lui-même que les représentations utilisées sont opérationnelles et les considérer dorénavant comme des outils potentiels. Avec cette démarche, nous souhaitons qu'il se fabrique des schémas mentaux associant des algorithmes de construction à des représentations de classes de problèmes [Julo 95].
Nous espérons qu'avec un entraînement sur des exemples variés, l'élève renforcera sa capacité à reconnaître des types d'exercices et qu'il se rendra compte qu'à chaque type d'exercice peut être associé un type d'algorithme, en liaison avec la structure et le comportement de la machine associée. Cela constituera une base solide pour résoudre des problèmes divers de dénombrement et préparera l'étude de domaines apparentés, comme l'algorithmique par exemple.

Nous présentons maintenant les problèmes qui se posent pour la conception, la spécification, et la réalisation des interfaces, en commençant par préciser ce qui fait la particularité des logiciels d'EIAO.

3. Spécificités d'une interface EIAO

Une interface destinée à un environnement d'apprentissage présente des particularités qu'on ne rencontre pas dans les interfaces classiques des logiciels outils généralistes, comme les traitements de texte. Nous comparons ici les deux et les qualités qu'on peut en attendre.

3.1. Interface d'un logiciel outil

Le but d'un logiciel outil (en bureautique par exemple) est d'aider l'utilisateur à effectuer une certaine tâche : éditer un texte, composer une image, stocker et rechercher des données. La qualité ergonomique des interfaces de ces logiciels peut être évaluée suivant plusieurs dimensions, dont voici quelques exemples [AFNOR 91] :
— *Compatibilité* (adéquation à la tâche et à l'utilisateur) : l'utilisateur doit pouvoir effectuer la tâche demandée avec une interface qui tient compte de ses connaissances, de son langage, de ses habitudes et de son environnement de travail.
— *Guidage* de l'utilisateur par retour d'information (feedback) et service d'aide : le logiciel doit manifester clairement son activité, ses changements d'état, les possibilités d'action de

l'utilisateur, les effets de ces actions. Il doit pouvoir répondre aux questions de l'utilisateur (aide).
— *Homogénéité* : les conventions de représentation et les comportements doivent suivre certaines règles de cohérence et d'uniformité. Les concepts similaires doivent être présentés de façon similaire et les tâches similaires doivent s'effectuer de façon similaire.
— *Souplesse* : l'initiative du dialogue doit être laissée le plus possible à l'utilisateur, le dialogue pouvant être partagé en plusieurs processus interruptibles. L'aspect de la présentation peut être configuré par l'utilisateur, et celui-ci doit pouvoir choisir entre différentes représentations d'un objet et entre différentes façons d'effectuer une tâche.
— *Contrôle* : même si le logiciel automatise la majeure partie du traitement et propose une solution, l'utilisateur doit pouvoir conserver le contrôle et accepter cette solution ou en proposer une autre (par exemple dans le cas d'un correcteur orthographique).
— *Concision* : la présentation et les procédures doivent être les plus simples et les plus courtes possibles tout en faisant figurer l'essentiel de façon claire.
— *Gestion des erreurs* : le logiciel doit détecter les erreurs le plus tôt possible (et même si possible empêcher leur apparition), les expliquer de manière significative et permettre à l'utilisateur de les récupérer.

En dehors de ces règles générales, de nombreuses recommandations ont été faites, portant sur des détails de présentation pour des systèmes d'exploitation particuliers (composition des fenêtres, des dessins, des textes, utilisation de la couleur) ou dans des domaines d'application particuliers. Voir par exemple [Apple 93] pour l'utilisation de la boîte à outils du Macintosh, [Microsoft 95] pour celle de Windows et [Kolski 93] pour le contrôle de procédés industriels.

3.2. Interface EIAO

Dans le cas d'un environnement d'apprentissage fondé sur la résolution de problèmes, la situation semble la même que dans un logiciel outil : l'utilisateur est l'élève, il a une tâche à remplir qui consiste à résoudre un problème et le logiciel fournit des outils pour remplir cette tâche. Mais la similitude est trompeuse : le logiciel n'est pas conçu comme un outil qu'on offre aux élèves pour leur permettre de résoudre des problèmes. Le but principal de l'activité n'est pas la *résolution*, mais l'*apprentissage*. La tâche de résolution de problèmes n'est qu'un moyen au service d'une méta-tâche : apprendre.
Dans un enseignement fondé sur des "situations-problèmes", la tâche proposée à l'élève comporte un obstacle dont seul le franchissement lui permettra d'acquérir la notion visée (on parle alors d'objectif-obstacle [Meirieu 93]). Une interface fondée sur cette stratégie devra donc présenter des obstacles et les rendre incontournables. Cela implique souvent de restreindre les possibilités de l'élève pour l'obliger à adopter une démarche particulière et à rencontrer effectivement l'obstacle sans pouvoir l'éviter. Mais cela entre en conflit avec les

recommandations ergonomiques d'un logiciel outil, qui demandent de laisser le choix à l'utilisateur de sa démarche et de surmonter autant que possible les obstacles à sa place (et surtout pas de lui imposer intentionnellement des obstacles). A la limite, un logiciel outil d'aide à la résolution de problèmes devrait résoudre le problème tout seul et fournir directement la réponse à l'utilisateur ! Ce n'est bien sûr pas ce qui est visé ici, même si effectivement le logiciel doit être capable de résoudre le problème dans un but ultérieur de diagnostic et d'explication.

Les recommandations ergonomiques générales restent cependant appropriées, mais elles doivent rester au service de l'objectif final : permettre à l'élève d'apprendre en affrontant des obstacles.

3.3. Types d'erreur en EIAO

De même, la notion d'erreur prend des significations variées dans un environnement d'apprentissage fondé sur la résolution de problèmes. On peut distinguer les erreurs relevant de l'utilisation des outils et celles qui correspondent à la résolution du problème posé.

Tentative d'utilisation illégale des outils

Une première forme d'erreur est celle qui apparaît lorsque l'élève cherche à utiliser les outils proposés d'une manière non conforme à leurs règles d'utilisation. Par exemple, demander à une calculatrice de diviser 1 par 0 ou, en dénombrement, demander à la machine de tirer quatre éléments distincts dans un ensemble qui n'en comporte que trois. La machine doit alors afficher immédiatement un message indiquant qu'elle ne peut pas effectuer cette opération et expliquer pourquoi.

Ce type d'erreur ne se limite pas aux opérations qui sont impossibles parce qu'elles ne sont pas définies mathématiquement, mais correspond aussi aux limitations imposées intentionnellement par la machine pour illustrer une notion. Par exemple, une machine à construire des ensembles par réunion de sous-ensembles tirés dans des "référentiels" impose à ces référentiels d'être disjoints deux à deux. Ce n'est pas une contrainte indispensable au but immédiat (construire un ensemble), mais à un but ultérieur (dénombrer les ensembles possibles en utilisant une méthode "constructive"). Pour construire une main de trois piques et deux coeurs, on interdira la construction suivante : "je tire d'abord un pique, puis deux piques, puis deux coeurs", bien qu'elle fournisse un résultat correct du point de vue de l'ensemble construit. L'élève est averti que chaque machine impose des contraintes et qu'il doit en tenir compte, même si dans un premier temps elles lui paraissent arbitraires (on peut d'ailleurs lui expliquer d'une manière informelle la raison de ces contraintes). Ces contraintes font justement partie des obstacles à surmonter dont on a parlé précédemment et doivent être prises par l'élève comme des règles du jeu.

Ce type d'erreur se retrouve également dans les logiciels outils, mais elles peuvent être traitées différemment : souvent le logiciel ne signale pas d'erreur mais effectue une action par défaut "inoffensive", sans commentaire. C'est souvent ce qu'on désigne sous le nom de "robustesse".

J'ai demandé à mon traitement de texte d'imprimer la page numéro 6 dans un texte qui n'en comportait que quatre : il a affiché le message habituel "impression en cours" et il ne s'est rien passé d'autre. Autrement dit, le logiciel fait semblant d'accepter la commande mais il ne l'effectue pas vraiment (il n'y a pas eu impression d'une page comme je l'avais demandé et comme il l'a prétendu). Ce type de réaction n'est pas approprié à un environnement d'apprentissage. Les erreurs sont en fait une partie intégrante du processus d'apprentissage et doivent être signalées et expliquées, sinon non seulement une confusion risque de s'installer mais de plus on prive l'élève d'une occasion d'apprendre quelque chose.

Mauvaise modélisation du problème

Dans notre environnement, l'élève doit non seulement résoudre un problème mais d'abord le modéliser en le reformulant en termes mathématiques. Le système possède son propre modèle du problème et peut le comparer à celui proposé par l'élève pour voir si les deux sont équivalents (il n'est pas nécessaire qu'ils soient identiques). La reconnaissance de l'équivalence de deux problèmes en général est difficile mais, dans le cadre d'une machine spécialisée qui ne permet d'utiliser que certains concepts organisés d'une manière prédéfinie, il devient facile de vérifier que deux problèmes sont équivalents.

Lorsque le modèle de l'élève n'est pas équivalent à celui du système, l' erreur doit être signalée et expliquée. Il s'agit ici d'une erreur portant sur la réponse obtenue à l'aide des outils et non plus sur l'utilisation des outils.

La question se pose de savoir si le système doit signaler l'erreur le plus tôt possible comme l'indiquent les recommandations ergonomiques de guidage et de gestion des erreurs. Un système possédant des capacités de raisonnement peut en effet détecter que le modèle que l'élève est en train de proposer ne peut pas être correct, avant même que sa description soit complète. Par exemple, si on demande de construire un ensemble à 5 éléments et que l'élève indique "taille = 6", on sait que le modèle est faux sans avoir besoin de connaître les autres données. Nous pensons ici que, dans un but pédagogique, il ne faut pas signaler l'erreur immédiatement mais attendre que l'élève complète le modèle et assume sa réponse. D'une part cela l'oblige à se sentir responsable de sa réponse au lieu de se décharger sur les possibilités de raisonnement du logiciel, et d'autre part cela lui laisse l'occasion de vérifier par lui-même sa réponse avant validation, de détecter tout seul des erreurs et de les corriger.

Utilisation légale mais réponse fausse au problème

De même, l'élève peut utiliser les outils correctement mais fournir une réponse qui n'est pas une solution du problème. Là encore, les mêmes principes pédagogiques s'appliquent : il vaut mieux attendre que l'élève ait fourni une réponse complète et qu'il l'ait assumée avant de signaler une erreur. Dans l'absolu, ce principe est à manier avec précaution, car il faut éviter que l'élève s'enferre dans des accumulations d'erreurs qui l'éloignent trop du sujet et qui rendent

impossible toute tentative d'explication cohérente. Ici, ce risque est minime car l'environnement impose des contraintes fortes sur la forme des solutions et sur le vocabulaire utilisé.
La question se pose cependant de savoir si le système doit attendre que l'élève ait effectué les deux tâches proposées (modéliser puis résoudre) avant de réagir. En effet, comment évaluer la résolution lorsque le modèle n'est pas correct ? Doit-on se référer à l'énoncé initial ou au modèle proposé par l'élève ? Nous avons choisi de privilégier la cohérence interne de la réponse de l'élève : le système évalue l'adéquation de la solution au problème tel que l'élève l'a modélisé (mais il signale bien sûr que le modèle est erroné si c'est le cas).

4. Spécification d'un logiciel interactif

Un environnement d'apprentissage est un type particulier de logiciel interactif, et à ce titre sa spécification et sa conception relèvent des méthodes générales de génie logiciel et d'ingénierie des interfaces utilisateur. Différentes méthodes et formalismes ont été proposés et nous n'allons pas les décrire ici. Nous indiquerons seulement les aspects que nous avons retenus ou introduits dans le cadre du projet Combien. Notons que les méthodes existantes (du moins celles que nous connaissons) ne prennent pas en compte tous les aspects qui nous intéressent, en particulier en ce qui concerne les particularités d'un environnement d'apprentissage.
Dans cette section, nous résumons les questions générales qui se posent ; nous développerons dans la suite la façon dont nous y répondons dans le projet Combien.

4.1. Modèle conceptuel

Une première chose à faire est de représenter le domaine d'étude (les dénombrements) sous forme de "modèle conceptuel". Ce terme, souvent utilisé, a des significations diverses suivant les auteurs. Il s'agit de décrire les concepts du domaine et les structures qui les représentent. Nous avons adopté une représentation par objets, utilisant les concepts classiques : classes, attributs, relations, héritage.
Il faut d'abord identifier les classes utilisées, définir les attributs de leurs instances et les relations entre les classes. L'ensemble des classes est structuré suivant une hiérarchie de généralisation / spécialisation (héritage conceptuel).
Les informations précédentes ne permettent pas de décrire entièrement quels sont les états autorisés des objets ni quels sont les assemblages d'objets autorisés. Il se peut qu'il soit nécessaire d'exprimer cela par des contraintes supplémentaires.

4.2. Transposition didactique

Le modèle conceptuel représente le domaine mathématique des dénombrements d'un point de vue théorique, expert. Or on ne cherche pas à présenter d'emblée à l'élève novice ce point de vue. Il faut effectuer une "transposition didactique" [Chevallard 91], qui consiste à ne montrer que certains aspects du modèle, éventuellement avec un vocabulaire différent et sous une forme qui n'est pas exactement celle du modèle. Cela revient en quelque sorte à passer d'un modèle conceptuel théorique à un modèle conceptuel didactique.
L'existence simultanée d'un modèle conceptuel théorique et d'une transposition didactique est une particularité des environnements d'apprentissage et n'est pas prise en compte par les méthodes usuelles. La proposition de [Bernat & Morinet-Lambert 95] de distinguer un "niveau conceptuel" et un "niveau de présentation" est une approche de ce problème.
Dans le projet Combien, toutes les machines partageront le même modèle conceptuel théorique mais chaque machine aura sa transposition didactique particulière, qui servira de filtre pour présenter certains aspects du modèle.

4.3. Interaction

Il faut ensuite définir les modalités d'interaction entre l'utilisateur et le système.
Le système et l'utilisateur peuvent tous les deux accéder aux données pour consulter des propriétés, créer des objets, en modifier ou en détruire. Mais chacun des deux a des responsabilités et des droits d'accès différents, de telle sorte que l'exécution de la tâche à accomplir est partagée entre eux. Il faut définir ces responsabilités et ces droits d'accès en tenant compte du fait qu'ils peuvent être modifiés dynamiquement lors de la session.
En ce qui concerne l'utilisateur, il faut préciser quels objets il est autorisé à voir et quelles actions il est autorisé à déclencher. Le système, lui, a accès à tous les objets et se charge de l'*exécution* des tâches déclenchées, mais il faut indiquer quelles sont ses responsabilités du point de vue du *déclenchement* des actions. Quelles sont les actions dont le déclenchement doit rester à l'initiative de l'utilisateur, et quand le système doit-il en déclencher d'autres ?
Les autorisations d'action ne peuvent pas être définies statiquement une fois pour toutes. On ne peut pas se contenter de déclarer une liste d'actions autorisées, ni même des séquences temporelles figées d'autorisations, car autoriser ou interdire une action est une décision qui dépend du contexte (en particulier de l'état des données et de l'historique des interactions). Il faut donc énoncer des connaissances permettant de déterminer dynamiquement, à tout moment, les autorisations et les interdictions.
La spécification des modalités d'interaction (qu'on appelle aussi la dynamique du dialogue) est un aspect de l'ingénierie des interfaces où la recherche n'a pas encore abouti à des solutions éprouvées. Différentes approches ont été tentées : grammaires, réseaux de transition (automates d'états finis, éventuellement récursifs et augmentés — Augmented Transition Networks),

diagrammes de flux de données (data flow diagrams), réseaux de Petri, règles de production, opérateurs de planification de type STRIPS (avec préconditions, ajouts, retraits), réseaux de contraintes.
L'étude des interactions est également un sujet fondamental pour les systèmes multi-agents (pour la représentation et le contrôle des actions, voir par exemple [Ferber 95]). L'une des tendances actuelles est d'ailleurs de gérer le contrôle de l'interface à l'aide de systèmes multi-agents.

4.4. Présentation

On désigne habituellement par le terme "présentation" tout ce qui concerne l'aspect graphique de l'interface (fenêtres, boutons, menus, champs de texte, dessins), couplé aux événements de base disponibles à l'utilisateur (clavier, souris). Il s'agit donc des entrées et des sorties perceptibles par l'utilisateur.
On trouve actuellement un certain nombre de bibliothèques et d'éditeurs d'objets de présentation (widgets) qui facilitent grandement la réalisation de la présentation d'une interface, en permettant de s'affranchir des détails du système de fenêtrage et de la gestion des événements de bas niveau. Mais ces outils n'offrent que des moyens assez rudimentaires de communication avec les objets et les fonctions du modèle conceptuel, et pratiquement aucun moyen pour spécifier à un haut niveau la dynamique du dialogue.
Le système AMULET [Myers et al. 97] propose des "interacteurs" qui sont des objets décrivant de manière abstraite des tâches de base génériques, comme sélectionner un élément parmi plusieurs ou déclencher directement une action. Cela permet de séparer nettement le comportement de l'interface des particularités de la boîte à outils utilisée, et donc de rendre réutilisable une plus grande partie du code.

4.5. Architecture

Plusieurs modèles d'architecture ont été proposés pour les systèmes interactifs. Ils partent le plus souvent d'un principe de séparation entre la partie qui gère les objets du modèle conceptuel du domaine (partie souvent appelée "application") et la partie qui s'occupe des interactions. Le principe de base est de n'introduire aucune connaissance concernant l'interaction dans la partie "application". Les différents modèles poussent plus ou moins loin cette décomposition en intercalant différents niveaux entre l'application et l'interface. Le modèle ARCH, par exemple, introduit les niveaux suivants (dans l'ordre, de l'application vers l'interface): application, adaptateur de domaine, contrôle du dialogue, présentation, boîte à outils d'interactions.
Cette décomposition en niveaux de responsabilité n'implique pas l'existence d'un module logiciel unique par niveau. La tendance actuelle est plutôt de répartir les tâches entre de nombreux agents, chaque agent assumant un certain niveau de responsabilité (dialogue,

présentation, etc.) vis-à-vis d'un élément particulier : il peut y avoir par exemple un agent responsable de la présentation d'un certain objet du domaine et un autre agent responsable du dialogue concernant ce même objet. Le fonctionnement de l'ensemble du système dépend alors de la structure des communications entre agents.

5. Modèle conceptuel : exemples de classes

Nous avons défini un modèle conceptuel pour les dénombrements (du moins pour les types d'exercices considérés). Voici quelques exemples de concepts intervenant dans ce modèle.

5.1. Problème de dénombrement

Un concept très général est celui de "problème de dénombrement". Un problème de dénombrement est de la forme suivante : "étant donné des ensembles (définis par des déclarations de *référentiels*), compter dans un certain *univers* les éléments vérifiant des *contraintes de sélection*. Exemple : avec un jeu de 32 cartes (référentiel), combien peut-on former de mains de 5 cartes (univers : ensemble de toutes les mains de 5 cartes possibles) comportant 2 piques et 3 coeurs (contraintes de sélection).
Un problème est représenté par une instance de la classe "Problème" et comporte entre autres les attributs : référentiels, univers, contraintes de sélection.

5.2. Référentiel

Un référentiel est un ensemble d'objets, désigné par un *nom de référentiel*. Chaque objet peut être représenté par un n-uplet de valeurs d'attributs. Par exemple, une carte à jouer est un couple (couleur, hauteur). Chaque attribut est désigné par un *nom d'attribut* et peut prendre ses valeurs dans un certain *domaine*.
Pour définir un référentiel, on le décrit comme réunion de *sous-référentiels* disjoints, ceux-ci étant eux-mêmes définis comme des produits cartésiens de domaines. Très souvent, il n'y a qu'un sous-réferentiel qui est le référentiel lui-même, mais cette façon de faire permet de décrire des situations plus complexes, comme la suivante : un ensemble de 9 jetons, dont 5 verts numérotés de 1 à 5 et 4 rouges numérotés de 1 à 4.
Certains référentiels sont prédéfinis, comme un jeu de 32 cartes ou l'alphabet usuel.
Précisons que la description d'un référentiel ne contient pas nécessairement l'énumération exhaustive de ses éléments. Il ne s'agit que d'une description de l'ensemble en compréhension, comme : le produit cartésien de l'intervalle [1, 1000] par l'intervalle [1, 2000]. Cette description

suffit pour pouvoir raisonner sur le référentiel sans qu'il soit besoin de représenter explicitement chacun de ses deux millions d'éléments.

5.3. Univers (ensemble de configurations)

Un univers est un ensemble de *configurations* formées à partir de *référentiels* donnés.
Exemples : l'ensemble des parties de taille 3 d'un référentiel R, l'ensemble des applications du référentiel A dans le référentiel B. La classe "Univers" admet deux sous-classes : "Univers des parties de taille donnée", "Univers d'associations". La première classe permet de représenter par exemple l'ensemble des mains de 5 cartes qu'on peut former avec un jeu de 32 cartes et la deuxième l'ensemble des mots de 5 lettres qu'on peut former avec l'alphabet usuel (un mot étant une application de l'ensemble des places dans l'alphabet).

5.4. Une comparaison

Les contraintes de sélection font intervenir des propriétés dites *attributives*, comme "couleur = pique". Une propriété attributive est une propriété pouvant s'appliquer à un élément d'un référentiel et concernant un attribut (la valeur de l'attribut "couleur" est "pique"). Cette dernière expression est représentée par un objet de classe "Comparaison" (sous-classe de "Propriété"). Une comparaison est une propriété attributive affirmant que la valeur d'un attribut donné est (ou n'est pas) égale à une valeur donnée.
Il faut distinguer une propriété de son application à un élément particulier : appliquer la propriété "couleur = pique" à la carte "7 de pique" renvoie Vrai, alors que cela renvoie Faux pour la dame de coeur. Une propriété joue donc le rôle de prédicat (fonction booléenne).

5.5. Une contrainte d'effectif

Un cas particulier de contrainte de sélection est une *contrainte d'effectif*, comme "Comporter 2 éléments tels que [couleur = rouge]". C'est une contrainte portant sur une configuration (c'est-à-dire un élément d'un univers) exprimant qu'un certain nombre de ses éléments doivent posséder certaines propriétés attributives. Autre exemple : comporter au moins 2 éléments dont l'attribut "numéro" vérifie le prédicat "pair".

5.6. Une construction

Une construction est la caractérisation d'une configuration comme réunion de sous-configurations disjointes. Par exemple, la partie "une main de 5 cartes contenant 2 piques et 2 coeurs" est la réunion de "une sous-partie contenant 2 piques", "une sous-partie contenant 2 coeurs" et "une sous-partie contenant une carte ni coeur ni pique".

La classe Construction comporte plusieurs sous-classes. L'exemple précédent correspond à la classe "Construction de partie". Une autre sous-classe est par exemple "Construction d'association" : une construction d'association est la caractérisation d'une association comme réunion de sous-associations disjointes. Par exemple, l'association "un mot de 5 lettres contenant deux a" est la réunion de "une sous-association avec 2 places contenant un a" et "une sous-association avec 3 places contenant une lettre différente de a".

Les machines à construire cherchent principalement à familiariser l'élève avec cette notion de *construction*, essentielle pour pouvoir dénombrer des ensembles de configurations. Chaque machine concerne un type de construction associé à un type de problème.

6. Modèle conceptuel : invariant d'une classe

Une description minimale d'une classe comporte l'indication de la superclasse et la liste des attributs et des relations vers d'autres classes (la distinction entre attribut et relation n'étant d'ailleurs pas toujours faite par les différentes méthodes). Mais ces informations sont en générales insuffisantes pour arriver à décrire entièrement la sémantique de la classe, c'est-à-dire pour définir en compréhension l'ensemble des instances potentielles légales de la classe. Nous donnons un exemple, puis nous indiquons les problèmes généraux qui se posent et les solutions envisagées.

6.1. Exemple

Voyons la structure nécessaire à la représentation du problème "construire une main de 5 cartes contenant 3 piques et 2 coeurs à l'aide d'un jeu de 32 cartes". On ne fait figurer que les objets et les attributs nécessaires à la présente discussion (les symboles courts comme P ou C1 représentent des objets, un symbole qui suit "est un" est un nom de classe, un symbole qui précède "=" est un nom d'attribut ou de relation) :

P est un Problème (Référentiels = {R}, Univers = U, Contraintes = {C1, C2})
R est un Référentiel (Attributs = {A1, A2})
A1 est un Attribut (nom = couleur)
U est un Univers-de-parties (Référentiel = R, taille-des-parties = 5)
C1 est une Contrainte-effectif (Propriétés = {P1}, taille = 3)
P1 est une Propriété-comparaison (Attribut = A1, relation = égalité, valeur = pique)
C2 est une Contrainte-effectif (Propriétés = {P2}, taille = 2)
P2 est une Propriété-comparaison (Attribut = A1, relation = égalité, valeur = coeur)

L'une des machines présente uniquement des problèmes de ce type. Mais comment définir exactement le type de problèmes visé ? On peut pour cela indiquer des contraintes portant sur les relations définies entre les classes. Dans ce qui suit, on les exprime à l'aide des concepts mathématiques de relation binaire, d'application, d'image d'une relation, de restriction d'une relation et de composition de relations :

— Référentiels est une application
Cela signifie : le problème possède un référentiel et un seul. Cela restreint la relation Référentiels définie au niveau le plus général pour la classe Problème : dans d'autres cas, un problème peut avoir plusieurs référentiels.
— Univers est à valeurs dans la classe Univers-de-parties
Cela restreint la relation Univers définie dans le cas général comme étant à valeurs dans la classe Univers.
— Univers.Référentiel ⊂ Référentiels
Cela signifie : le *référentiel* de l'*univers* du problème doit être l'un des *référentiels* du problème (et donc ici le seul référentiel du problème).
— Contraintes.Propriétés.Attribut ⊂ Référentiels.Attributs
Cela signifie : Si A est un attribut figurant dans une des *propriétés* d'une des *contraintes* du problème alors A doit être l'un des *attributs* d'un des *référentiels* du problème.
— La restriction de Propriétés à l'image de Contraintes est une application à valeurs dans la classe Comparaison.
Cela signifie : chacune des *contraintes* du problème possède une *propriété* et une seule, et cette propriété est de classe Comparaison.
— L'image de Contraintes.Propriétés.Relation est incluse dans {égalité}
Cela signifie : si R est la *relation* d'une des *propriétés* d'une des *contraintes* du problème alors R = égalité.

6.2. Spécification d'un invariant

L'ensemble des conditions exprimant qu'un objet est une instance légale d'une classe est souvent appelé l'*invariant* de cette classe. Différentes méthodes de conception permettent de prendre en compte cet invariant à des degrés divers (par exemple la méthode Classe-Relation [Desfray 96]). La plupart du temps, cela se traduit dans l'implémentation par un test programmé permettant de reconnaître si une instance respecte l'invariant de sa classe, test qui n'est utilisé que lors de la mise au point. Mais l'invariant d'une classe doit pouvoir être utilisé pour d'autres buts, en particulier pour maintenir automatiquement la cohérence d'une représentation en mémoire. Le système peut par exemple créer et relier des objets pour que l'invariant soit respecté lors de la création ou de la modification d'une instance.

Pour pouvoir utiliser une connaissance de différentes façons, le mieux est de l'exprimer de façon déclarative. Nous allons passer en revue le type de connaissances que nous avons besoin d'exprimer et les formalismes envisageables permettant de les exploiter. L'invariant d'une classe s'exprime par une conjonction de contraintes portant sur une instance et sur les liens qu'elle peut avoir avec d'autres objets.

• On veut pouvoir exprimer qu'un attribut de l'objet doit avoir une valeur constante donnée à l'avance, ou une valeur appartenant à un ensemble de valeurs donné à l'avance.

• On veut pouvoir exprimer la cardinalité d'un attribut ou d'une relation, c'est-à-dire le nombre de valeurs qu'il peut avoir (au minimum, exactement ou au maximum). La plupart des méthodes permettent d'exprimer une telle condition (mais celle-ci n'est pas forcément traduite entièrement dans l'implémentation).

• On veut pouvoir exprimer qu'un attribut est à valeurs dans une certaine classe, sous-classe de l'ensemble des valeurs autorisées dans le cas général (par exemple, imposer une Propriété-comparaison là où on attend simplement une Propriété dans le cas général). On dit souvent qu'on impose ainsi une contrainte de restriction de rôle. La plupart des méthodes permettent d'exprimer ce type de contrainte.

• On veut pouvoir exprimer par exemple que la somme des valeurs d'un attribut multivalué doit être égale à une valeur donnée (cela intervient en dénombrement).

• On veut pouvoir exprimer des contraintes de coréférence, comme "l'attribut A doit avoir une valeur égale à celle de l'attribut B", ou plus généralement une propriété faisant intervenir deux chemins partant de l'instance. Un chemin correspond à une relation composée, comme Contraintes.Propriétés.Attribut dans l'exemple précédent. Celui-ci illustre plusieurs contraintes de chemin. A notre connaissance, rien de particulier n'est prévu dans les méthodes classiques pour exprimer déclarativement ce genre de contrainte et surtout pour les maintenir automatiquement.

6.3. Formalismes envisageables

En dehors des méthodes de conception introduites en génie logiciel, différents formalismes offrent des possibilités pour exprimer déclarativement les contraintes précédentes.

Le langage mathématique

En interprétant une classe comme l'ensemble de ses instances potentielles et un attribut comme une relation binaire (éventuellement comme une fonction si l'attribut est monovalué), on peut utiliser les concepts mathématiques d'ensemble, de relation et de fonction pour exprimer les

contraintes précédentes. En les codant par exemple dans un langage logique du premier ordre, on obtient un formalisme déclaratif d'une puissance d'expression plus que suffisante.
S'il s'agit simplement d'énoncer des spécifications déclaratives, le langage mathématique général est un choix tout à fait satisfaisant. Mais nous souhaitons aussi que le système exploite ces spécifications pour différents usages. Réaliser un tel système n'est pas un mince travail, comme le montre différentes expériences (ALICE, MUSCADET, DESCARTES), et de plus ce n'est pas notre objectif principal. Notre but est simplement ici de définir un langage minimal et assez facilement exploitable, pour servir d'outil lors de la construction des "machines à construire".

Logiques terminologiques (KL-One)

Les logiques terminologiques, ou logiques de descriptions, permettent de définir des concepts (analogues à des classes), des rôles (analogues à des relations binaires) et des individus en exprimant déclarativement des contraintes à leur sujet. La plupart des types de contraintes que nous avons envisagés peuvent être exprimés dans ces langages, sauf les contraintes de chemin générales (le langage CLASSIC peut exprimer des contraintes de coréférence, mais uniquement pour des chemins de rôles fonctionnels, c'est-à-dire d'attributs monovalués).
Ces logiques ont été introduites comme supports de systèmes de classification de concepts (ordonner partiellement les concepts pour la relation de généralisation) et de reconnaissance d'individus (trouver les concepts les plus spécifiques que vérifie un individu). Mais comme il s'agit de formalismes déclaratifs, on peut envisager de les utiliser à d'autres fins. Il s'agit en fait d'une version réduite du langage mathématique des ensembles et des relations qui n'utilise pas de variables ni de quantificateurs.

Autres formalismes

Il existe d'autres formalismes apparentés aux précédents et servant de support à des systèmes à base de connaissances (souvent des systèmes d'analyse du langage naturel), comme les graphes conceptuels et les descriptions fonctionnelles. Nous avions nous-mêmes introduit une variante de ces formalismes pour analyser les énoncés d'exercices de dénombrement [Tisseau et al 94], mais nous avons provisoirement abandonné cette voie pour nous concentrer sur le résolveur de problèmes et sur son exploitation pédagogique.

Une fois le modèle conceptuel défini, il faut pouvoir spécifier une transposition didactique interactive : que va-t-on présenter à l'élève (quels concepts du modèle), suivant quelle structure (il faut éventuellement encapsuler plusieurs concepts du modèle initial en un nouveau concept didactique) et avec quels moyens d'actions ? Cela revient à définir l'"adaptateur de domaine" dans l'architecture ARCH de système interactif, c'est-à-dire le niveau chargé de la traduction entre les concepts du modèle conceptuel et la présentation interactive qui en est faite.

7. Variables, actions et contextes

L'étude de différents systèmes interactifs fait ressortir deux notions importantes sur lesquelles nous allons fonder notre réalisation : celle de *variable* (qui dans l'implémentation sera associée à un objet chargé des accès à une certaine donnée du modèle conceptuel) et celle d'*action* (qui sera associée à un objet chargé de contrôler l'autorisation et le déclenchement d'une procédure). De plus, nous introduirons la possibilité de regrouper des variables et des actions dans un *contexte d'interaction*.

7.1. Introduction d'un niveau d'indirection supplémentaire

L'idée est de réifier les concepts de base de la programmation orientée objet : suivant cette vision, tout objet a des attributs (ou variables d'instance, slots, champs, etc.) et répond à des messages en exécutant des méthodes (ou fonctions membres) définies dans sa classe ou une classe ascendante. Ce que nous voulons faire, c'est introduire des *conteneurs* pour ces éléments : des conteneurs d'objet, des conteneurs d'attribut (ce que nous appellerons des *variable*s) et des conteneurs de méthode (que nous appellerons des *actions*). Le conteneur d'objet sera une structure composée qui "possédera" des conteneurs d'attributs et des conteneurs de méthode. Le scénario d'utilisation consiste à affecter (ou "brancher") d'abord un objet au conteneur d'objet. Ensuite, en s'adressant à un conteneur d'attribut, on pourra consulter ou modifier la valeur d'un attribut de l'objet et, en s'adressant à un conteneur de méthode, on pourra exécuter une méthode portant sur l'objet. Si le mot *conteneur* donne une assez bonne image du rôle des objets qu'on veut introduire, il n'en couvre pas tous les aspects, et d'autres mots peuvent suggérer d'autres images : *adaptateur*, *proxy* (mandataire, délégué, suppléant, substitut) ou *médiateur* (pour utiliser le vocabulaire des *Design Patterns* [Gamma et al. 95]).

Dans la réalisation de l'interface, on s'imposera la discipline de n'accéder aux objets du modèle conceptuel que par l'intermédiaire des conteneurs. Cela permet de rendre l'implémentation du modèle conceptuel totalement indépendante de l'interface. De plus, la structure des conteneurs n'est pas astreinte à être isomorphe à celle des objets : d'une part certains attributs et certaines méthodes de l'objet initial peuvent ne pas avoir de conteneur associé, ce qui les protège et les rend invisibles à l'interface, et d'autre part un conteneur d'attribut n'est pas forcément en lien direct avec un champ (au sens de l'implémentation) de l'objet. Il peut s'agir d'un attribut virtuel (par exemple calculé, ou stocké dans le conteneur plutôt que dans l'objet) ou indirect (un champ d'un autre objet en lien avec l'objet de départ). De même un conteneur de méthode n'est pas forcément lié directement à une méthode définie dans la classe de l'objet. Le conteneur peut implémenter lui-même la procédure correspondante (on parle alors de "réparation sémantique" : un conteneur d'objet permet de présenter l'objet comme s'il avait plus de fonctions qu'il n'en a

en réalité). Autrement dit, l'introduction des conteneurs permet de modifier le point de vue qu'on peut avoir sur les objets du modèle conceptuel. C'est une façon de réaliser la transposition didactique : l'élève ne verra pas les objets tels qu'ils sont dans le modèle conceptuel, mais tels que les conteneurs les présentent.
Un autre intérêt d'introduire des conteneurs est de pouvoir leur associer des informations de niveau méta concernant l'avancement de l'interaction et de pouvoir ainsi contrôler celle-ci : une variable pourra être marquée comme valide ou invalide, une action pourra être marquée comme autorisée ou interdite, sous la responsabilité du système ou de l'utilisateur. On pourra de plus exprimer déclarativement des contraintes entre conteneurs : telle action ne peut être autorisée que si telle variable est valide. Pour maintenir ces contraintes satisfaites, on associera aux conteneurs des comportements réactifs : ils pourront réagir aux demandes de l'utilisateur et aux modifications internes pour maintenir la cohérence des données, mettre à jour les validités et les autorisations, et demander l'affichage ou le réaffichage de certaines données.

La présentation qui précède est une synthèse d'idées de diverses origines : la réification des attributs et des méthodes est un prolongement des "facettes" des systèmes à base de *frames*, et a été étudiée dans le cadre des systèmes réflexifs. Introduire un niveau d'indirection pour l'accès aux données résulte du principe d'encapsulation. Une première encapsulation est déjà réalisée dans les langages objets pour obliger les clients d'un objet à passer par un protocole (ou interface logicielle) pour s'adresser à lui. Ici nous ajoutons un second niveau d'encapsulation pour rendre les objets accessibles à l'interaction avec l'utilisateur. Une technique analogue est utilisée dans certaines implémentations des agents d'un système multi-agents. De nombreux environnements de développement d'interface proposent des mécanismes analogues à ceux des conteneurs. C'est le cas en particulier dans l'environnement que nous avons choisi (VisualWorks), qui utilise par exemple des "Value Holders" et des "Aspect Adaptors", ou dans Amulet, qui offre des "Command Objects".

7.2. Variables

Nous appellerons ici *variable* un objet servant d'intermédiaire (conteneur, adaptateur, etc.) pour accéder à un attribut d'un autre objet appelé son "sujet", le mot "attribut" étant pris dans un sens très large de "donnée liée", c'est-à-dire directement mémorisée dans un champ de l'objet ou obtenue par indirection ou par calcul à partir de l'objet. La variable connaît son "sujet" et sait quel message lui envoyer pour obtenir la valeur de l'attribut. Tout agent d'interface (client) qui veut obtenir la valeur de cet attribut n'est autorisé à le faire qu'en envoyant le message "valeur" à la variable (on peut remarquer que le client peut le faire même s'il ne connaît pas le sujet).
Comme une variable est destinée à être associée à une présentation graphique interactive, la structure des valeurs qu'elle peut prendre est importante, car elle conditionne les possibilités

d'interaction. De ce fait, il est pratique de distinguer les types de variable suivants : variable atomique (c'est-à-dire en fait dont la valeur est vue par l'utilisateur comme un objet indécomposable), variable collection de valeurs vues comme atomiques et variable composée (sa valeur est vue comme un objet qui a lui-même des attributs).

7.3. Actions

Nous appellerons ici *action* un objet servant d'intermédiaire (conteneur, adaptateur, etc.) pour accéder à une méthode procédurale d'un autre objet appelé son "sujet". La différence avec une variable vient de ce qu'ici la méthode est procédurale, c'est-à-dire qu'elle a pour effet de modifier l'environnement, alors que la méthode associée à une variable n'est que fonctionnelle (elle retourne simplement une valeur sans produire d'effets de bord). Pour déclencher la procédure associée à l'action, un agent client envoie le message "déclencher" à l'action, éventuellement avec des paramètres.
Une action peut être liée à une variable : affecter une valeur à une variable, ajouter une valeur à une variable collection, actualiser une variable (calculer sa nouvelle valeur et l'affecter), supprimer une valeur d'une variable collection. D'autres actions ont un effet global et peuvent affecter différentes variables.

7.4. Contexte d'interaction

Nous appellerons ici *contexte* (contexte d'interaction) un objet regroupant ("possédant", "connaissant") un ensemble de variables et d'actions ayant le même sujet. Par construction, ces variables et actions sont contraintes à avoir toujours le même sujet, même si celui-ci change au cours du temps. Ce sujet commun est aussi appelé le sujet du contexte. Le contexte joue le rôle de conteneur d'objet (il "contient" le sujet) et il possède des conteneurs d'attributs (les variables) et des conteneurs de méthodes (les actions).

8. Dynamique de l'interaction

Dans le type de logiciels interactifs que nous considérons, la tâche de l'utilisateur consiste essentiellement à éditer une structure (un ensemble d'objets informatiques) à travers une représentation à l'écran qui traduit la façon dont il voit les concepts associés. Cette représentation matérialise des contextes (par exemple sous forme de fenêtres), des variables (par exemple sous forme de textes affichés) et des actions (par exemple sous forme de boutons).
La représentation d'un contexte ne montre pas forcément toutes les variables du contexte et n'autorise pas forcément l'utilisateur à effectuer toutes les actions du contexte. Variables et

actions peuvent en effet être affectées d'attributs décrivant leur "visibilité" ou leur "accessibilité" et indiquant si elles doivent être affichées ou non, autorisées ou non.

8.1. Les attributs et le comportement d'une variable

Une variable est essentiellement un moyen d'accès à l'attribut d'un objet. Les informations importantes sont l'existence d'une valeur et sa validité.

A-t-elle une valeur ?

A un instant donné, une variable peut avoir une valeur ou non. Cette information est stockée dans la variable sous forme d'un attribut "affectée".

Est-elle utilisable par les actions ?

L'un des rôles d'une variable est de servir de paramètre d'entrée à des actions. Il est alors intéressant que celles-ci puissent savoir si la variable est dans un état qu'elles peuvent exploiter pour leur exécution. On dit dans ce cas que la variable est "valide". L'attribut "valide" rend compte de cette propriété.

8.2. Les attributs et le comportement d'une action

Ce sont les actions qui sont les supports principaux des informations concernant l'interaction. Elles peuvent également faire référence aux variables.

Quelles variables utilise-t-elle ?

Pour pouvoir s'exécuter, une action a besoin en général de consulter et d'utiliser les valeurs de certaines variables, qui jouent le rôle de paramètres d'entrée. Une action possède un attribut "entrées" qui contient l'ensemble de ces variables.

Quelles variables rend-elle utilisables (valides) ?

L'exécution d'une action produit des effets sur les objets du modèle conceptuel. En particulier, elle peut modifier la valeur de certaines variables. Une action possède un attribut "sorties" qui contient l'ensemble des variables modifiées par son exécution. Cet ensemble est connu a priori avant l'exécution de l'action et constitue donc une connaissance prédictive. Cela sous-entend qu'il faut uniquement introduire des actions dont les effets puissent être prévus à l'avance, ce qui limite quelque peu le type d'action envisageable. Dans les logiciels que nous réalisons, cette limitation n'est pas vraiment contraignante.

Qui en est responsable (système / utilisateur) ?

Le déclenchement d'une action est placé sous la responsabilité exclusive d'un des deux acteurs intervenant lors d'une session : l'utilisateur ou le système. Une action possède un attribut "responsable" pouvant prendre les valeurs "utilisateur" ou "système".

Son déclenchement est-il autorisé ?

Le déclenchement d'une action n'est pas nécessairement autorisé à tout instant, car l'action peut exiger certaines conditions pour s'exécuter de façon valide. Une action possède un attribut "autorisée" qui indique si elle est autorisée ou non.

A quelles conditions est-elle autorisée ?

Pour déterminer si une action est autorisée ou non, il faut entre autres tester des *préconditions*. Un attribut "précondition" contient une procédure effectuant ce test. Pour savoir si une action est autorisée ou non, le système utilise les attributs "entrées" et "précondition" : une action est autorisée si toutes ses entrées sont valides (on consulte l'attribut "valide" des variables) et si sa précondition est satisfaite.

A quelles conditions son déclenchement provoque-t-il une erreur ?

Les interfaces que nous réalisons ayant une vocation pédagogique, il n'est pas toujours souhaitable que le système protège l'utilisateur contre toutes les erreurs : il faut parfois laisser à celui-ci l'occasion de faire des erreurs pour qu'il puisse apprendre quelque chose. Les préconditions d'une action expriment seulement les conditions pour que l'utilisateur soit autorisé à déclencher l'action, mais ne garantissent pas que le déclenchement soit valide. Un attribut "erreur" contient une procédure qui teste (lors de la demande de déclenchement, mais avant l'exécution effective) si le déclenchement est invalide.

Comment doit-elle réagir lors d'une demande de déclenchement ?

Lorsque l'utilisateur demande le déclenchement de l'action et que toutes les conditions sont réunies pour que cette demande soit acceptée, elle est alors prise en compte. Cela a pour effet principal d'envoyer au "sujet" de l'action le message associé à l'action : c'est l'exécution proprement dite. Mais d'autres effets secondaires peuvent se produire avant et après cette exécution, et ces effets dépendent d'un attribut appelé "mode de déclenchement", dont nous indiquerons plus précisément l'influence par la suite. Le système exécute l'algorithme suivant :

Si on est dans un cas d'erreur, rendre un message d'erreur. fin

Sinon :

1. suivant le mode de déclenchement, commencer par rendre invalides certaines variables et interdire certaines actions.
2. exécuter la procédure associée à l'action.
3. valider les variables de sortie.
4. remettre à jour le contexte (autorisations, etc.).

8.3. Modes de déclenchement

Nous avons introduit trois modes de déclenchement correspondant à des types d'interaction courants : consommation, reprise, annulation.

Consommation

Typiquement, le mode “consommation” est affecté aux actions de validation qui créent des objets structurés à partir d’informations introduites auparavant par l’utilisateur : la création de l’objet consomme ces informations. Par exemple, la création d’une contrainte (“couleur = rouge”) à partir d’un attribut (couleur), d’une relation (=) et d’une valeur (rouge). D’une part, après validation, ces informations n’ont plus besoin d’être mémorisées à part car elles se retrouvent dans l’objet construit et d’autre part il n’y a pas de lien dynamique de mise à jour automatique entre les morceaux d’information et l’objet construit. Si on choisit un nouvel attribut après avoir validé une première contrainte, on ne modifie pas cette première contrainte, mais on prépare la création d’une deuxième contrainte. L’utilisateur ne doit pas croire qu’en modifiant une information, il va modifier l’état de l’objet construit précédemment.

De plus, il est fréquent que l’objet construit soit ajouté à un ensemble d’objets ou à une structure. Il ne faut pas qu’il soit ajouté plusieurs fois, d’où le mode “consommation” pour empêcher une duplication accidentelle de l’objet. L’idée générale est d’obliger l’utilisateur à fournir explicitement de nouvelles informations s’il veut créer un nouvel objet et lui faire comprendre qu’il ne pourra pas éditer l’ancien objet.

Dans l’algorithme de déclenchement, cela se traduit par le fait que les variables d’entrée de l’action sont invalidées (consommées) avant l’exécution de l’action.

Reprise

Typiquement, le mode “reprise” est affecté au choix d’une valeur par menu. L’idée est que, lorsqu’on choisit à nouveau une valeur, on veut remettre en cause le choix précédent (“je me suis trompé, j’efface et je recommence”) et donc il faut invalider les conséquences du choix précédent. La plupart des entrées d’information par l’utilisateur sont en mode “reprise”. Par exemple : “choix d’un exercice”, “choix d’un référentiel”.

Dans l’algorithme de déclenchement, cela se traduit de la manière suivante : on commence par invalider les variables de sortie de l’action (ce qui est une façon de défaire ses effets directs antérieurs). Cette invalidation peut se propager à d’autres actions de reprise car une action de reprise a un comportement réactif : dès qu’une de ses entrées change, elle invalide ses variables de sortie. Ce n’est qu’après cette phase de préparation qu’on déclenche l’exécution proprement dite de l’action.

Annulation

Typiquement, le mode “annulation” est affecté aux actions qui réinitialisent un contexte (par exemple, remettre à vide un ensemble).

Une action en mode annulation invalide toutes les *actions* du contexte, c’est-à-dire les interdit et invalide leurs variables de sortie (dans la phase de préparation). Elle n’invalide pas forcément toutes les *variables* du contexte, mais seulement les variables de sortie des actions. Il se peut qu’il y ait d’autres variables qui restent valides, ce qui permettra par exemple d’autoriser à nouveaux certaines actions lors de la phase de réactualisation.

8.4. Comportement des contextes d'interaction

Nous avons décrit comment un contexte réagissait aux demandes d'action de la part de l'utilisateur. Mais un contexte peut aussi déclencher lui-même des actions (celles qui sont sous sa responsabilité). Ces actions sont déclenchées de manière réflexe, dès qu'elles deviennent déclenchables (c'est-à-dire dès que les variables d'entrée sont valides, que la précondition est satisfaite et que l'on n'est pas dans un cas d'erreur). Pour réaliser un tel comportement, on peut doter le contexte d'un moteur fonctionnant cycliquement, chaque cycle correspondant à l'algorithme suivant, exécuté après chaque exécution d'une action utilisateur :

1. Tant qu'il existe des actions système déclenchables, en choisir une et la déclencher.
2. Actualiser les autorisations des actions utilisateur.
3. Attendre une demande de déclenchement de la part de l'utilisateur.
4. Déclencher l'action correspondante.

Pour pouvoir réaliser une interface de manière modulaire, nous avons introduit la possibilité d'imbriquer un contexte dans un autre, et de le coupler au contexte englobant. Ce couplage est défini par des liaisons entre variables, une liaison étant définie par un triplet : une variable du contexte englobant , une variable du contexte englobé et un sens de liaison (du contenant vers le contenu ou l'inverse). Une liaison est ainsi orientée et comporte une variable "source" et une variable "destination". Une fois que ces couplages sont mis en place, chaque changement effectué sur la variable source (affectation, validation) provoquera le changement analogue sur la variable destination.
On peut ainsi concevoir des contextes indépendamment les uns des autres et les réutiliser dans différentes interfaces, en les imbriquant dans d'autres contextes et en définissant des liaisons.
Ces contextes rappellent les contextes d'exécution mis en place par un interpréteur de langage de programmation dans la "pile" d'un système informatique. La différence est qu'ici, ces contextes sont créés à l'avance et qu'ils ne sont pas activés à l'initiative du système, mais à celle de l'utilisateur. De plus leur ordre d'activation n'est pas fixé : on peut activer un contexte contenu avant son contexte contenant, si les contraintes déterminées dynamiquement le permettent.

9. Conclusion

Dans cet article, nous avons décrit un logiciel d'EIAO que nous sommes en train de réaliser dans le cadre du projet COMBIEN?, puis nous avons présenté les difficultés rencontrées pour concevoir, spécifier et réaliser ce logiciel. Nous avons précisé notre approche sur ces différents points, en la comparant à d'autres travaux.

Nous nous sommes efforcés d'exprimer les spécifications sous une forme déclarative : modèle conceptuel défini par des classes et des contraintes, description du comportement de l'interface à l'aide de contextes, variables et actions. Nous cherchons ainsi à répondre à certains besoins :

— Nous travaillons en groupe en nous répartissant les tâches. Il faut alors pouvoir communiquer facilement nos idées et nos réalisations et les intégrer en un tout.

— Nous cherchons à réaliser un logiciel utilisable par différents types d'utilisateurs, sur des machines variées (Mac, PC, Unix). Il est alors nécessaire de disposer d'une spécification indépendante de l'implémentation.

— Notre travail est un travail de recherche nécessitant une approche par essais et erreurs. Les spécifications doivent alors être facilement modifiables.

— Nous voulons réaliser un assez grand nombre d'interfaces (une bonne vingtaine). Dans ces conditions, il est nécessaire de pouvoir réutiliser des composants.

— Les interfaces que nous voulons réaliser doivent être configurables dynamiquement, par exemple pour s'adapter au niveau de l'élève. Il est alors important que la plus grande partie du comportement soit décrite déclarativement et que cette description soit représentée explicitement (réifiée) dans l'implémentation, de manière à ce que le système puisse la modifier lui-même.

— Un logiciel d'EIAO a non seulement pour but d'offrir des outils à l'utilisateur, mais aussi d'observer celui-ci pour pouvoir l'aider plus efficacement. Il est alors important d'introduire des objets représentant l'interaction (les contextes, les variables, les actions), car ils pourront servir de support à l'observation du comportement de l'utilisateur.

La réalisation du logiciel est en cours et avance sur plusieurs points : nous avons défini un modèle conceptuel du domaine des dénombrements, nous avons spécifié plusieurs interfaces de "machines à construire", nous avons réalisé des prototypes de certaines de ces interfaces, ainsi qu'un prototype d'un moteur permettant de gérer les contextes, variables et actions. Nous entamons maintenant une réalisation plus "définitive" qui intègre ces différents aspects, après avoir choisi un nouvel environnement de programmation adapté au modèle présenté ici (VisualWorks, fondé sur le langage Smalltalk).

Nous pensons avoir maintenant les concepts et les outils nécessaires pour réaliser assez rapidement les interfaces du logiciel et les relier au modèle conceptuel. De plus, indépendamment du projet COMBIEN?, ces concepts et ces outils présentent en eux-mêmes un intérêt pour la conception et la programmation de logiciels interactifs, en particulier la gestion de la disponibilité des actions fondée sur notre modèle de contextes, d'actions et de variables.

Références bibliographiques

[AFNOR 91] Traitement de l'information, Évaluation des produits logiciels, Partie 1 : Définition des critères ergonomiques de conception et d'évaluation des interfaces utilisateurs, *norme AFNOR Z 67-133-1*, décembre 1991.

[Apple 93] Apple Computer, Inc, *Guide de l'interface utilisateur Macintosh*, Addison-Wesley France, Paris, 1993.

[Bernat & Morinet-Lambert 95] Bernat P., Morinet-Lambert J., Spécificités et modélisation de l'interaction dans un EIAO, dans: Guin D., Nicaud J.-F., Py D., (eds), *Environnements Interactifs d'Apprentissage avec Ordinateur, Tome 2,* Eyrolles, Paris, 1995, p. 209-220.

[Chevallard 91], *La transposition didactique. Du savoir savant au savoir enseigné*, Grenoble, La pensée sauvage, 1985 réed. 1991.

[Desfray 96] Desfray P., *Modélisation par objets, la fin de la programmation*, Masson, Paris, 1996.

[Dubourg 95] Dubourg X., Delozanne E., Grugeon B*., Situation d'interaction en EIAO : le système REPERES*, Environnements Interactifs d'Apprentissage avec Ordinateur, Tome 2. Paris, Eyrolles, 1995,p 233-244.

[Ferber 95] : Ferber J., Chapitre 4 : Action et comportement, dans: *Les systèmes multi-agents*, InterEditions, Paris, 1995, p. 157-237.

[Gamma et al. 95] Gamma E., Helm R., Johnson R., Vlissides J., *Design Patterns, Elements of Reusable Object-Oriented Software,* Addison-Wesley, 1995.

[Guin 95] Guin N., Giroire H., Tisseau G., Le classement de problèmes : Une méthode de résolution pour le module expert d'un EIAO. Application au problème de dénombrement. Guin D., Nicaud J.F., Py D., (eds) *Environnements Interactifs d'Apprentissage avec Ordinateur,* Tome 2. Paris, Eyrolles, 1995,p 113-124.

[Julo 95] Julo J., *Représentation des problèmes et réussite en Mathématiques*. Rennes, Presses Universitaires de Rennes, Collection Psychologie, 1995.

[Kolski 93] Kolski C., *Ingénierie des interfaces homme-machine, conception et évaluation*, Hermès, Paris, 1993.

[Le Calvez et al 97] Le Calvez F., Urtasun M., Tisseau G., Giroire H., Duma J. "Les machines à construire : des modèles d'interaction pour apprendre une méthode constructive de dénombrement" *EIAO'97,* actes des 5èmes Journées francophones EIAO, Cachan, M. Baron, P. Mendelsohn, J.F. Nicaud (eds), 14-16 mai 1997, Hermès, 1997, p.49-60.

[Meirieu 93] Meirieu P., Objectif-obstacle et situations d'apprentissage, dans Houssaye J. (ed), *la pédagogie : une encyclopédie pour aujourd'hui*, ESF Editeur, Paris, 1993, p. 289-300.

[Meyers et al. 97] Myers B.A.,McDaniel R.G.,Miller R.C.,Ferrency A.,Faulring A., Kyle B.D.,Mickish A., Klimovitski A.,Doane P. "The Amulet Environment: New Models for

Effective User Interface Software Development", *IEEE Transactions on Software Engineering*, Vol. 23, no. 6. June, 1997. pp. 347-365.

[Microsoft 95] Microsoft Corporation (Tandy Trower), *The Windows Interface Guidelines for Software Design*. Redmond, WA: Microsoft Press, 1995. ISBN 1-55615-679-0.

[Tisseau et al. 94] Tisseau G., Duma J., Giroire H., Le Calvez F., Urtasun M., Un langage de descriptions fonctionnelles, relationnelles et ensemblistes pour des problèmes de dénombrement. *Actes du 9ème congrès Reconnaissance des Formes et Intelligence Artificielle (Paris 11-14 janvier 1994), volume 2*, Paris, AFCET 1994 (ISBN 2-903677129-X), p. 739-744.

[Tisseau et al. 96] Tisseau G., Giroire H., Le Calvez F., Urtasun M., Duma J., *Une méthode "constructive" de résolution de problèmes de dénombrement et sa mise en oeuvre*. Rapport interne Laforia 96/11, mai 1996.

[Van-Eylen & Hiraclides 96] Van-Heylen H., Hiraclides G., *Graal : en quête d'une démarche de développement d'interfaces utiisateur*, Angkor Editions, Paris, 1996.

Colloque Intelligence Artificielle
BERDER, 16-18 septembre 1997

Connaissances Temporelles et Raisonnement

Michel Masson

LIP6
Université Pierre et Marie CURIE
masson@poleia.lip6.fr

Résumé: Nous abordons quelques problèmes liés à l'expression du temps dans une expertise en privilégiant les activités de type diagnostic liées au domaine médical. Nous passons en revue les modélisations basées sur la logique ainsi que celles basées sur une algèbre d'intervalles avec leur contraintes respectives. Deux réalisations spécifiquement adaptées au temps sont ensuite décrites.
Mots-clés: raisonnement temporel, représentation du temps.

1. Introduction

La prise en compte du temps dans les systèmes à base de connaissances orientés vers le diagnostic nécessite l'ajout de connaissances spécifiques au temps. Parmi les connaissances transmises par l'expert d'un domaine, la plupart utilisent explicitement ou implicitement le temps:

ex_1: une brusque variation de s_1 implique e_1
ex_2: la conjonction de s_2 s_3 et s_4 évoque e_2

Dans le premier cas, il faut observer le signe s_1 sur un intervalle dépendant du contexte (une journée, une semaine, ...). Outre la présence des trois signes dans le deuxième cas, il faut s'assurer qu'ils sont simultanés ou très proches dans le temps (la plage temporelle dépendant encore du contexte). Il est essentiel de pouvoir manipuler des connaissances temporelles dans un raisonnement; le temps est une grandeur complexe dont la prise en compte pénalise la recherche de solutions: "A common disclaimer by an AI author is that he has neglected temporal considerations to avoid complication" [McDermott 82]. Nos connaissances sur le temps sont sujettes à fluctuations, une même connaissance (ex_1) pourra être interprétée différemment en fonction de son utilisateur.

Le paragraphe suivant décrit quelques problèmes spécifiques à l'utilisation du temps dans une situation de diagnostic. Le paragraphe 3 analyse différentes catégories de solutions adaptées à la prise en compte du temps.

2. Quelques difficultés à prendre en compte.

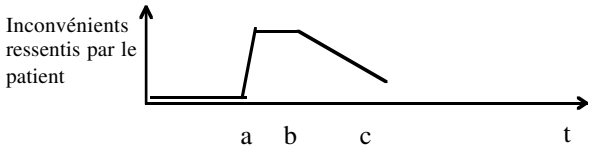
Dans ce paragraphe, nous désignerons par **fait temporel** un ensemble de propriétés vérifiées pendant un intervalle de temps donné et par **connaissance temporelle** toute expertise utilisant un ou plusieurs faits temporels ou exprimant une modification de ces faits.
Nous avons délibérément choisi des situations rencontrées lors d'un diagnostic. L'interaction "système - utilisateur" est importante, les questions posées par le système peuvent se référer à des faits temporels avec des différences importantes entre ces faits et leur perception. De même, les connaissances temporelles fournies par un expert contiennent une grande part d'informations implicites.

2.1 Evolution de la perception du temps

Un système de diagnostic pose (éventuellement) des questions pour améliorer ses conclusions; lorsque ces questions concernent un fait temporel, sa perception peut varier d'un utilisateur à un autre ou au fur et à mesure du temps pour un utilisateur donné. Ainsi le cas de la fracture d'un membre peut se décomposer en 3 temps:

(a) la fracture (toujours brève)
(b) le traitement
(c) la consolidation (au cours de laquelle la patient va récupérer plus ou moins complètement son état antérieur)

Les différents schémas de la Figure 1 traduisent l'évolution dans le temps de la perception de ce fait temporel par le patient. Le souvenir du fait peut aller de l'oubli à la perception permanente en fonction de son évolution. Dans le premier cas, un tel fait oublié par le patient devient inaccessible au système de diagnostic bien que sa réminiscence puisse expliquer d'autres troubles (il est courant dans le cas d'une fracture d'observer ultérieurement des douleurs d'origines vertébrales résultant de positions antalgiques prises lors de la fracture initiale). L'évolution de la perception d'un fait temporel peut également être analysée en termes de causalité (cf. § 2.4): il faut alors tenir compte du délai pouvant séparer l'effet de la cause.



En fonction de la récupération obtenue et du recul par rapport à l'événement, deux profils de perception sont possibles:

Dans le cas d'une récupération totale, la perception du trouble initial s'estompera jusqu'à devenir un incident ponctuel éventuellement susceptible d'être oublié .

Lors d'une récupération incomplète, il y aura persistance du souvenir de l'incident en fonction de la perception des séquelles.

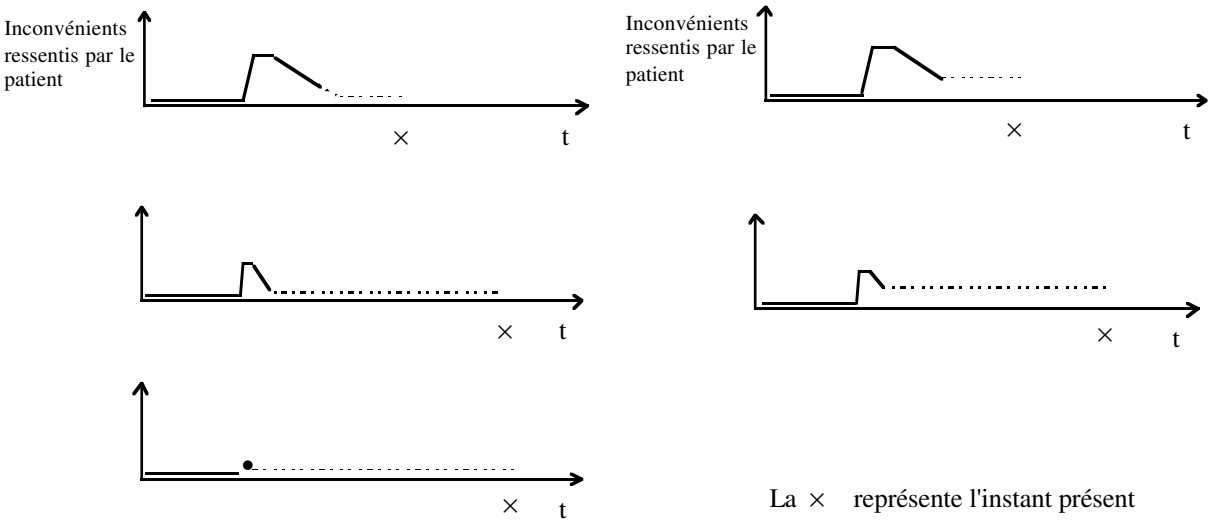
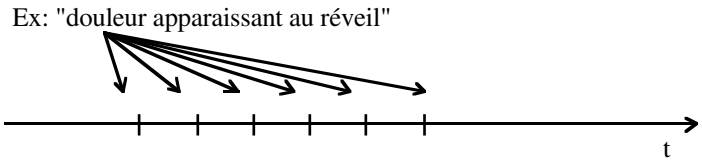


Fig. 1

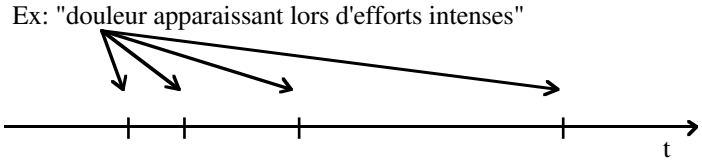
2.2 Les phénomènes périodiques

La périodicité d'un signe (i.e. sa corrélation avec un fait temporel) est fréquemment rencontré lors d'un diagnostic. Plusieurs cas sont possibles:

Référence à une période fixe:



Référence à une période fluctuante:



Dans ce cas la douleur est corrélée à une situation qui peut être modulée par le patient. Un comportement possible est d'espacer les situations liées à un effort, voire de les supprimer. La mise en évidence de la périodicité implique alors une analyse du phénomène accompagnateur.

Masquage d'événements:
Un événement à période fixe peut être ponctuellement perturbé (douleur aiguë, événement imprévu): il y aura alors occultation de l'événement périodique. La mise en évidence de la périodicité impose la mise en évidence de ces perturbations.

2.3 Problèmes liés à la granularité

Les faits temporels observés doivent pouvoir être décrits dans leur durée; plusieurs unités de temps peuvent être utilisées dans une même application. L'évolution d'une maladie infectieuse peut être mesurée en jours, semaines, mois. Par contre, l'évolution d'une maladie chronique se décrit en mois et/ou années.

2.4 Causalité

La causalité ne peut s'exprimer qu'entre des faits temporels; il importe donc de traduire le rôle du temps entre ces faits. La relation "A est la conséquence de B" peut être décrite par les assertions suivantes [McDermot 81]:

- si B est observé, alors A est toujours observé
- l'observation de A intervient durant un intervalle de temps (éventuellement nul) à partir de l'observation de B
- l'avènement de A est conditionné par l'existence d'une condition à vérifier pendant cet intervalle

Ainsi le problème lié à la perception d'un fait temporel évoqué au paragraphe 2.1 peut être exprimé en termes de causalité. En effet, une fracture peut être l'origine de désordres vertébraux résultant des positions antalgiques adoptées pendant la période de consolidation. Dans ce cas, la longueur du délai existant entre la cause et l'effet explique sa possible omission. Il est à

remarquer que la relation de causalité est interprétée différemment selon son utilisation: dans une activité liée à la compréhension du langage naturel, la causalité entre des faits temporels peut avoir une nature épistémologique. Elle sera alors traduite par la locution *depuis que* plutôt que par *parce que*. Plusieurs travaux [Nazarenko-Perin 92], [Mokhtari & Kayser 96] proposent un modèle basé sur l'existence du libre choix d'un agent extérieur: A est la cause de B, si B est l'action la plus plausible choisie par cet agent parmi un ensemble d'actions possibles (assimilables aux chroniques de McDermott - cf. § 3.1).

3. Analyse des approches existantes

L'introduction du temps dans un ensemble de connaissances augmente la quantité d'informations à traiter, le choix de ces informations et leur représentation conditionnent leur utilisation. Le temps peut être considéré comme une grandeur continue ou discrète, sa prise en compte peut se limiter aux événements passés ou présents ou bien servir à établir des projections dans le futur.
Un grand nombre de travaux utilisent soit une approche logique, soit une algèbre d'intervalles, soit encore une combinaison des deux. Quelques solutions ont été développées pour répondre spécifiquement aux problèmes exposés au paragraphe précédent.

3.1 Les modélisations obtenues par extension de la logique

McDermott définit une logique [McDermott 82] où le temps est une succession continue d'états partiellement ordonnés par la relation *précède* notée $<$. Une quarantaine d'axiomes permettent de modéliser le temps. Toutes les expressions sont en notation "polonaise parenthésée", ?x désigne une variable libre.
Axiome de densité: $(\forall\ s_1\ s_2\ (\ si\ (< s_1\ s_2)\ (\ \exists\ s\ (\ < s_1\ s\ s_2\ )\ )\ )\ )$
(les expressions commençant par s désignent un état)
A chaque état peut être associée une date (l'ensemble des dates est totalement ordonné). Un axiome permet de définir une chronique (une chronique est désignée par un symbole commençant par 'ch'): c'est un ensemble d'états totalement ordonné. Lorsqu'elle s'étend au delà du temps présent, une chronique décrit une éventualité d'événements; le non-déterminisme du temps étant traduit par l'aspect "branchu" de la représentation.

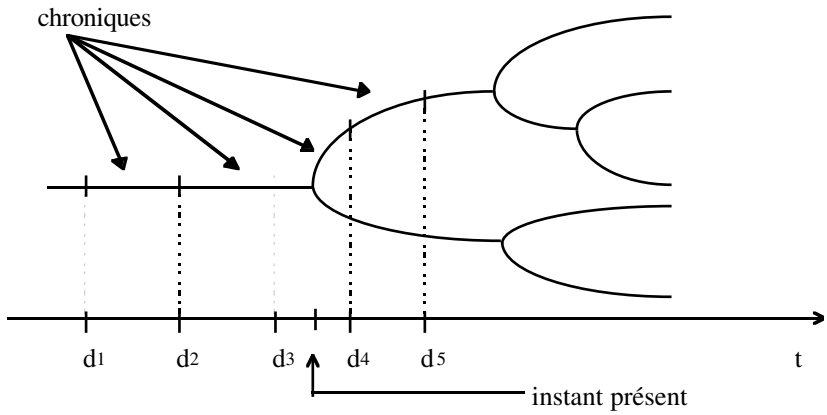


Fig. 2

Un fait temporel est défini comme une propriété (toute propriété commence par la lettre p) vérifiée entre deux états s_1 et s_2: (Occ [s_1 s_2] p). En combinant ces définitions, on peut construire des concepts plus complexes. Ainsi Jour et Nuit peuvent être définis comme deux événements se succédant alternativement de la façon suivante:

JOUR et NUIT sont deux événements mutuellement exclusifs, sauf aux extrémités, sera traduit par l'assertion:

(if (and (Occ $?s_1$ $?s_2$ JOUR) (Occ $?s_3$ $?s_4$ NUIT))

(∀ s (if (and (≤ $?s_1$ s $?s_2$) (≤ $?s_3$ s $?s_4$)) (or (= s $?s_2$ $?s_3$) (= s $?s_1$ $?s_4$)))))

JOUR ET NUIT alternent:

(and (follows JOUR NUIT) (follows NUIT JOUR))

Le prédicat *follows* traduit la succession des événements $?e_1$ $?e_2$ en imposant que toute chronique ch postérieure à $?e_1$ contienne l'événement $?e_2$.

(iff (follows $?e_1$ $?e_2$)

(if (Occ $?s_1$ $?s_2$ $?e_1$) (∀ ch (if (Occ ch $?s_2$) (∃ s_3 (and (Occ ch s_3) (Occ $?s_2$ $?s_3$ $?e_2$)))))))

Un tel système rend alors possible la déduction de l'énoncé: "tout événement est suivi soit de Jour, soit de Nuit".

Plus récemment, Schults et Kuipers ont proposé le système **CTL*** [Schults & Kuipers 97] mêlant logique temporelle et logique modale pour simuler des déroulements de processus physico-chimiques continus.

J.-F. Allen [Allen 84] propose une approche basée sur la notion d'intervalle: toute action occupe un laps de temps mesurable (même s'il est très court). Un petit nombre de prédicats définit des propriétés relatives à un intervalle de temps.

HOLDS (p , i)	p est vraie sur l'intervalle i
BEFORE (i_1 , i_2)	i_1 précède i_2
OCCURS (e , i)	l'événement e est vrai sur l'intervalle i

$\text{CAUSE}(e_1, t_1, e_2, t_2)$ l'événement e_1 à l'instant t_1 provoque e_2 à l'instant t_2

Outre les propriétés de transitivité, anti-symétrie et anti-réflexivité, les deux axiomes suivants décrivent les propriétés de la relation CAUSE (en particulier, l'effet ne peut être antérieur à la cause):

$$\text{OCCURS}(e_1, t_1) \wedge \text{CAUSE}(e_1, t_1, e_2, t_2) \rightarrow \text{OCCURS}(e_2, t_2)$$

$$\text{CAUSE}(e_1, t_1, e_2, t_2) \rightarrow \text{BEFORE}(t_1, t_2) \vee \text{MEETS}(t_1, t_2) \vee \text{OVERLAPS}(t_1, t_2) \vee \text{EQUAL}(t_1, t_2)$$

J.F. Allen montre qu'il est alors possible de modéliser la croyance ou l'intention concernant des actions ou des événements exprimés à l'aide du langage naturel.

Cette notion d'intervalle, développée ici dans un but linguistique, est reprise par plusieurs auteurs; ainsi dans [Terenziani 96] elle permet de manipuler des événements périodiques. L'auteur propose le système **TEMP+** effectuant des intersections et des compositions d'intervalles, il peut ainsi représenter des phrases décrivant des phénomènes périodiques simples (*chaque premier Lundi du mois, Jean vérifie les factures*) ou des phénomènes périodiques imbriqués, nécessitant plusieurs niveaux de granularité pour leur description (*trois fois par semaine, Jean vérifie chaque facture deux fois avant d'en faire une copie*).

3.2 La représentation du temps à l'aide d'une algèbre d'intervalles

Les travaux précédents utilisent soit une représentation continue, soit une représentation à base d'intervalles; ils ont en commun l'emprunt à la logique d'un formalisme de représentation et d'un système d'inférences. Or la notion d'intervalle est également le point de départ de plusieurs travaux basés sur la propagation de contraintes portant sur des intervalles temporels [Allen 83]. Si une représentation à base d'intervalles semble s'opposer à une représentation continue du temps, elle est néanmoins plus proche de la réalité: toute action, tout événement a une durée (même brève) et tout intervalle peut être subdivisé en sous-intervalles (problème posé par la granularité).

En premier lieu, J.F. Allen définit un ensemble de relations sur des intervalles de temps:

Relation	Symbole	Interprétation
X before Y	<	X X
X equal Y	=	X Y
X meets Y	m	X Y
X overlaps Y	o	X Y
X during Y	d	X Y
X starts Y	s	X Y
X finishes Y	f	X Y

Chacune de ces relations (hormis l'égalité) possèdent une relation inverse (suffixée par la lettre i). On obtient 13 relations exclusives que l'on peut composer grâce au tableau suivant:

	<	**>**	**d**	**di**	**o**	**oi**	**m**	**mi**	**s**	**si**	**f**	**fi**
<	<		< o m d s	<	<	< o m d s	<	< o m d s	<	<	< o m d s	<
>		>	> oi mi d f	>	> oi mi d f	>	> oi mi d f	>	> oi mi d f	>	>	>
d	<	>	d		< o m d f	> oi mi d f	<	>	d	> oi mi d f	d	< o m d s
di	< o m di fi	> oi di mi si	o oi d di =	di	o di fi	oi di si	o di fi	oi di si	di fi o	di	di si oi	di
o	<	> oi di mi si	o d s	< o m di fi	< o m	o oi d di =	<	oi di si	o	di fi o	d s o	< o m
oi	< o m di fi	>	oi d f	> oi mi di si	o oi d di =	> oi mi	o di fi	>	oi d f	oi > mi	oi	oi di si
m	<	> oi mi di si	o d s	<	<	o d s	<	f fi =	m	m	d s o	<
mi	< o m di fi	>	oi d f	>	oi d f	>	s si =	>	d f oi	>	mi	mi
s	<	>	d	< o m di fi	< o m	oi d f	<	mi	s	s si =	d	< m o
si	< o m di fi	>	oi d f	di	o di fi	oi	o di fi	mi	s si =	si	oi	di
f	<	>	d	> oi mi di si	o d s	> oi mi	m	>	d	> oi mi	f	f fi =
fi	<	> oi mi di si	o d s	di	o	oi di si	m	si oi di	o	di	f fi =	fi

Etant donné un ensemble d'intervalles reliés par des contraintes, il est alors possible de propager ces contraintes en les composant 2 par 2.

Ex. Soient u, v, w trois intervalles reliés par les contraintes: u {o , m} v et u {< , m} w

En utilisant les relations inverses de o et m, on peut écrire:

$$v \{ oi , mi \} u \{ < , m \} w$$

En prenant la réunion des composés de oi × < , oi × < m et mi × < , mi × m on obtient:

$$v \{ < , o , m , di , fi , s , si , = \} w$$

La mise à jour des contraintes, la gestion de cohérence est un problème NP-complet [Vilain & Kautz 86], plusieurs possibilités permettent de contourner cette difficulté. Ainsi la transformation d'une algèbre d'intervalles en algèbre d'instants munie des opérateurs { < , = , > } ramène la propagation à un temps polynomial, plusieurs auteurs ont proposé de réduire la vérification de la consistance entre 3 intervalles seulement, on obtient - pour un problème à n intervalles - une solution en $O(n^3)$. Sur des algèbres d'intervalles restreintes, des algorithmes en

$O(n^2)$ ont été proposés [Van Beek 92]. Une autre approche remplace la propagation de contraintes par le parcours d'un graphe temporel utilisant des algorithmes plus rapides pour trouver quelles relations existent entre deux intervalles [Ghallab & Mounir Alaoui 89].

3.3 Solutions basées sur la réification du temps

Les approches précédentes avaient en commun de replacer le temps dans un cadre plus général: la logique ou la propagation de contraintes. Les approches suivantes procèdent d'une démarche différente: au lieu d'étendre un cadre existant, leurs auteurs y ajoutent des éléments (sous forme réifiée) exprimant les connaissances temporelles du domaine. Les deux études suivantes traitent de sujets complexes (optimisation des opérations de chargement/déchargement d'un navire, contrôle de l'assistance respiratoire en milieu hospitalier) dont les modèles associés doivent prendre en compte une grande quantité de connaissances temporelles. En fonction du but à atteindre, la réification permet d'utiliser plusieurs types de raisonnement; cette démarche s'avère plus efficace qu'un principe général appliqué à l'ensemble des connaissances du domaine.

Advisor est un système de planification du chargement/déchargement de la cargaison d'un navire de type "vraquier". En effet, la nature de la cargaison, sa répartition dans les soutes du navire, la disponibilité des équipements portuaires, l'organisation des tâches élémentaires influent sur la durée de chargement/déchargement d'un navire de ce type. Le système **Advisor** utilise des connaissances sur:

- La structure du navire: le tirant d'eau indique la limite d'enfoncement de la coque, cette limite peut être dépassée en cours de chargement mais doit être respectée lors de la navigation. De même la valeur limite de déjaugeage (angle entre l'horizontale et l'axe longitudinal du navire) est soumis à une valeur limite pour la navigation. Le couple de cisaillement traduit une mauvaise répartition du fret dans les différents compartiments, il est également soumis à une valeur limite pour la navigation. Le remplissage des compartiments de la soute est tributaire de leur revêtement (peinture, inox).
- Les contraintes d'exploitation: plusieurs cuves de même contenu peuvent être regroupées en divisions. A chaque division doit être affectée une conduite de chargement/déchargement: ces installations dépendent de l'environnement portuaire et peuvent varier dans le temps. Le respect des valeurs limites concernant le tirant d'eau, le déjaugeage, le couple de cisaillement peut imposer de remplir temporairement des ballasts.
- Les contraintes stratégiques: il faut planifier chaque tâche élémentaire de façon à minimiser le temps total de chargement/déchargement.

Trois sous-systèmes permettent de manipuler ces connaissances: **SCS** (un résolveur de contraintes symboliques: ainsi l'ensemble de contraintes $Y = 8$, $X \leq Y + 4$ est transformé en $X \leq 12$), **TRLi** (un interpréteur de connaissances temporelles, en fait une extension de PROLOG) et **TRLp** (un système de planification temporelle indépendant du domaine d'application).

Les exemples suivants montrent comment peuvent être pris en compte les connaissances précédemment évoquées:

capacité-transfert (c1 , 1000) la conduite c1 a une capacité de transfert de 1000 l/h.

position (div3 , avant) la division3 se trouve à l'avant du navire

(10,11) : contient (div3 , huile , 500) la division3 contient 500 l. d'huile durant l'intervalle (T1,T2)

(T1,T2) : état-critique (couple-cisaillement):- le couple de cisaillement atteint une valeur critique pendant

(T1,T) valeur (couple-cisaillement), l'intervalle (T1, T2) si sa valeur courante dépasse la valeur

val-max(couple-cisaillemnt , C-max), maximale autorisée pendant une partie de cet intervalle.

couple-cisaillemnt > C-max , T < T2 .

Les connaissances stratégiques sont décrites à l'aide d'un concept d'action temporelle susceptible de décrire toute action en terme d'agencements de sous-actions: toute action correspond à un n-uplet de la forme:

(nom + paramètres , durée , liste_sous_actions , liste_conditions_initiales , liste_conditions_finales , contraintes)

Ainsi le déchargement d'un ensemble de cuves (division) correspond à la planification des tâches suivantes:

Nom (T1,T2) : déchargement (T1 , div , Fret , Q-Début , Q-Fin)
Durée [D1 , D2]
Ss-Actions (T1 , T3) : affecter_conduite (T1 , C , Div)
(T4 , T5) : ouvrir_valve (Div , Débit)
(T6 , T7) : déchargement (T1 , Div , Débit , Q-Début , Q-Fin)
(T8 , T9) : fermer_valve (Div , 0)
(T10 , T2) : libérer (T1 , Div)
Cond-init (Ta1 , Ta2) : contient (Div , Fret , Q-Début)
Cond-fin (Te1 , Te2) : contient (Div , Fret , Q-Fin)
Contraintes Ta1 < T1 , Ta2 > T1 , Ta2 ≤ T6 , Te1 ≥ T7 , Te1 ≤ T2 , Te2 > T2 , T3 < T4 ,
T5 ≤ T6 , T7 ≤ T8 , T9 < T10

Cette connaissance stratégique indique comment décharger une division d'une quantité donnée de fret (Q-Fin - Q-Début) à l'aide d'une conduite C sur un intervalle (T1 , T2). Chaque sous-action correspond soit à une action élémentaire soit à une action elle-même décomposable. **Advisor** construit une solution à partir des informations de départ; les connaissances stratégiques génèrent des contraintes temporelles. Le module **SCS** vérifie qu'elles satisfont aux contraintes d'intégrité existantes (ex. respect des conditions imposées au couple de cisaillement).

Lorsqu'une action génère des contraintes qui ne peuvent être satisfaites, le module **TRLp** cherche à remplacer cette action par une autre dont les contraintes associées sont transmises à leur tour au module **SCS**. Il y a arrêt lorsque la liste des buts non satisfaits est vide.
Advisor est un système adapté à la planification de tâches bien définies sans problème de granularité (l'unité de mesure des intervalles de temps est l'heure). Toutes les connaissances nécessaires à l'élaboration de la solution sont explicitées, ce qui facilite la mise au point du système.

Le système **NéoGanesh** développé par M. Dojat [Dojat 94] procède également par réification. NéoGanesh assure les fonctions de monitoring respiratoire en milieu hospitalier: son but est d'assurer au mieux le sevrage de l'assistance respiratoire d'un patient. Ce système est autonome et utilise des connaissances médicales de surface. Les contraintes à respecter sont essentiellement dynamiques: évacuer le gaz carbonique à la vitesse selon laquelle il est produit, éviter toute surpression pulmonaire, ménager la surcharge cardiaque, maintien des paramètres physiques, des cycles respiratoires, du volume échangé et de la pression partielle des gaz.
La particularité de **NéoGanesh** est de gérer les connaissances temporelles à l'aide de deux mécanismes d'abstraction: l'agrégation et l'oubli. L'expertise de ventilation utilisée (Hôpital H. Mondor) comporte 33 règles:

Règle 1: Ne jamais diminuer l'aide la nuit sauf en cas d'hyperventilation.

Règle 2: Diminuer le niveau d'aide si le patient a une ventilation correcte et stable pendant au moins 15 min.

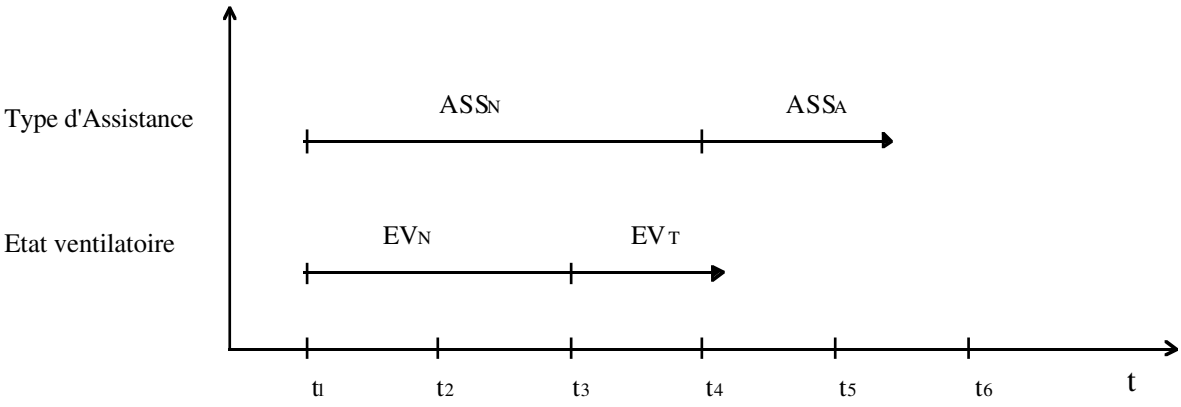
Règle 5: Envisager de sevrer le patient lorsque le niveau d'aide est minimal.

Règle 11: La fréquence respiratoire est de 12 à 18 cycles/min. pour un patient convenablement assisté.

Règle 20: Si le patient est tachypnéique, augmenter l'aide de 2 à 4 cm H_2O. Refaire une expertise au bout de 5 min.

La modélisation du temps repose sur la notion d'objet temporel défini à l'aide d'une logique à événement [Kowalski & Sergot 86]: un événement introduit une période pendant laquelle une propriété perdure. Ainsi un événement termine une période et en commence une autre; un objet temporel est déterminé par l'événement fixant son début et l'événement fixant sa fin. Un ensemble d'axiomes modélise les connaissances temporelles parmi lesquelles l'agrégation et l'oubli.
Examinons à partir de la Règle 20 comment fonctionnent ces abstractions. Une mesure de la fréquence respiratoire est effectuée toutes les 5 minutes (t_1 t_2 t_3 ...). Tant que cette mesure est dans son intervalle de normalité (ici $[t_1 , t_2]$ $[t_2 , t_3]$), l'état ventilatoire du patient est jugé normal (EV_N); il y a agrégation en un nouvel état ASS_N traduisant un état d'assistance normal. A l'instant t_4 la fréquence respiratoire est hors limite, cet événement provoque la création d'un nouvel état EV_T en interrompant l'état antérieur EV_N. A partir de t_4 , le mode d'assistance passe dans un nouvel état ASS_T. Si la modification portée à l'assistance respiratoire permet de retrouver un état ventilatoire normal, un nouvel état EV_N sera alors crée; l'incident correspondant à l'événement ASS_T pourra donc être oublié grâce à la création d'un état ASS_N

couvrant les périodes $[t_1 , t_2],...,[t_6 , t_7]$.



Ces mécanismes sont assurés par un ensemble d'axiomes portant sur des objets temporels; ainsi l'agrégation est exprimé par l'axiome A_4 (simplifié):

$$\forall\, ot_1\ \forall\, ot_2\ [\text{persistant}\ (ot_1) \wedge \text{équivalent}\ (ot_1 , ot_2) \wedge ot_1 < ot_2\ \wedge \text{durée}\ (ot_1 , d_1) \wedge \text{durée}\ (ot_2 , d_2) \rightarrow \text{valide}\ (ot_1 , d_1 + d_2)\ ]$$

Si un objet temporel ot_1 précède un objet ot_2 de même nature, alors les caractéristiques du premier objet demeurent valides sur la durée cumulée des deux objets.

Il est alors possible de représenter des connaissances temporelles correspondant à plusieurs types d'expertise; à cette distribution de compétences correspond une architecture de contrôle distribuée implémentée en NéOpus [Pachet 92].

4. Conclusion

Cette étude comparative montre l'intérêt de la prise en compte du temps dans l'expression d'une expertise de diagnostic. Un inconvénient apparaît avec certaines représentations; ainsi la modélisation utilisant des algèbres d'intervalles repose sur des techniques de propagation de contraintes dont le coût en complexité peut conduire à limiter le nombre d'objets utilisés dans la représentation du problème. Il nous paraît préférable de modifier une modélisation existante pour prendre en compte les éléments de connaissance se référant au temps. Ainsi le système **Advisor** utilise une extension de PROLOG pour traiter des faits et des règles se référant au temps. Le système **NéoGanesh** gère des objets temporels auxquels s'appliquent des connaissances spécifiques.

Bibliographie

[Allen 1983] Allen J.F. Maintening Knowledge about Temporal Intervals. Communications of the ACM 26 - 11, 832-843, 1983.

[Allen 1984] Allen J.F. Towards a General Theory of Action and Time. Artificial Intelligence 23, 123-154, 1984.

[Dojat 94] M. Dojat. Contribution à la Représentation d'Expertises Médicales Dynamiques: Application en Réanimation Médicale. Thèse d'Université, Université de Technologie de Compiègne, 1994.

[Ghallab & Mounir Alaoui 89] Ghallab M. Mounir Alaoui A. Managing efficiently temporal relations through indexed spanning trees. Proc. IJCAI, 1297-1303, 1989.

[Kowalski & Sergot 86] Kowalski R.A. Sergot M.J. A logic-based calculus of events. Nex Generation Computing, 4, 67-95, 1986.

[McDermott 82] McDermott D. A Temporal Logic for Reasoning About Processes ans Plans. Cognitive Science 6, 101-155, 1982.

[Marinagi & all. 96] Marinagi C.C. Panayiotopoulos T. Vouros G.A. Spyropoulos C.D. Advisor: A Knowledge-Based Planning System. International Journal of Expert Systems 9 - 3, 319-355, 1996.

[Mokhtari & Kayser 96] Mokthtari A. Kayser D. Normative and Epistemic Aspects in Causality. ECAI 96, 28-32, 1996.

[Nazarenko-Perin 92] Nazarenko-Perin A. Causal Ambiguity in Natural Language: conceptual representation of 'because' and 'since'. COLING, 880-884, 1992.

[Pachet 92] Pachet F. Représentation par objets et par règles: le système NéOpus. Thèse de Doctorat, Université Paris VI, 1992.

[Quek 96] Quek Ch. Wahab A. TR - Temporal Reasoner. Expert Systems with Applications 10, 449-455, 1996.

[Schults & Kuipers 97] Schults B. Kuipers B.J. Proving Properties of Continuous Systems: Qualitative Simulation and Temporal Logic. Artificial Intelligence 92, 91-129, 1997.

[Terenziani] Terenziani P. Towards an Ontology dealing with Periodic Events. ECAI 96, 43-47, 1996.

[Van Beek 92] Van Beek P. Reasoning about qualitative temporal information. Artificial Intelligence 58, 297-326, 1992.

[Vilain et Kautz 86] Vilain M. Kautz H. Constraint propagation algorithms for temporal reasoning. Proc. AAAI, 377-382, 1986.