

SAVANT 3 : un enseignement des concepts assisté par ordinateur

J.-L. Dessalles



Jean-Louis Dessalles, né le 21 octobre 1956.

Ancien élève de l'École Polytechnique et de Télécom Paris, il étudie actuellement, comme enseignant-chercheur à Télécom Paris, les applications des sciences cognitives à l'enseignement assisté par ordinateur.

1. W.L. Johnson,
E. Soloway
PROUST:
*An Automatic
Debugger
for Pascal Programs*,
in G. P. Kearsley :
*Artificial Intelligence
& Instruction -
Applications and
Methods*, pp. 49-67,
Addison-Wesley
Publishing
Company,
Menlo Park, USA,
1987.

2. B. F. Skinner
*La révolution
scientifique
de l'enseignement*,
Dessart, Bruxelles,
1969.

3. W. J. Clancey,
R. Letsinger
*Neomycin :
Reconfiguring
a Rule-Based
Expert System
for Application
to Teaching*,
STAN-CS-82-908,
Stanford University,
Stanford, 1982.

On ne saurait imaginer l'enseignement du siècle prochain sans ordinateur. Certains affirment même que quelques séances où l'étudiant interagit avec la machine remplaceront bon nombre d'heures passées à écouter le monologue du professeur, à déchiffrer des livres ou à peiner sur des exercices. Pourtant, malgré la dimension de l'enjeu, personne n'est en mesure, à cette date, d'indiquer la manière de doter l'ordinateur d'une compétence suffisante pour qu'il tienne son rôle dans un tel scénario. Les principes qui sont à la base du système SAVANT 3, développé à Télécom Paris, pourraient constituer un élément de cet enseignement assisté par ordinateur du futur ■

L'enseignement assisté par ordinateur (EAO), au cours de ses quelques décennies d'existence, a donné lieu à la réalisation d'excellents logiciels avec lesquels l'étudiant apprend réellement, parfois même avec plaisir. Malheureusement, les bons produits sont extrêmement rares et leur développement reste très coûteux, car les méthodes et les outils qui ont permis leur conception ne sont pas transposables. L'EAO est une denrée tellement rare que la plupart des étudiants ne le rencontrent jamais au cours de leurs études !

Un logiciel comme PROUST¹, par exemple, est conçu pour aider les étudiants à écrire leurs premiers programmes en langage Pascal. PROUST est capable, en analysant un programme, de reconnaître les intentions de l'apprenti programmeur, de détecter les éventuelles fautes de logique et de fournir à l'étudiant une correction personnalisée. Mais PROUST, prévu pour le langage Pascal, ne peut traiter que les programmes écrits pour résoudre des problèmes simples et connus de lui. La correction d'exercices moins élémentaires poserait des problèmes fort délicats et la transposition à un autre langage comme Lisp ou Prolog nécessiterait une refonte complète.

Ce problème de la transposition d'un sujet à un autre constitue le

véritable défi de l'EAO et, comme nous allons le voir, il n'est pas lié à des limitations quantitatives du matériel ou du logiciel. Il a pour origine la compétence dont on doit doter le système d'EAO et c'est cette compétence qui est pour l'instant spécifique de chaque sujet enseigné.

Les compétences d'un système d'EAO

La figure 1 montre l'architecture d'un système d'EAO. Même si aucun logiciel ne peut prétendre posséder toutes ces composantes à la fois, ce schéma se révèle fort utile pour comparer les différents systèmes entre eux. Il permettra aussi de comprendre ce qui rend le système très dépendant du sujet enseigné.

La compétence sur le sujet peut aller de la simple connaissance de bonnes et mauvaises réponses (cas de l'EAO classique, dit tutoriel, issu de l'enseignement programmé de Skinner²), jusqu'au système expert (comme le système NEOMYCIN³ qui utilise les règles du célèbre système expert MYCIN). Idéalement, un système d'EAO doit être capable de résoudre pour lui-même les problèmes qu'il pose à l'étudiant, afin de comprendre les réponses de ce dernier.

La compétence pédagogique conduit le dialogue étudiant/machine à partir d'un monitoring local. Dans un système tutoriel classique, c'est ici que se prend la décision d'aiguiller l'élève vers une révision. Mais le pilotage peut être beaucoup plus fin (par exemple, le système MENO-TUTOR⁴ adapte le style de dialogue en fonction de la qualité des dernières réponses de l'étudiant).

Le modèle de l'élève constitue pour beaucoup ce qui distingue un système d'EAO intelligent d'un autre et qui lui permet d'avoir un comportement radicalement différent avec chaque étudiant. A partir d'un monitoring global, qui peut même s'étendre sur plusieurs sessions, le système tient à jour les connaissances que l'étudiant semble maîtriser. Certains systèmes sont même capables de fabriquer des raisonnements faux pour pouvoir comprendre et modéliser la réponse de l'élève (par exemple IDEBUGGY décrit par Burton⁵). Le profil psychologique fixe un ensemble de paramètres censés représenter les compétences ou les préférences de l'étudiant. Ce modèle est utilisé pour changer le niveau ou la forme de l'interaction.

Enfin un système d'EAO doit communiquer avec l'étudiant et, lorsque cela se fait par l'intermédiaire du langage naturel, une compétence spécifique est nécessaire pour analyser la syntaxe et la sémantique des réponses de l'étudiant (comme le fait SOPHIE, de Brown et Burton⁶), ainsi que pour produire des répliques pertinentes. Parfois négligée dans certains systèmes, car reléguée dans l'interface, cette compétence de dialogue semble fondamentale. C'est sur elle que se sont portés tous les efforts dans le cadre du développement de SAVANT 3 et cette approche est justifiée dans ce qui suit.

On pourrait penser que, seules, l'expertise sur le sujet et la modélisation des connaissances de l'étudiant sont spécifiques du sujet enseigné et doivent être entièrement réécrites pour chaque cours et pour chaque chapitre. En fait, tous les composants d'un système d'EAO sont, à des degrés divers, dépendants du domaine. La solution consiste-t-elle alors à augmenter « la surface » de compétence de la machine, à la rendre omnisciente ? Cela reste envisageable pour un enseignement assez scolaire destiné à de jeunes enfants ; on peut par exemple espérer doter la machine d'une compétence en mathéma-

tiques équivalente à celle que l'enfant est censé acquérir en un an. Mais cela semble encore hors de portée si l'on parle d'enseignement supérieur (on peut constater l'importance de ces limites dans Modélis⁷, pourtant une belle tentative pour donner à un système une compétence générale en thermodynamique).

La solution adoptée dans SAVANT 3 procède de la démarche inverse ! Puisqu'on a de grandes difficultés à procurer une compétence étendue à l'ordinateur, on essaie de couvrir les sujets à enseigner avec de nombreuses expertises très locales, faciles à mettre au point et à gérer. Cette approche n'est pas contradictoire avec la précédente, dans la mesure où elle ne concerne que la compétence de dialogue. Mais les motivations de cette démarche ont des racines théoriques plus profondes qui en assurent le bien-fondé.

La nécessité de fondements théoriques

Comme cela a été mentionné plus haut, les échecs de l'EAO ne sont pas dus à de simples limitations logicielles ou matérielles. Il n'est pas non plus certain que le fait de doter l'ordinateur d'une grande compétence, quoique nécessaire pour résoudre le problème de la transposition, soit suffisant pour en faire un outil attrayant et quotidien de l'apprentissage. Le problème de l'EAO, dont l'enjeu est d'importance, a été longtemps sous-estimé. Il semble, à la lumière des difficultés rencontrées, que l'EAO ait beaucoup à espérer de ce qu'il est convenu d'appeler les sciences cognitives. Cela s'est vérifié dans le cas de SAVANT 3, car ce système tire parti d'une distinction essentielle sur le plan cognitif...

Les concepts et les savoir-faire

Les informaticiens et les psychologues ont coutume de distinguer deux représentations courantes des connaissances : la représentation déclarative, aisément exprimable et modifiable, et la représentation procédurale, constituée d'algorithmes. Ces deux modes de représentation sont réputés interchangeables en principe, mais, comme nous allons le voir et contrairement à une opinion souvent défendue, ils ne sont plus du tout équivalents lorsqu'il s'agit d'enseigner la connaissance

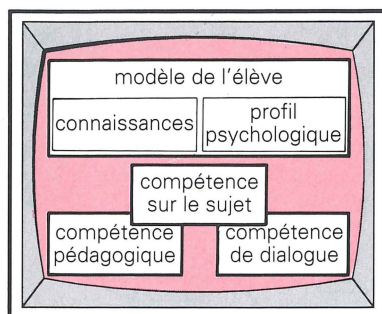
4. B. P. Woolf
Theoretical Frontiers in Building a Machine Tutor, in G. P. Kearsley: *Artificial Intelligence & Instruction - Applications and Methods*, pp. 229-268, Addison-Wesley Publishing Company, Menlo Park, USA, 1987.

5. D. Sleeman, J. S. Brown
Intelligent Tutoring Systems, pp. 157-183, Academic Press, Londres, 1982.

6. E. Wenger
Artificial Intelligence and Tutoring Systems, Morgan Kaufmann Publishers, Inc., Los Altos, Cal., USA, 1987.

7. G. Tisseau
Modélisation à partir d'un énoncé informel: le système Modélis. Application à des exercices de thermodynamique, Thèse de doctorat de l'université Paris 6, Paris, 1990.

**Fig. 1 Architecture
d'un système moderne d'EAO**



Un système moderne d'EAO doit être capable de s'adapter à l'élève. Il possède bien sûr une compétence sur le sujet enseigné pour apporter des connaissances et comprendre les réactions de l'élève, mais seules sa compétence pédagogique et la modélisation de l'élève lui permettent d'adapter à chaque instant sa façon d'enseigner.

correspondante. La transmission d'une connaissance conceptuelle, qui est par essence de type déclaratif, ne peut pas, comme nous allons le vérifier, être réduite à la transmission d'une procédure.

La notion de concept a été longtemps proscrite du vocabulaire de l'EAO, car elle n'entrait pas dans le cadre théorique qui a servi au début de fondement à cette discipline et qui fut défini par Skinner. La théorie « behavioriste », dont Skinner est l'un des principaux artisans, et la « pédagogie par objectifs » qui en est issue ne prennent en compte que les comportements observables de l'élève. Selon ces théories, l'acquisition d'un concept n'est pas un problème scientifique, dans la mesure où un concept n'est pas quelque chose d'observable, contrairement à la performance liée à l'acquisition d'un savoir-faire. Il ne faut donc pas chercher à enseigner des connaissances, mais des savoir-faire qui les mettent en œuvre.

Deux exemples

Il est important de comprendre que cette conception est erronée et que l'acquisition d'un concept est d'une nature qualitativement différente de celle d'une acquisition de savoir-faire. Prenons deux exemples : la notion de protéine et la conception de filtres analogiques. Le bon sens dicte que le premier est un concept

et le second un savoir-faire. En EAO classique, enseigner la notion de protéine n'a pas de sens. En revanche, on peut par exemple mettre l'élève en situation de discriminer parmi différentes formules chimiques celles qui décrivent des protéines, de même qu'on le mettra en situation de concevoir un filtre pour réaliser un traitement donné. Pourtant, une différence fondamentale sépare ces deux exemples d'apprentissage.

La remise en question du rôle de l'exercice

Cette différence concerne la performance de l'élève. Un élève qui utilise trop de composants pour réaliser son filtre peut apprendre comment les économiser. Il peut donc progresser. Cela a donc un sens d'évaluer sa performance au cours d'un exercice, comme on le fait systématiquement en EAO, et cela permet à l'élève de mettre au point

la procédure qu'il doit acquérir⁸. En revanche, on ne saurait affirmer que l'élève qui fait moins d'erreurs en discriminant des formules chimiques de protéines a progressé dans la compréhension de ce qu'est une protéine. Une protéine, c'est la mise bout à bout de différents acides aminés. L'élève qui se trompe identifie peut-être mal les acides aminés, ce qui relève d'un autre problème, ou bien il n'a rien compris à ce qu'est une protéine, mais en aucun cas il ne peut comprendre qu'« à moitié ». En résumé, mesurer quantitativement l'acquisition d'un concept n'a pas de sens et, pour cette raison, la démarche de l'EAO classique devient caduque dans le cas des concepts.

Tester des liens logiques

Dans l'analyse qui va être présentée ici, l'acquisition d'un concept par l'étudiant entraîne un processus de nature logique (voir encadré 1).

1. Représentation logique d'un concept

Les concepts humains (comme par exemple le concept de voiture) ont une triple définition.

- Tout d'abord une définition cognitive, liée à la fonction du concept (une voiture me permet d'aller rapidement où je veux), à des représentants typiques (ma vieille voiture), à différents contextes (autoroute, mécanique, embouteillage), etc.
- Ensuite, on associe aux concepts des fonctions de reconnaissance efficaces (je détecte une voiture au simple bruit d'un moteur, ou à la vue d'un pare-chocs dépassant d'un garage).
- Enfin, et c'est ce qui est important ici, on ne peut s'empêcher d'avoir présents à l'esprit, à chaque instant, des liens de nature logique qui relient les concepts « activés » à d'autres concepts. Ces liens dépendent fortement du contexte du moment. Ainsi, s'il s'agit d'un péage d'autoroute :

$\forall v ; (\text{non moto} (v) \text{ et non camion} (v) \text{ et non caravane} (v)) \Rightarrow \text{voiture} (v)$
 $\forall v ; \text{voiture} (v) \Rightarrow \text{non moto} (v)$
 $\forall v ; \text{voiture} (v) \Rightarrow \text{non camion} (v)$
 $\forall v ; \text{voiture} (v) \Rightarrow \text{non caravane} (v)$
...

La première relation signifie que tout ce qui n'est ni moto, ni camion, ni caravane est nécessairement une voiture. Du point de vue de l'employé d'un péage, cela peut suffire.

En revanche, si le contexte est celui d'un voyage, on est capable d'expliquer des relations du genre :

$\forall v ; (\text{est_venu_de_Lyon_en} (\text{Adèle}, v) \text{ et non train} (v) \text{ et non avion} (v))$
 $\Rightarrow \text{voiture} (v)$
 $\forall v ; \text{voiture} (v) \Rightarrow \text{non train} (v)$
...

Ces formules constituent le contexte logique que l'on trouve, par exemple, derrière toute conversation⁹. La définition logique d'un concept que l'on peut retirer de ces relations est toujours locale, provisoire et perfectible (un side-car est-il une voiture?). Mais elle est essentielle pour sa transmission et son utilisation dans le langage.

L'étudiant, confronté à des informations visant à lui présenter un nouveau concept, reconstruit ce concept pour lui-même. Cette reconstruction comporte un ensemble de liens logiques et ce sont ces liens qui sont accessibles à un test qualitatif. Autrement dit, si l'élève a

correctement compris le concept, les liens logiques qu'il a élaborés pour relier ce concept à d'autres concepts déjà connus seront les mêmes que ceux du professeur. La question qui se pose alors est de savoir comment un tel test peut être effectué. Peut-on proposer à

2. Représentation des connaissances à l'aide de clauses paradoxales

Dans un système expert, une partie de la connaissance est stockée sous la forme de règles qui ont la forme suivante :

$$(h_1 \text{ et } h_2 \text{ et } \dots \text{ et } h_n) \Rightarrow c$$

Ce genre de formule, appelée règle de production, distingue clairement les hypothèses de la conclusion. Cela est bien sûr fort utile dans le cadre des systèmes experts, par exemple pour un système qui chercherait à déduire un diagnostic de maladie (c) à partir de différents symptômes ($h_1, \dots, h_n$).

Pour ce qui est du traitement des conversations et de l'argumentation didactique, en revanche, on ne peut tolérer que les conclusions soient a priori distinguées. Par exemple, à partir de :

$$(\text{suspension_filtre_passe_bas et suspension_dure}) \Rightarrow \text{fréquence_de_coupure_haute}$$

on veut pouvoir déduire que la suspension n'est pas dure si l'on peut la considérer comme un filtre passe-bas et que la fréquence de coupure est basse. Un bon moyen de faciliter ce genre de déduction consiste à stocker la connaissance sous forme de clauses paradoxales¹⁰, ce qui a pour avantage de symétriser le rôle des prédicats entrant dans une règle. Ainsi, la clause ci-dessus peut se réécrire sous la forme paradoxale suivante :

$$(h_1 \text{ et } h_2 \text{ et } \dots \text{ et } h_n \text{ et non } c) \Rightarrow \text{FAUX}$$

Cette formule signifie qu'il serait paradoxal que $h_1, h_2, \dots, h_n$ et non c soient vraies simultanément. FAUX représente une formule toujours fausse. La formule $p \Rightarrow \text{FAUX}$ affirme ainsi la fausseté de p.

Le traitement des conversations exige cependant que l'on fasse intervenir une nouvelle modalité, la modalité INDÉSIRABLE. Toutefois, si l'on conserve le mode d'écriture des clauses paradoxales, le traitement de cette nouvelle modalité se fait sans difficulté :

$$(p_1 \text{ et } p_2 \text{ et } \dots \text{ et } p_n) \Rightarrow \text{INDÉSIRABLE}$$

On interprète cette formule en disant qu'il serait indésirable que $p_1, p_2, \dots, p_n$ soient vraies simultanément.

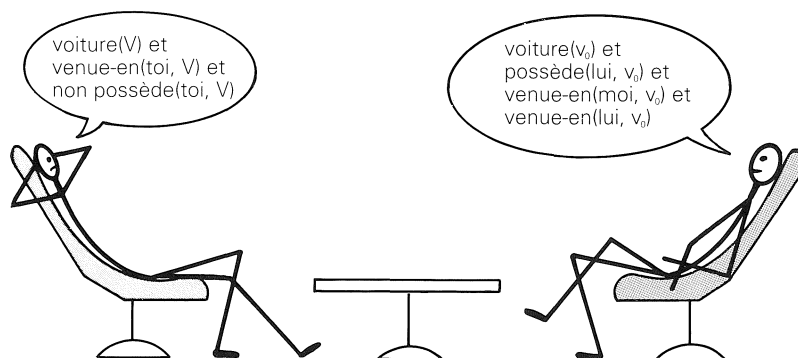
10. J.-L. Dessalles
*Computer Assisted
Concept Learning*,
in D.H. Norrie &
H.-W. Six : *Lecture
Notes in Computer
Science 438 -
Computer Assisted
Learning*,
pp. 175-183,
Springer-Verlag,
Berlin, 1990.

l'étudiant différentes formulations en langue naturelle des liens logiques que l'on veut tester, mêlées à des formulations fausses, en lui demandant d'identifier les formulations correctes ? Une telle méthode se révélera vite fastidieuse pour l'étudiant, sans parler des effets

néfastes de telles questions fermées qui suggèrent des réponses à la fois fausses et vraisemblables.

La solution retenue dans SAVANT 3 consiste pour la machine à engager une véritable conversation avec l'étudiant.

Fig. 2 Les conversations sont-elles logiques ?



Copier les conversations naturelles

Il semble utile, lorsque l'on cherche à transmettre des concepts avec une machine, de se demander comment s'opère la transmission naturelle des concepts. Or, la relation professeur-élève n'est pas une situation naturelle. La situation naturelle privilégiée pour l'échange des concepts, c'est celle où le langage est utilisé de la manière la plus libre : la conversation. L'étude attentive des conversations spontanées^{9,11} montre que celles-ci sont le siège de phénomènes logiques qui ont justement pour effet de comparer les liens logiques inter-conceptuels élaborés par les interlocuteurs ! Vérifions ceci en examinant la structure logique des conversations quotidiennes.

Le modèle de la conversation naturelle : partir d'un paradoxe

L'analyse précise de scripts de conversations montre que celles-ci se laissent tout naturellement classer en trois types selon le statut logique de la première intervention, celle qui introduit le nouveau sujet de conversation.

Dans les conversations du premier type, un interlocuteur commence par énoncer un paradoxe. En voici un exemple réel :

– « Tout à l'heure, j'ai vu un village

jumelé avec une ville d'Alsace ! » Cette première réplique, jointe à la connaissance selon laquelle deux villes jumelées appartiennent nécessairement à deux pays distincts, constitue un paradoxe, que l'on peut traduire logiquement par :

$\forall \text{ Ville_A, Ville_B ;}$
 $(\text{ jumelées(Ville_A, Ville_B)}$
 $\text{ et non étrangères(Ville_A, Ville_B) })$
 $\Rightarrow \text{ FAUX}$

Dans cette écriture, FAUX désigne une proposition toujours fausse. Affirmer $(p \Rightarrow \text{FAUX})$ revient à dire que p est fausse (voir encadré 2).

Le deuxième type de conversation est centré sur un événement a priori très improbable, énoncé dans la première réplique. Nous n'en parlerons pas ici.

Le troisième type résulte de l'énoncé d'un fait désirable ou indésirable, comme dans l'exemple réel suivant :

K1 – « Hé, ça doit pas être très bon pour lui [petit enfant], ça ? Non ? »
 L1 – « De la béchamel, si, c'est bon. »
 K2 – « Oui, mais il y a du curry dedans, ça fait rien ? »

que l'on peut rendre par :

$\forall E, A ; (\text{ mange (E, A)}$
 $\text{ et jeune_enfant(E)}$
 $\text{ et contient(A, curry) })$
 $\Rightarrow \text{ INDÉSIRABLE}$

Ici, INDÉSIRABLE est employé comme FAUX ci-dessus, mais dans le cadre d'une logique « augmentée ». Toutes les règles de réécriture de la logique du premier ordre restent valables. Simplement $(p \Rightarrow \text{INDÉSIRABLE})$ a pour signification que p est indésirable.

Il est remarquable que les interlocuteurs prennent toujours le soin de respecter cette règle d'introduction. Lorsque ce n'est pas le cas, par exemple en situation de malentendu ou d'expérience provoquée, l'auditeur manifeste une réaction d'incompréhension éventuellement agressive¹¹. La tentation est alors forte d'obliger la machine à débiter sa conversation didactique par l'énoncé d'un paradoxe ou d'un fait (in)désirable. L'interaction de l'élève et de la machine en devient peut-être « humainement » admissible. Mais l'intérêt d'une imitation des conversations naturelles ne s'arrête pas là.

La conversation : un réparateur logique

En commençant un sujet de conversation par l'émission d'un paradoxe, un interlocuteur signale une incohérence de sa « base de règles ». Dans le cas (fréquent) où le vis-à-vis ne connaît pas la même incohérence, ce dernier émet un argument qui casse le paradoxe (figure 2). Dans le premier extrait sur le jumelage, il pourrait mettre en doute l'un des termes du paradoxe, par exemple le caractère non étranger (« cette ville n'est pas en Alsace, mais en Allemagne »). Il pourrait aussi modifier le paradoxe pour qu'il ne s'applique plus (« les villes de moins de 500 habitants sont jumelables au sein du même pays ») :

$\forall \text{ Ville_A, Ville_B ;}$
 $(\text{ jumelées (Ville_A, Ville_B)}$
 $\text{ et non étrangères (Ville_A, Ville_B)}$
 $\text{ et non village (Ville_A) })$
 $\Rightarrow \text{ FAUX}$

Le principe des conversations sur le mode (in)désirable est sensiblement le même. En effet, il est facile de le décrire à l'aide d'une logique appropriée⁹. Cette analyse logique des conversations rend compte de leur principale fonction, au point qu'elle suffit pour permettre à un programme comme PARADISE¹¹ de réaliser une reconstitution acceptable de la conversation à partir de la seule connaissance logique des interlocuteurs !

11. J.-L. Dessalles
The simulation of conversations,
 Actes du colloque
 Cognitiva 90,
 pp. 303-310,
 AFCET, Madrid,
 1990.

Extrait de dialogue 1

M – Si vous considérez votre suspension comme un filtre, quel type de filtre réalise-t-elle ?
 E – C'est un filtre passe-bande.
 M – Je présume que votre suspension ne donne pas entièrement satisfaction... !
 E – Si.
 M – Mais encore ?...
 E – Elle est souple.
 M – Mmm... Je présume que votre suspension ne donne pas entièrement satisfaction... !
 E – []
 M – Est-ce que, par exemple, votre suspension peut entrer en résonance ?
 E – Non.
 M – C'est surprenant, dans la mesure où votre suspension est un filtre passe-bande !
 E – Pourquoi ?
 M – [la machine explicite le paradoxe]
 E – [l'élève reconnaît l'entrée en résonance]
 M – Votre suspension n'est pas en très bon état, n'est-ce pas ?
 E – Exact.

(adapté d'une interaction réelle entre SAVANT 3 et un étudiant).

Extrait de dialogue 2

M – Si vous considérez votre suspension comme un filtre, quel type de filtre réalise-t-elle ?
 E – C'est un filtre passe-haut.
 M – C'est surprenant, dans la mesure où l'on suppose que le système de suspension est transparent aux très basses fréquences !
 E – Il n'est pas transparent aux TBF.
 M – Donc vous considérez que la voiture ne suit pas les lentes variations du terrain.
 E – OK, il est transparent aux TBF. Il doit couper les hautes fréquences.
 M – Donc vous considérez que votre suspension n'est pas un filtre passe-haut.
 E – C'est un passe-bande, mais déplacé vers le bas.

(adapté d'une interaction réelle).

Extrait de dialogue 3

M – Si vous considérez votre suspension comme un filtre, quel type de filtre réalise-t-elle ?
 E – C'est un filtre passe-bas.
 M – Considérez-vous votre suspension comme « dure » ?
 E – Oui.
 M – Que pensez-vous de la « fréquence de coupure » de votre suspension, considérée comme un filtre ?
 E – Elle est assez élevée.
 M – OK

(adapté d'une interaction réelle).

On conçoit que les conversations, en traquant ainsi les incohérences, mettent en jeu un mécanisme de réparation logique, mécanisme qui est reproduit dans SAVANT 3.

La conversation didactique

Considérons l'extrait de dialogue n°1. Ce dialogue très simple était proposé après une introduction sur les différents types de filtrage. La machine, à travers ses premières répliques, cherche à amener un paradoxe. Les répliques de l'étudiant lui permettent effectivement d'enfermer ce dernier dans une contradiction. Le pari pédagogique de cette démarche est double :

- l'étudiant est amené à corriger par lui-même ce qui est probablement une définition imparfaite du concept de filtre passe-bande ;
- la contradiction étant une stratégie omniprésente dans les conversations naturelles, on est en droit d'espérer que l'élève ne la ressent pas comme une agression, mais au contraire prend plaisir à argumenter avec la machine.

Il est impossible de connaître avec précision le défaut de conception qui est à la base d'une réponse erronée de l'élève. Il n'est même pas certain qu'il y en ait un, mais, s'il existe, le mérite de cette approche est de le mettre indirectement en évidence avec une bonne probabilité et d'amener l'élève à le corriger par lui-même. Mais il faut bien comprendre que ce processus de correction ne se produit que parce que l'élève et la machine argumentent.

Les concepts à l'épreuve de l'argumentation

Dans l'extrait précédent, la machine adapte son argumentation aux répliques de l'élève. Au début, elle s'attendait en fait à la réponse *filtre passe-bas*. Confrontée à la réponse *passe-bande*, elle choisit d'argumenter sur le confort et l'état de la suspension. Mais la différence essentielle entre ce type d'argumentation et les dialogues tutoriels habituels, c'est qu'ici, il n'y a plus de bonne ou de mauvaise réponse. La machine ne se contente pas de refuser une réponse aberrante sans argumenter (cf. extrait 2). Toute réponse, même acceptable, doit être défendue par l'élève. C'est le cas pour *passe-bande* dans l'extrait 1, ou même pour la réponse « normale » *passe-bas* dans l'extrait 3. Il importe de bien comprendre que ces dialogues ne sont pas prévus à

l'avance, mais que les répliques de la machine sont en partie fabriquées au moment de l'interaction. Voyons pour cela comment SAVANT3 gère sa connaissance logique, ce qui nous permettra ensuite de vérifier que la part de connaissance contingente est très réduite.

La stratégie argumentative de SAVANT3

Le module conversationnel de SAVANT3, dans sa version actuelle, a une stratégie simple : il essaie constamment de déceler un paradoxe dans les propos de l'élève. Il sélectionne donc dans sa connaissance logique une clause paradoxale (cf. encadré 2), dont aucun terme n'est encore invalidé, c'est-à-dire dont tous les termes sont soit vrais, soit non encore déterminés. La vérité ou la fausseté des énoncés est établie un peu comme dans les systèmes experts. Ici, c'est le moteur de clauses paradoxales qui en est chargé (cf. encadré 3).

Lorsque tous les termes du paradoxe peuvent être prouvés, la machine proteste, comme dans l'extrait 1, et l'élève est invité à constater la contradiction et à y remédier. Mais lorsque certains termes restent indéterminés, la machine lance une réplique destinée à amener l'élève à préciser ces termes : par exemple, la machine suggère que la suspension ne donne pas entière satisfaction, en espérant que l'élève parlera spontanément de la résonance ; C'est la stratégie de base (figure 3).

En fait, SAVANT3 exploite certaines stratégies afin d'éviter de piéger l'élève. Tout d'abord, l'élève a toujours la possibilité d'exprimer son désaccord, sous la forme d'un message aux auteurs, ce qui a pour conséquence immédiate l'invalidation du paradoxe en cours. Par ailleurs, chaque fois que cela est possible, SAVANT3 feint la surprise de manière prématurée, par exemple après avoir prouvé tous les termes d'un paradoxe sauf un. L'élève peut alors saisir l'occasion qui lui est donnée d'avoir le beau rôle en invalidant le paradoxe, tel que dans l'extrait de la figure 4. Lors de la discussion sur le problème de la transposition d'un sujet à un autre dans un produit d'EAO, il est apparu que la connaissance contingente, celle qui dépend du sujet enseigné, envahissait tellement le système que la transposition était souvent impossible. Qu'en est-il pour SAVANT3 ?

3. Le moteur de clauses paradoxales

La gestion d'une base de connaissances stockées sous forme de clauses paradoxales commence par la possibilité de prouver certains énoncés. Si, pour simplifier l'écriture, nous considérons des propositions, alors une proposition p peut être prouvée de deux façons. Soit p constitue une clause à elle seule :

$$(\text{non } p) \Rightarrow \text{FAUX}$$

et la preuve est terminée, soit p peut être trouvée dans une clause de la forme :

$$(q_1 \text{ et } q_2 \text{ et } \dots \text{ et } q_n \text{ et non } p) \Rightarrow \text{FAUX}$$

et on doit prouver $q_1, q_2, \dots, q_n$ de la même façon, avec toutefois l'interdiction d'utiliser p dans ces démonstrations. Lorsque son emploi est autorisé au cours du traitement, la modalité INDÉSIRABLE (voir encadré 2) n'est pas distinguée de FAUX par le moteur de clauses paradoxales.

Un moteur chargé de mener de telles preuves a été écrit en Prolog. En faisant jouer un rôle tout à fait symétrique à l'ensemble des prédicats contenus dans les clauses, il peut donc démontrer des prédicats qui seraient indémonstrables si la base de connaissances était par exemple directement exprimée sous la forme de clauses Prolog. Notons au passage qu'il travaille en monde « ouvert », autrement dit qu'il n'assigne pas la valeur « faux » à une proposition qu'il ne sait pas démontrer. Il ne le fait que s'il parvient à démontrer son contraire. Une proposition peut donc avoir trois valeurs de vérité : « vraie », « fausse », et « non encore déterminée ».

En revanche, ce que l'on gagne en capacité de démonstration est compensé par une perte d'efficacité. Dans une connaissance où les hypothèses et les conclusions ne sont pas distinguées a priori, le nombre accru de combinaisons, qui fait la richesse de la connaissance stockée sous cette forme, handicape quelque peu le chaînage arrière. Cependant, pour des applications comme la gestion logique de dialogues, le bilan est incontestablement positif.

Fig. 3 Architecture logicielle de SAVANT3

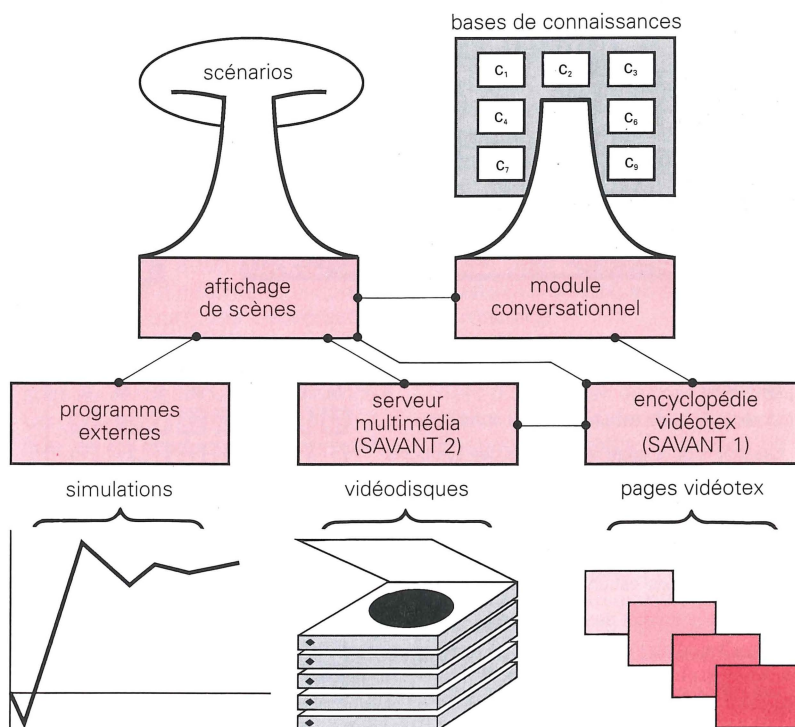
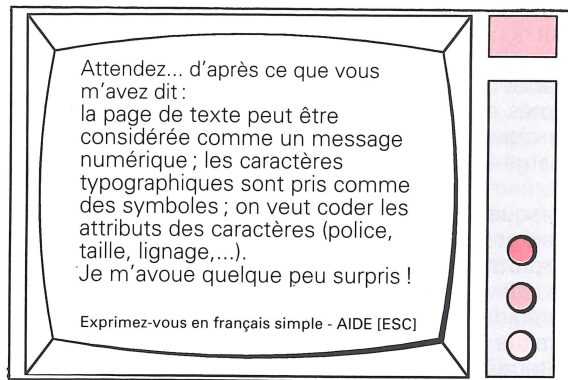
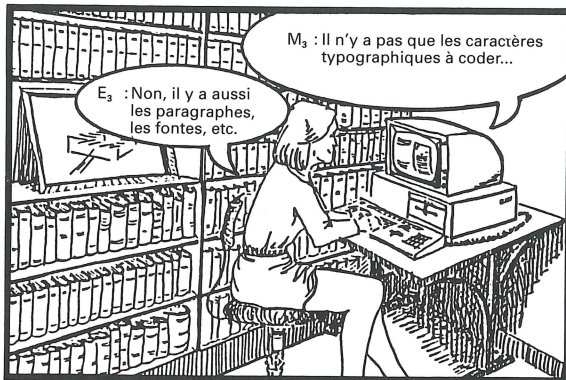
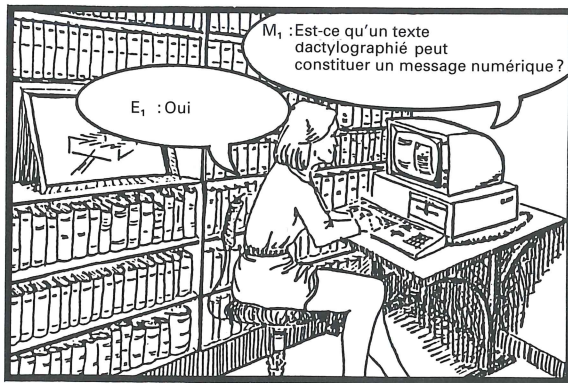


Fig. 4 Un exemple de stratégie de dialogue



paradoxe 1

La page de texte est un message numérique
Les points de la page ne sont pas des symboles
On veut coder les graphiques, la signature,...

M ₁	E ₁	M ₂	E ₂	M ₃	E ₃	M ₄	E ₄
?	V	V	V	V	V	V	V
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	?

paradoxe 2

La page de texte est un message numérique
Les caractères sont des symboles
On veut coder les tailles, les polices, le lignage,...
Les symboles sont limités aux caractères

M ₁	E ₁	M ₂	E ₂	M ₃	E ₃	M ₄	E ₄
?	V	V	V	V	V	V	V
?	?	?	V	V	V	V	V
?	?	?	?	?	V	V	V
?	?	?	?	?	?	?	F

Dans le dialogue, il s'agit d'une page de texte que l'on veut transmettre. La connaissance qui permet à la machine de dialoguer avec l'élève est stockée sous forme de paradoxes. Avec ses deux premières questions M₁/M₂, la machine essaie de vérifier le paradoxe 1. Mais après E₂, elle préfère se rabattre sur le paradoxe 2, dont deux des quatre termes sont maintenant vérifiés, qu'elle feint de croire totalement vérifié en M₄. Mais le dernier terme du paradoxe est encore indéterminé et l'élève a beau jeu de l'invalider en E₄.

Le fonctionnement de SAVANT3

Tout est conçu, dans SAVANT3, pour faciliter la transposition, c'est-à-dire pour limiter la part de connaissance contingente. Chaque dialogue est conduit à partir d'une connaissance extrêmement limitée. Par exemple, la connaissance qui a servi dans les dialogues des extraits 1 à 3 tient en une page de listing ! Une telle connaissance comporte trois types de données :

- une base de règles qui contient quelques clauses paradoxales ;
- certaines répliques prévues par avance ;
- des combinaisons de mots-clés.

Les répliques préconçues sont exprimées lorsque la machine a besoin d'amener l'élève à préciser la valeur de vérité de l'un des prédicats contenus dans les clauses de la base de règles. C'est le cas par exemple de la question M_2 : « Quels sont les symboles que vous considérez ? » dans l'extrait de la figure 4. La machine a besoin de connaître la vérité ou la fausseté de propositions comme « les caractères constituent des symboles du message » ou « les points (pixels) de la page de texte constituent des symboles du message ». Quant aux mots-clés, le programme les utilise pour détecter dans les répliques de l'étudiant l'affirmation ou la négation de l'un de ces prédicats.

Le découpage de la connaissance en petits modules indépendants présente un grand intérêt, car il facilite considérablement la mise au point des règles (assurer la cohérence, la complétude et la non-redondance d'un ensemble de quinze règles est tâche aisée !). Mais ce découpage n'est possible que grâce aux contraintes que l'interlocuteur humain s'impose à lui-même. En effet, il ne changera pas spontanément de sujet tant qu'un problème reste en suspens. Et, comme il assure la pertinence de ses propres répliques, la machine peut espérer reconnaître celles-ci dans les quelques prédicats contenus dans sa connaissance locale.

Le système auteur lié à SAVANT3 contient un éditeur qui permet à l'auteur de constituer cette connaissance assez facilement. La façon dont les répliques sont choisies ou enchaînées par la machine est déterminée au moment de l'exécution par le module conversationnel et l'auteur n'a donc pas à s'en préoccuper. De plus, l'équipe de Télécom

Paris travaille actuellement à un programme qui permettra à un auteur non logicien d'explicitier facilement sa connaissance sous la forme d'un ensemble de règles dont le système assure qu'il sera complet, cohérent et non redondant. Idéalement, l'auteur devrait pouvoir « accoucher » de cette connaissance au cours d'une conversation. Mais nous n'en sommes pas encore là...

Rappelons toutefois que SAVANT3 ne se limite pas à son module conversationnel et que la connaissance contingente n'y est pas limitée à ce qui vient d'être décrit. La figure 3 montre l'architecture complète de SAVANT3. On constate qu'il s'agit d'un système complet, faisant appel à des exposés « narratifs » graphiques et sonores, à des programmes de simulation, à une encyclopédie conceptuelle (il s'agit de l'encyclopédie SAVANT1¹²) et à des ressources vidéo (comme le serveur multimédia SAVANT2). Mais tous ces modules sont strictement cloisonnés, ce qui limite la difficulté de la transposition.

SAVANT3 est encore dans l'enfance. Développé en 1988, il fut expérimenté une première fois en 1989 auprès des étudiants de première année de Télécom Paris, sur un cours de signaux et systèmes continus. Les retours, avant tout qualitatifs, de ces premiers essais permettent déjà d'améliorer le module conversationnel qui fait l'originalité de ce système. Mais nous comptons aussi intégrer les retombées des études menées sur la simulation des conversations¹¹, en donnant la priorité au respect de la pertinence.

SAVANT3 : le respect de la pertinence

L'observation attentive des échanges naturels d'informations conceptuelles au cours des conversations spontanées montre que les interlocuteurs apportent énormément de soin au respect de la pertinence⁹. Un locuteur n'émet une réplique que si sa traduction sous forme logique vient « s'ancrer » sur les constructions logiques établies au cours des répliques précédentes. Par exemple, une proposition venant modifier le paradoxe de départ, pour l'empêcher de s'appliquer à la situation présente, satisfait aux exigences de pertinence. Il n'en serait pas de même pour une proposition qui n'interférerait pas logiquement avec les répliques précédentes.

Cette exigence de pertinence se comprend bien si l'on reconnaît qu'elle doit considérablement faciliter la gestion de la connaissance logique. Les termes qui entrent dans les formules logiques élaborées, par exemple au cours des conversations, viennent se mettre en place l'un après l'autre et n'apparaissent donc qu'en contexte. Les interlocuteurs n'ont pas besoin de mémoriser des éléments dépourvus de contexte logique, contrairement à ce qui est demandé aux élèves dans l'enseignement magistral traditionnel.

Le développement du programme PARADISE¹¹ a permis d'explorer toutes les possibilités d'ancrage logique et donc les conditions de la pertinence. Si l'on parvient à imposer à la machine d'ancrer ses répliques sur celles de l'étudiant, le dialogue semblera beaucoup plus naturel, avec notamment une part beaucoup plus réduite de questions. Par ailleurs, il est prévu d'incorporer la possibilité de converser sur le mode (in)désirable, ce qui sera fort utile pour gérer des dialogues sur des sujets technologiques. Par exemple, le choix d'un type de codage numérique donné peut n'avoir aucun aspect paradoxal, mais peut se révéler fortement indésirable dans certaines conditions (s'il s'agit d'un codage redondant) ou désirable dans d'autres conditions (si ce même codage est rapide ou résistant aux erreurs).

Perspectives

Les premiers résultats sont encourageants, car ils établissent la faisabilité de cette nouvelle approche du dialogue didactique. En effet, une bonne partie des étudiants acceptent d'entrer dans la conversation initialisée par SAVANT3 et s'y comportent de manière naturelle, malgré le caractère pour l'instant assez fruste des stratégies employées par la machine.

Le prochain travail va être maintenant d'améliorer de manière significative la souplesse de l'interaction. Ensuite, de rendre la gestion de la connaissance de la machine plus sophistiquée, à la fois horizontalement (possibilité de changer de sujet) et verticalement (gestion des niveaux de détails). Enfin, puisque le moteur de clauses paradoxales est capable de gérer des prédicats, il sera possible de reconnaître des prédicats dans les répliques de l'élève, et pas seulement des propositions.

L'EO à l'heure des sciences cognitives

La « renaissance cognitive » que nous connaissons depuis ces dernières années, en partie grâce au développement des systèmes experts, est loin d'avoir fini la conquête de l'EO, domaine où pourtant son influence sera certainement décisive. Les systèmes experts tirent leur succès, non d'une prouesse technique, mais de l'idée (autrefois hardie, mais maintenant bien admise) que l'expertise humaine pouvait être traduite (en partie) sous forme logique. Cet article a pour but d'attirer l'attention sur le fait que tout contenu pédagogique peut, de la même façon, être partiellement traduit sous forme de règles logiques. Paradoxalement, ce sont les savoir-faire à transmettre qui ont été les premiers transcrits dans les programmes d'EO. Cela est étonnant, si l'on pense que ce sont plutôt les concepts qui, étant par essence de nature logique, auraient dû profiter les premiers des recherches en ce domaine. Le travail mené dans le cadre de SAVANT3 vise à montrer que les conséquences de ce constat sont dignes d'être explorées. ■

Remerciements

Le projet SAVANT3, comme ses prédécesseurs SAVANT1 et SAVANT2, a pu voir le jour à Télécom Paris grâce à la volonté constante dont fit preuve Daniel Cadé de promouvoir l'innovation en matière de techniques d'enseignement. Les développements actuels de ce système doivent beaucoup à l'environnement scientifique privilégié dont on peut jouir au Département Informatique de Télécom Paris. Je remercie en particulier Alain Grumbach pour ses conseils et suggestions. J'ai apprécié l'aide apportée par Georges Sallé dans la réalisation du projet et pour les illustrations de cet article.