Association de Psychologie Scientifique de Langue Française
Symposium Intelligence Naturelle et Intelligence Artificielle
Rome, 23-25 septembre 1991

LE DEVELOPPEMENT DE L'ENSEIGNEMENT INTELLIGEMMENT ASSISTE PAR ORDINATEUR

Patrick MENDELSOHN
Pierre DILLENBOURG[1]

Introduction

Depuis les premiers systèmes d'EAO, mis au point dans les années 60, les chercheurs se sont efforcés de construire des logiciels dont l'objectif est d'améliorer les conditions dans lesquelles l'enseignement est dispensé dans des domaines bien définis. TICCIT et PLATO (Morisson, 1975; Murphy et Appel, 1977) sont deux exemples sur une grande échelle de ce que peuvent être des projets dont le but explicite est de promouvoir un enseignement plus efficace et à un moindre coût. Ces environnements proposent tout une game d'activités autour du jeu situation-problème et des simulations interactives, mais sans réellement s'intéresser aux raisons de l'efficacité des systèmes proposés.

Depuis quelques années en revanche on peut noter que les recherches sur l'apprentissage humain et plus généralement sur la cognition ont recentré les applications de l'informatique sur des problèmes plus directement liés à l'apprentissage qu'à l'enseignement. Une première conséquence de cette évolution est que certains systèmes reposent maintenant sur des modèles du fonctionnement cognitif ou plus généralement sur des théories (LOGO est un bon exemple) dont il est possible d'étudier ainsi les contraintes d'implémentation. Leur statut initial d'outil au service de l'enseignement s'est parfois transformé en celui de laboratoire pour le chercheur en Sciences Cognitives. L'intrusion des techniques d'intelligence artificielle dans ce domaine autrefois réservé aux classiques architectures dites "Tourne-Page" a largement contribué à cette évolution.

Notre objectif est de proposer une vue générale sur la contribution de l'IA au développement de l'enseignement assisté par ordinateur. L'Intelligence Artificielle sera pris ici comme un ensemble de techniques informatiques qui apportent à l'ordinateur une certaine compétence à résoudre des problèmes dans des domaines comme ceux du raisonnement ou de la compréhension du langage naturel. Nous soulignons qu'il n'entre pas dans notre propos de nous intéresser ici à la simulation des processus cognitifs humains impliqués dans l'apprentissage.

---

[1] TECFA - Faculté de Psychologie et des Sciences de l'Education, Université de Genève
CH - 1211 - Genève 4

Pour caractériser cette évolution, nous reprendrons partiellement une thèse défendue il y a quelques années par Park et Tennyson (1983) qui constataient à l'époque un déplacement significatif de l'intérêt des chercheurs. Si la conception des sytèmes reposait au départ exclusivement sur des méthodes d'enseignement dont il apparaît en général qu'on ignore les effets sur les processus d'acquisition, on constate maintenant que les environnements d'apprentissage représentent des tentatives pour établir des connections directes entre telle ou telle variable de situation et les performances du sujet. Par la suite, les premiers seront qualifiés de systèmes "orientés modèles d'enseignement", les seconds de systèmes "orientés théories de l'apprentissage". La plupart des systèmes classiques d'EIAO relèvent de la première catégorie. Ils sont orientés vers la démonstration que des techniques d'Intelligence Artificielle (par exemple, le traitement du langage naturel ou encore l'utilisation de systèmes experts) peuvent être appliquées au domaine de l'enseignement sans préoccupation de validation expérimentale des effets de ces systèmes sur telle ou telle variable liée aux processus d'acquisition. A l'opposé, les environnements reposant sur des théories de l'apprentissage comme LOGO, les tuteurs de J.R. Anderson (Anderson et Reiser, 1985) ou encore plus modestement celui que nous développons à Genève sont directement utilisables comme des stations expérimentales pour valider une composante ou un modèle bien précis de l'apprentissage.

Cette distinction n'est pas classiquement admise dans la littérature. Mais elle nous semble tout à fait pertinente à l'heure où les frontières entre les disciplines que nous représentons sont l'objet d'un débat animé. Sur le problème des aides fournies à l'élève par exemple, il est fréquement invoqué par les chercheurs qui s'intéressent aux environnements informatiques d'apprentissage que la représentation spatiale de l'historique d'une procédure de résolution de problème (comme dans ALGEBRALAND ou GEOMETRY-TUTOR) contribue à une meilleure gestion de la planification de la tâche sans validation directe des effets de cette représentation, ni du domaine de validité d'une telle assertion.

L'EIAO classique: les systèmes "orientés modèles de l'enseignement"

Les applications des techniques et des principes de l'Intelligence Artificielle à l'enseignement ont donné naissance à un certain nombre de dénominations pour caratériser les produits qui en sont issus. Parmi les plus fréquement utilisés on peut citer les termes d'"Enseignement Intelligemment Assisté par Ordinateur" (traduction approximative de ICAI), "Tuteur Intelligent" (Intelligent Tutoring System) ou encore Environnement Intelligent d'apprentissage (Intelligent Learning Environment). Nous n'entrerons pas ici dans le débat sémantique qui caractérise les subtiles différences entre ces systèmes, nous dirons simplement qu'ils représentent chacun des compromis différents sur la question du degré de flexibilité dans l'interaction entre l'élève et le programme (le modèle de référence en ce qui concerne cette flexibilité dans le contrôle de l'activité restant presque toujours la relation maître-élève ou élève-élève).

En effet, un système d'EIAO ne doit pas être simplement capable de produire des réponses toutes faites aux interrogations de l'élève, il doit aussi pouvoir utiliser et traiter des connaissances stockées dans le système pour faire face à des questions non-prévues spécifiquement dans l'implémentation. En théorie, un système d'EIAO produit des réponses sur la base de connaissances accumulées au cours d'une session d'apprentissage. Ces systèmes peuvent aussi mener des dialogues en langue naturelle bien que cette dernière caractéristique soit de moins en moins pertinente en raison, d'une part, des progrès des interfaces graphiques

qui offrent des solutions moins coûteuses et, d'autre part, des difficultés inhérentes à la production du langage naturel.

Il existe plusieurs manières de classer les systèmes d'EIAO. D'un point de vue historique, on peut considérer comment les systèmes se sont regroupés spontanément en familles de logiciels autour d'une problématique bien définie. On peut aussi les catégoriser en compréhension à partir de leurs composantes. Nous avons choisi de combiner ces deux approches en espérant que le lecteur pourra en tirer un meilleur profit.

Principales composantes d'un système d'EIAO:

Comme c'est le cas pour toute situation d'enseignement, les principales fonctions d'un tutoriel intelligent doivent lui permettre de répondre à trois questions: Quel contenu enseigner ? Comment diagnostiquer les difficultés de l'élève ? Et quelle méthode ou stratégie d'enseignement adopter ? Dans notre jargon de concepteur de systèmes, ces fonctions se réfèrent, dans l'ordre, au module expert, au modèle de l'élève et au modèle pédagogique. A ces trois composantes il faut ajouter une entité qui joue un rôle important bien qu'il soit difficile de la considérer comme une composante fonctionnelle du système, l'interface utilisateur.

L'histoire et/ou la taxonomie des environnements d'EIAO peut s'écrire de différentes façons mais tous ceux qui ont tenté de le faire - en particulier Wenger (1987) - ont constaté que l'évolution des systèmes est souvent liée à une attention plus ou moins grande portée à l'une ou l'autre de ces composantes.

Les premiers systèmes, mis au point au début des années 70, sont principalement centrés sur la représentation des connaissances dans le domaine de référence, par exemple SCHOLAR et SOPHIE (Carbonell, 1970; Brown, Burton et Bell, 1975). A partir du milieu des années 70, on peut facilement constater que c'est le modèle de l'élève et plus précisément le diagnostic des erreurs qui retient l'attention des chercheurs. La tentative la plus représentative reste celle de BUGGY (Brown et Burton, 1978). Simultanément à cet intérêt pour le modèle de l'élève, le module pédagogique et les stratégies d'enseignement commencent à faire l'objet de représentations calculables comme dans WEST (Burton et Brown, 1976). Enfin, ce n'est que tardivement que les chercheurs réalisent que l'interface est une composante de tout premier ordre dans les processus d'apprentissage. En combinant les possibilités de manipulations directes et le réalisme des représentations, les interfaces graphiques actuelles ont contribué à créer un véritable langage avec une sémantique relativement familière et triviale.

L'objectif commun à toutes les projets de recherche en EIAO est de construire un système performant sur ces quatre axes. Les temps de développement et les problèmes rencontrés malheureusement font que bien souvent un travail approfondi dans une des directions citées se traduit par des faiblesses sur les autres axes de recherche. Un tutoriel doit être cependant évalué autant sur sa capacité à maîtriser l'ensemble de ces problèmes que sur les performances du système sur une de ces dimensions.

Le Module Expert

Une des principales caractéristiques d'un tutoriel intelligent, comme pour un enseignant, est d'être compétent dans la matière qu'il enseigne. Imaginons par exemple un logiciel qui enseigne la détection de panne sur un circuit électrique. Le concepteur pourrait prévoir de

faire travailler l'élève sur un certain nombre de cas définis à l'avance et pour lesquels il dispose de solutions toutes faites, qu'il présente à l'apprenant lorsqu'il constate des échecs répétés. En EIAO, l'approche consiste à doter le système d'une capacité de raisonner sur n'importe quel problème du domaine (bien sûr, dans les limites imposées par la syntaxe du langage de commande). Par exemple, le système doit être capable de résoudre un problème que lui proposera l'élève. Il pourra aussi suivre l'élève dans sa démonstration sans que celle ci suive obligatoirement un cheminement prévu d'avance. En outre, cette compétence permet au tutoriel d'exposer son raisonnement et de justifier ses décisions. Une conséquence majeure de cette avancée se retrouve dans l'évolution globale des systèmes d'enseignement assisté par ordinateur qui de la programmation de solutions prédéfinies se sont orientés vers celle des processus de résolution.

Cette composante est appelée modèle du domaine ou modèle de l'expert. Plusieurs systèmes, dont le célèbre GUIDON (Clancey, 1979) ont d'ailleurs, en toute logique, été construits à partir de systèmes experts préexistants comme modèle du domaine. Ces premières tentatives ont cependant montré que les explications fournies par un système-expert (en général, la trace des règles qu'il a utilisées) n'étaient pas suffisantes dans le contexte éducatif. Ce constat est à l'origine d'une évolution des modèles de l'expert (et de certains systèmes-experts) vers, d'une part, des modes de raisonnement plus proches de ceux qu'utilisent les experts humains et, d'autre part, vers une représentation explicite des stratégies de résolution de problème (auparavant implicitement distribuées parmi les prémisses des règles). Grace à ces stratégies, le tutoriel peut avantageusement modifier le raisonnement de l'expert, par exemple pour le contraindre à ne pas utiliser des concepts complètement étrangers à l'élève (sinon ce dernier ne pourrait comprendre ni les résultats, ni les explications de l'expert).

Importance des modèles qualitatifs

Cette approche se complique encore si l'on tient compte du fait que l'élève adopte parfois un point de vue qui diffère radicalement de celui de l'expert. Les développements récents convergent vers la nécessité de doter l'expert de multiples points de vue sur son domaine. Dans certains cas, ces points de vue n'ont pas la même validité et il devient intéressant de les considérer comme des étapes au sein d'une progression pédagogique. L'aprentissage apparaît alors davantage comme une succession de re-conceptualisations (globales ou partielles) du domaine. Cette conception est illustrée par le tutoriel QUEST (White et Frederiksen, 1985) qui prévoit la progression de l'apprenant à travers trois modèles de fonctionnement d'un circuit électrique. Il commence par apprendre à raisonner sur un modèle qualitatif (présence ou absence de courant dans le circuit), en second lieu l'élève est confronté à un modèle semi-quantitatif (augmentation ou diminution de l'intensité) et, en dernier lieu seulement, le modèle quantitatif est abordé (loi d'Ohm, etc.).

Modes de représentation des connaissances

Les modes de représentation utilisés pour le module expert sont très variés et les auteurs de systèmes ont fait appel à presque toutes les solutions disponibles en IA:

- les réseaux sémantiques comme dans SCHOLAR (Carbonell, 1970). La matière à enseigner, ici la géographie de l'Amérique du Sud, est représentée dans un réseau dont les noeuds sont des objets géographiques et des concepts. Du fait de l'organisation hiérarchique de ces objets par des relations du type "superpart", "superconcept" ou superattribute" des inférences simples peuvent être réalisées par propagation. Les limitations inhérentes à ce mode de

représentation sont bien entendu liées au fait qu'il n'est pas possible de représenter des connaissances procédurales sur un tel mode.

- les systèmes de production sont utilisés pour construire des représentations modulaires de compétences avec une organisation en catégories abstraites indépendantes des processus qui permet de réaliser des calculs sur cette connaissance. Une des contributions majeures de Clancey (1983) avec le programme GUIDON est d'avoir proposé une méthodologie pour ce type de modèle dont on peut résumer les étapes essentielles de la manière suivante:

1) analyse du domaine de référence à partir de protocoles d'experts et de novices.

2) construction d'un modèle de l'expert qui tienne compte de toutes nuances possibles de raisonnement.

3) construction du module d'enseignement.

- les schémas qui sont des structures de données incluant des informations déclaratives et procédurales sont maintenant largement utilisés par les concepteurs de programmes en raison des facilités de programmation apportées par les langages de programmation orientés objets.

Tous ces modes de représentation (sans oublier aussi la logique de premier ordre ou les automates à états finis) sont bien sûr interchangeables et il arrive bien souvent que les concepteurs utilisent plusieurs modes de représentation dans un même système pour bénéficier des avantages de chacun.

Le modèle de l'élève

Un logiciel d'EAO classique adapte ses réactions aux comportements de l'élève en choisissant, par exemple, un feedback approprié à l'erreur que l'élève a commise ou en ajustant encore la difficulté de l'exercice au nombre d'erreurs commises au cours de la leçon. Les concepteurs de tutoriels ont comme ambition d'améliorer cette prise d'information, limitée généralement dans les environnements classiques à une réponse ou quelques seuls paramètres quantitatifs. Cette amélioration est réalisée en construisant en quelque sorte un portrait des connaissances de l'élève qui s'enrichit à chaque étape de l'apprentissage. Ce portrait porte le nom de modèle de l'élève.

Pour reprendre l'exemple des circuits electriques cité précédemment, imaginons qu'un élève doive reconnaître un montage en parallèle ou en série à partir d'une dizaine d'exemples présentés à l'écran. Au terme de la série d'exercices, il est évident que connaître le score de cet élève, par exemple 4 réussites sur 10 est bien moins intéressant que de déterminer la cause commune à toutes ses erreurs. Certaines techniques d'IA permettent ainsi de remonter la chaîne d'inférences que le sujet a hypothétiquement suivie afin de déterminer la cause de son échec.

Le tutoriel pourra par exemple découvrir que l'élève a appliqué la formule du calcul de résistance en parallèle chaque fois que le circuit présentait des résistances géométriquement parallèles (mais fonctionnellement en série). Le modèle de l'élève sert ainsi à construire un diagnostic qui pourra servir au système pour prendre une décision de nature didactique (questionnement, explications, remédiation...).

Le modèle de recouvrement (overlay model)

Dans les systèmes d'EIAO, l'état des connaissances de l'élève est généralement représenté comme un sous-ensemble des connaissances du module expert. Dès lors, le modèle de l'élève est construit en comparant la performance de l'apprenant avec celle que l'expert aurait produite dans les mêmes circonstances. Cette technique a été baptisée "overlay paradigm", elle est applicable chaque fois que l'expertise est représentée sous la forme d'un système de règles de production. Le tutoriel "WUSOR", adapté sur un jeu vidéo, utilise une technique de ce type et Goldstein (1982) rappelle que la mise à jour du modèle de recouvrement repose sur plusieurs sources d'information:

- les informations implicites qui découlent de la comparaison entre le comportement de l'élève et les décisions de l'expert. L'acquisition de ces informations dépend de la capacité du concepteur à relier un comportement donné à un ensemble bien défini de réalisations effectives. C'est sur ce point peut-être que la collaboration avec le didacticien s'avère la plus fructueuse (sémantique des actions).

- les informations structurelles qui relèvent du réseau de dépendances et de complexité relative aux différentes compétences en jeu. Ce réseau détermine en quelque sorte les frontières à l'intérieur desquelles l'apprentissage doit se centrer.

- les informations explicites qui peuvent être obtenues par un questionnement direct de l'élève (test ou questionnaire).

- les prérequis supposés, inférés ou stockés lors d'une précédente séance pour initialiser le modèle.

Deux limitations sont généralement invoquées à propos de ces techniques de recouvrement: la première difficulté tient au problème plus général de la représentation des connaissances dans le module expert, étant donné que la représentation de celles de l'élève en découle directement. Les connaissances du novice correspondent bien souvent non pas à un sous-ensemble des connaissances de l'expert mais à une conceptualisation globalement différente du domaine. Pour dépasser cette difficulté, il faudrait imaginer pour chaque problème plusieurs modèles d'experts régulés par un meta-expert qui choisirait le modèle approprié à la démarche de l'élève. La seconde difficulté est liée au modèle lui-même. Le paradigme de recouvrement présuppose qu'une performance non optimale est le résultat d'un mauvais choix stratégique et non celui d'une erreur opératoire commise lors du déroulement d'une stratégie dont la validité est pourtant avérée. Cette nuance importante sera au centre des préoccupations des modèles dits "buggy models".

Le modèle des erreurs (buggy model)

Une autre approche consiste à utiliser les erreurs les plus fréquemment rencontrées chez les élèves comme des variantes possibles du modèle de l'expert (Brown et Burton, 1978). BUGGY et les systèmes DEBUGGY et IDEBUGGY dans le domaine de l'enseignement de l'algèbre sont les exemples les plus représentatifs de cette approche. Ils ont démontré à la fois la faisabilité et la pertinence, pour un domaine restreint donné, de la mise au point d'un catalogue complet d'erreurs de nature procédurale. Les implications de recherches de ce type sur l'étude des processus d'acquisition tiennent au fait que comprendre l'origine des "bugs" (en

étant capable de les reproduire artificiellement) a des conséquences non négligeables sur les stratégies pédagogiques de remédiation.

A quoi sert le modèle de l'élève ?

C'est peut être Self (1988) qui a le plus clairement analysé le rôle et l'usage effectif du modèle de l'élève dans les tuteurs intelligents. Il a pu ainsi dégager six fonctions principales: correction de l'erreur, aide à l'élaboration d'une action à entreprendre quand l'erreur est due à une procédure incomplète, mise en oeuvre d'une stratégie globale pour proposer un nouveau plan d'actions, construction d'un diagnostic, prévision d'actions futures (fonction prédictive) et enfin évaluation du travail fourni par l'élève.

L'ambiguité du statut du modèle de l'élève illustre bien la complémentarité des collaborations qui sont nécessaires entre les psychologues et les informaticiens. Les premiers s'intéressent en priorité à la validité psychologique du modèle construit. Celle ci est évaluée en comparant les diagnostics produits par le système et ceux produits une démarche expérimentale. Pour Self (1988), par contre, l'informaticien devrait garder à l'esprit l'utilité pédagogique du diagnostic construit, c'est à dire la décision pour laquelle la prise de décision s'avère nécessaire. Ce principe fut appliqué à PROTO-TEG (Dillenbourg, 1989), un système qui construit ses critères de diagnostic directement en observant pour quel type d'élève chacune de ces stratégies pédagogiques est efficace.

Ce modèle de l'élève peut être aussi utilisé pour différencier les erreurs dues à des connaissances mal apprises ou mal utilisées de celles qui auraient pour origine la fatigue, la distraction ou encore une charge cognitive trop importante. Le chantier ouvert par la modélisation de l'apprenant, on le voit, révèle des problèmes théoriques et méthodologiques qui ne devraient pas laisser les psychologues indifférents.

Le module pédagogique

En tant qu'héritier des bases théorique et idéologique de l'enseignement programmé, l'EAO classique a privilégié des méthodes pédagogiques minimisant le risque d'erreur de l'apprenant en forçant l'apprenant à rester dans le "droit chemin" des connaissances correctes. A l'opposé et sous l'influence des théorie constructivistes qui soulignent l'importance de l'activité du sujet dans la construction des connaissances, les partisants de LOGO ont mis l'accent sur la valeur éducative des erreurs que commet l'élève lorsqu'il explore librement un micromonde. Le défi des concepteurs de tutoriels intelligents est d'adopter une voie intermédiaire qui permet à l'élève d'explorer ses propres solutions tout en étant capable de lui fournir une aide lorsqu'il est "perdu", qu'il tourne en rond ou que ses erreurs ne lui apportent plus d'informations pertinentes.

En raison de la libertée accordée à l'élève, l'intervention du tutoriel ne peut pas être prédéfinie, elle doit être élaborée sur la base des informations présentes dans le modèle de l'élève et dans le module expert du domaine. Un module pédagogique est donc un ensemble de spécifications sur la manière dont le système doit construire ses interventions. Ce module interagit avec l'élève plus ou moins directement en sélectionnant les problèmes qu'il doit résoudre, en le guidant vers la solution, en critiquant ses performances, en lui fournissant une aide appropriée lorsque l'élève le.lui demande, en montrant des exemples.

Le module pédagogique peut être plus ou moins complexe suivant les prétentions des concepteurs dans ce domaine. Il peut mettre en oeuvre simplement un système de règles de production sur la base de connaissance fournie par le modèle de l'élève. Il peut en outre être capable d'élaborer de véritables dialogues avec l'élève comme dans SCHOLAR ou SOPHIE (Carbonell, 1970; Brown et al., 1975). Il peut aussi tenter d'adapter les méthodes de planification dynamique développée en IA (blackboard architecture) afin de permettre au tutoriel de réorganiser ses propres plans en fonction des réactions de l'élève.

Deux modèles d'interaction sont le plus souvent utilisés dans les tuteurs existants:

- la méthode socratique qui consiste à questionner l'élève de manière à le faire progresser dans ses connaissances sur un sujet donné en traquant ses erreurs. Cette méthode a été utilisée avec un certain succès dans les premiers systèmes tuteurs: SCHOLAR, SOPHIE et WHY. Sur ces tuteurs qui sont orientés vers l'acquisition de connaissances plus factuelles que procédurales, l'initiative mixte permet d'obtenir une certaine souplesse dans la conversation (le tuteur ou l'élève peuvent chacun poser des questions). Le seul inconvénient de cette stratégie est un risque de déstructuration du dialogue si l'initiative mixte est utilisée de manière trop intempestive.

- le guidage dans un environnement de type micromonde, ou sur une simulation, consiste à procurer une aide interactive à un élève qui se trouve engagé dans la résolution d'un problème par l'intermédiaire d'un langage de commande. Cette méthode se retouve dans les tuteurs WEST ou STEAMER dont nous avons déjà parlé. Dans le cas de WEST par exemple, le guidage est particulièrement discret et efficace grace à une bonne analyse des heuristiques du sujet. Ceci est rendu possible par la faible étendue de l'espace du problème. Le guidage obéit ainsi à un certains nombre de principes plus ou moins explicites qu'il est possible de résumer de la manière suivante:
    - le pointage d'opportunité: ce rôle consiste à faire prendre conscience à l'utilisateur des possibilités non encore explorées du système.
    - la transformation des échecs en expériences: le guide doit pouvoir détecter les erreurs de parcours en rapport avec un objectif donné et proposer de nouveaux points de départ sans pour autant systématiquement empêcher le sujet de se tromper.

Pour Burton et Brown (1979), ces objectifs se traduisent par un certains nombre de règles de bonne conduite de la part du guide:

1) interventions effectives du tuteur pour éviter les effets "plancher" et "plafond" susceptibles de détruire l'intérêt pour la tâche à réaliser.

2) interventions pertinentes du point de vue du déroulement de la tâche. Cela suppose un monitoring efficace du plan d'action suivi par le sujet.

3) non intervention s'il s'avère qu'elle est non pertinente.

La composante dialogue est cependant moins utilisée dans les systèmes les plus récents en raison des difficultés inhérentes à la production du langage naturel. Elle est progressivement remplacée par des messages organisés sous la forme d'hypertexte ou par des représentations graphiques sophistiquées qui permettent à l'élève d'observer ses propres stratégies comme dans ALGEBRALAND (Brown, 1983) ou GUIDON (Clancey, 1983).

L'interface

Comme nous venons de le voir, la conception des stratégies est fortement assujettie aux modalités de communication entre l'élève et l'ordinateur. La qualité de cette interface est fondamentale afin que les échanges entre l'élève et le système ne soient pas réduits à des échanges de messages codés.

Malgré les difficultés inhérentes au traitement du langage naturel, il reste que le dialogue est une des manières par lesquelle l'élève exerce sa capacité pour construire une argumentation. Certains tutoriels récents vont plus loin que les prototypes SCHOLAR et SOPHIE dans cette direction, c'est le cas de DECIDER, élaboré par Bloch et Farrel (1988). La tâche de l'élève est de proposer une forme d'intervention américaine au Nicaragua. Le tutoriel ne cherche pas à enseigner ce que serait une bonne ou une mauvaise intervention, il l'ignore lui-même. Il se contente de contrer les propositions ou les arguments de l'élève en décrivant des événements historiques (Vietnam, Cuba..) au cours desquels une intervention similaire n'a pas donné les résultats escomptés. De plus, il observe la façon dont l'élève tient compte de ses contre-arguments. La description verbale de ces événements est accompagnée de la présentation d'une brève séquence vidéo.

D'autre part, les ordinateurs actuels et certains logiciels utilisés dans notre secteur de recherche offrent des possibilités graphiques très sophistiquées. La conception graphique de l'interface apparaît de plus en plus comme une composante cruciale dans la conception d'un tutoriel. Sans faire référence aux seules qualités esthétiques ou décoratives que peuvent avoir un logiciel, il est possible d'augmenter la qualité de l'interaction avec l'élève par une représentation plus réaliste et bien organisée des informations. L'usage combiné de la souris et des fenêtres a accru les possibilités de manipulation directe des objets, comme dans les situations de construction d'une figure en géométrie ou d'un circuit electrique par exemple. Par ailleurs, plusieurs travaux récents visent à amener l'élève à participer à la construction ou à observer une représentation spatialisée d'une entité abstraite telle que sa représentation du problème ou son cheminement dans le graphe du problème (Collins et Brown, 1987). En devenant observables, ces entités permettent de concrétiser des activités de niveau méta-cognitif, telles que discuter de "ce que l'on a compris de l'énoncé" ou de l'efficacité d'une stratégie de résolution de problème.

L'émergence actuelle des théories dites "situated learning" accroît encore l'importance du rôle de l'interface au sein d'un logiciel d'enseignement. L'écran n'est plus vraiment pensé comme une interface au sens d'un medium permettant l'accès aux représentations symboliques internes du modèle de l'expert. Dans cette perspective, l'écran devient moyen de construction par l'apprenant des représentations externes indispensables à la résolution d'un problème. Son statut dans les processus d'apprentissage s'en trouve ainsi modifié.

Limites des systèmes d'EIAO "orientés modèles d'enseignement"

Ce rapide tour d'horizon sur les systèmes exitants montre le formidable potentiel de recherche offert par ce domaine d'application de l'IA, mais il fait ressortir aussi les limitations inhérentes à cette entreprise. Tout d'abord il existe très peu de systèmes qui dépassent le simple stade de prototype, quand ce n'est pas celui de maquette. Ensuite, les performances de ces systèmes ne sont pas satisfaisantes. En 1982, Sleeman et Brown décrivaient les défauts de ces tutoriels de la manière suivante:

1) le niveau de réponse à un questionnement ou à une erreur commise par l'élève est souvent à un mauvais niveau de détail; les systèmes anticipent mal le niveau de connaissance du sujet.

2) aucun des systèmes dont nous avons parlé ne découvrent vraiment les conceptualisations idiosyncratiques des élèves qui leur permettraient de diagnostiquer efficacement ses "bugs".

3) les stratégies tutorielles et les critiques sont ad hoc et reflètent des intuitions souvent sans fondement sur la manière dont il faudrait gérer le comportement de l'apprenant.

4) l'interactivité système-élève est encore trop souvent dominée par les interventions du système.

Ces défauts pourraient être encore, plus de dix ans après, attribués à la plupart des systèmes actuels bien que sur certains points des progrès notables aient été accomplis. Cela tient au fait qu'ils sont liés à des limitations propres aux différentes disciplines impliquées dans le développement des tutoriels intelligents:

1) le manque de moyens pour communiquer efficacement entre l'élève et l'ordinateur est un obstacle majeur que tous les développeurs rencontrent. A l'heure actuelle, le dialogue en langue naturelle n'est possible qu'à l'intérieur de situations fortement structurées.

2) la possibilité de saisir les connaissances d'un élève à partir du module expert est très limitée car elle suppose que l'on soit capable de comprendre les différents procédés de raisonnement d'un élève singulier.

3) la quantité de travail méthodologique et technique requise par la spécification du domaine de connaissance et par celles des stratégies d'enseignement est très importante. L'obligation de travailler en général sur des systèmes puissants n'allège pas vraiment le travail de développement.

4) les techniques IA sont plus difficilement applicables à des domaines moins bien structurés que l'algèbre, la programmation, la géométrie ou encore l'electricité.

Ces limitations sont accentuées par le fait que les systèmes dont nous avons parlé jusqu'à présent ont mis l'accent presque exclusivement sur les capacités d'interaction entre l'homme et la machine. La plupart des problèmes abordés par les logiciels d'EIAO sont très simples et limités à des domaines bien définis sans espoir de pouvoir à brève échéance généraliser leurs résultats. Une manière de sortir de cette impasse, qui lie les progrès des tutoriels aux progrès des techniques d'intelligence artificielle, serait de fonder plus intimement l'architecture et la conception de ces systèmes sur les résultats des recherches dans le domaine de l'apprentissage et de l'instruction. C'est une voie initiée il y a déjà quelques temps par Anderson (Anderson et Reiser, 1985) avec GEOMETRY et LISP TUTOR (qui a produit d'ailleurs les seuls systèmes que l'on trouve dans le commerce); ainsi que Tennyson, Christensen et Park, (1985) avec MAIS. Si l'on compare les objectifs et les méthodes de l'EAO et de l'EIAO, on se rend compte que l'évolution des systèmes EIAO les conduit à intégrer progressivement le souci d'efficacité et les objectifs de modélisation que les premiers logiciels d'enseignement avaient à coeur de défendre.

Dans la seconde partie de ce chapitre, nous décrirons brièvement les fondements de l'approche alternative à celle des systèmes orientés "modèles de l'enseignement" en illustrant notre propos par une description du système MEMOLAB que nous développons actuellement à Genève.

Les systèmes "orientés théorie de l'apprentissage"

Si l'on s'intéresse à l'éducation et à l'enseignement, et qu'on a l'ambition de doter un système artificiel de compétences dans ce domaine, il n'est pas possible d'ignorer tout ce qui touche à l'apprentissage, la mémoire ou la cognition. Une théorie de l'instruction, pour reprendre une terminologie de Gagné (1987), doit pouvoir définir avec précision les variables et les conditions qui 1) améliorent la prise d'information et l'acquisition de connaissances et 2) favorisent la rétention et la recherche de ces informations en mémoire. A priori, une telle théorie, à condition qu'elle existe, est difficilement applicable par un enseignant dans son cours, car ses capacités de traitement et la quantité de tâches annexes qu'il doit accomplir rendent peu probable le recours à de telles méthodes. Mais il ne paraît pas impossible de penser qu'un ordinateur, dont le talent pour traiter des paramètres n'est plus à démontrer, ne puisse pas effectuer en temps réel les calculs nécessaires à la gestion d'un modèle formel complexe.

Pour la construction de tels systèmes, Park et Tennysson (1987) proposent de distinguer deux niveaux de représentation pour la manipulation des variables affectant l'instruction. Le premier niveau baptisé "MACRO Level" se réfère au curriculum et concerne principalement la mémoire et la cognition; il gère globalement l'environnement et les conditions dans lesquelles un élève travaille (style cognitif, résultat à des tests de niveaux, connaissances). Le second niveau dénommé "MICRO Level" est relié aux événements associés directement avec l'apprentissage; c'est là que sont pris en compte au fur et à mesure les besoins de l'apprenant (temps d'exposition, quantité d'information fournies, format de représentation).

Cette distinction, bien que discutable sur le choix de certains paramètres annoncés, a le mérite d'ancrer la prise d'information et les processus de décisions sur le fonctionnement de l'apprenant plus que sur des modèles ad hoc de l'enseignement. Nous avons adopté pour MEMOLAB une approche similaire sur le fond, mais quelque peu différente sur la forme. Au lieu de nous intéresser à des paramètres tirés isolément de leur contexte, nous avons choisi d'implémenter un modèle qui prend en compte simultanément les contraintes du découpage d'un domaine de connaissances et celles du fonctionnement du sujet.

MEMOLAB: objectifs et description du système.

L'objectif pédagogique de MEMOLAB est l'apprentissage des connaissances impliquées dans la démarche expérimentale en psychologie (et plus généralement en Sciences Humaines). Ces connaissances sont mises en oeuvre dans un laboratoire artificiel qui permet la simulation d'expériences sur la mémoire humaine. Globalement, l'activité d'un sujet sur l'environnement est cyclique:

- il commence par des questions de recherche et la formulation d'hypothèses qu'il peut se procurer dans un hypertexte intitulé "Encyclopaedia of Memory" (ce dernier est accessible à tout moment à partir du Lab ou peut être activé par le tuteur).
- le sujet conçoit le plan d'expérience qui permet de répondre à la question posée. Pour cela, il construit un ou des groupes de sujets et il définit les étapes du traitement expérimental ainsi que le matériel à mémoriser. Pour fixer les idées, voici un exemple d'expérience que le sujet peut réaliser:

> Présenter une liste de 10 mots à deux échantillons de 20 sujets. Pour le groupe 1, les mots sont choisis comme appartenant au même champ sémantique. Pour le groupe 2, les mots n'ont aucune relation sémantique. Les sujets des deux

groupes étudient la liste pendant 5 minutes. La performance mesurée sera le nombre de mots correctement rappelés.

- le système réalise l'expérience de l'élève. Il génère les résultats en comparant l'expérience conçue par l'élève à une librairie d'expériences trouvées dans la littérature. La sélection et l'adaptation des résultats de la littérature sont réalisées au moyen de techniques de type "case-based reasoning" (Riesbeck et Schank, 1989).
- le dernier stade de ce cycle est l'analyse des données recueillies pour élaborer une conclusion (temporaire) relative à la validité des hypothèses initiales.

Concrètement, construire une expérience dans MEMOLAB, consiste à assembler, sous la forme d'une séquence, des événements discrets sur un "plan de travail". Un événement, c'est l'association de quatre composantes: un groupe de sujets, un matériel à mémoriser et une tâche. Construire un plan d'expérience, c'est créer une séquence non instanciée sur le "plan de travail" et l'instancier par la définition d'une table de facteurs à n-dimensions (n étant le nombre de facteurs indépendants manipulés dans l'expérience).

Il est évident que l'ensemble des compétences impliquées dans la maîtrise de l'expertise que nous venons de décrire ne peut pas faire l'objet d'une acquisition en bloc dès les premières manipulations (Beauvois, Roulin et Tiberghien, 1990). L'intégration des connaissances de plus en plus complexes dans un domaine quelconque suppose que l'environnement d'apprentissage propose une hiérarchie en niveaux homogènes et cohérents que l'apprenant peut maîtriser progressivement. Cette "simplification didactique" peut être réalisée a priori, arbitrairement, suivant un plan issu d'observations empiriques ou bien en référence à un modèle cognitif de l'intégration des connaissances. C'est cette dernière voie que nous avons choisie. Le modèle qui nous paraît le mieux adapté est celui que Case (1985) a formulé dans le cadre des recherches dites "néo-piagétiennes".

Le modèle de l'intégration hiérarchique de R. Case

Pour Case, la notion de base de l'analyse du fonctionnement intellectuel et de son développement en niveaux (ou stades) est la structure de contrôle (executive control structure). Quel que soit le domaine de connaissance, Case analyse la conduite du sujet en la considérant comme l'exécution d'un plan interne (mental plan) de résolution de problème. Ce plan peut être défini comme une séquence (un programme) composée de schèmes figuratifs et/ou opératifs (les schèmes figuratifs représentant des états et les schèmes opératifs des transformations). L'organisation d'ensemble du plan se fait suivant trois composantes:

1. la représentation de la situation-problème: c'est l'ensemble des conditions pour lesquelles le plan d'action est pertinent. Cette représentation peut être plus ou moins complexe, cette complexité étant déterminée par le nombre et la richesse des propriétés de la tâche à résoudre.
2. la représentation des objectifs à atteindre: c'est l'ensemble des nouveaux états vers lesquels le plan d'ensemble est orienté (dans notre environnement ce sont des challenges proposés à l'élève).
3. la représentation de la stratégie à mettre en oeuvre: c'est la séquence des opérations qui permet de passer de la situation-problème à la situation-objectif.

Chacune de ces composantes comprend plusieurs étapes: l'articulation de chaque élément de la représentation de la situation-problème avec son correspondant sur les objectifs; ensuite, l'articulation de chaque objectif avec l'étape suivante de la situation problème; enfin, la relation de ces deux types d'articulation avec la stratégie qui en découle. Cet ensemble constitue une unité qui permet de parler de structure et la notation utilisée par Case permet

d'avoir une vue d'ensemble du processus de résolution de problème associé à une tâche bien définie. L'analyse qui permet de construire ces structures se fait d'un point de vue développemental en procédant domaine par domaine.
Une des caractéristiques du modèle de Case est d'articuler les changements quantitatifs à l'intérieur d'un stade (augmentation du nombre d'unités activables en mémoire de travail) avec les changements qualitatifs entre les stades du développement.

Ce qui distingue les stades du développement des sous-stades c'est la nature des éléments que traitent les structures de contrôle: Case les appelle "les unités de base de la pensée". Dans le développement (et probablement aussi dans l'acquisition d'expertise), on retrouve ainsi la succession classique en quatre niveaux:
Stade 1: objets perçus et activités motrices
Stade 2: relations entre activités motrices
Stade 3: manipulation de dimensions (variables quantifiables)
Stade 4: dimensions de second ordre (ratios)

La question de savoir comment se forment les nouvelles unités et comment un sujet change de stade conduit de la description du développement à son explication. Chaque nouveau sous-stade d'un même stade se caractérise par la coordination d'une nouvelle unité de base à l'intérieur de la structure de contrôle. On passe ainsi de deux unités coordonnées au premier sous-stade à quatre unités au troisième sous-stade. La coordination des unités de base au dernier sous-stade atteint une complexité telle qu'elle correspond à une unité de base du stade supérieur. Celui-ci commence à dérouler ses sous-stades quand la structure de contrôle coordonne deux de ses nouvelles unités de base et ainsi de suite. Le dernier sous-stade peut être considéré comme le dernier sous-stade du nouveau stade suivant. En d'autres termes, la structure de contrôle d'un stade à quatre unités (mesuré par le nombre d'éléments qui peuvent être maintenus en mémoire de travail) peut être traduite et formulée en une structure de contrôle à une unité du stade suivant. Il s'agit là d'un processus d'abstraction. Le nouveau stade consiste en l'intégration dans une seule structure de contrôle de deux structures de contrôle formées pendant le stade précédent et dont la fonction était différente à cet ancien stade. C'est le processus de l'intégration hiérarchique. Au nouveau stade, l'une de ces anciennes structures est subordonnée à l'autre, en général comme moyen pour atteindre un but, alors qu'auparavant les deux structures fonctionnaient séparément. Une des modifications les plus spectaculaires du passage d'un niveau de compétence donné à un niveau supérieur réside dans le fait que les structures de contrôle sont au départ indépendantes les unes des autres et deviennent ensuite intégrées de façon hiérarchisée.

Le phénomène qui permet le passage d'un sous-stade au suivant, c'est l'augmentation de la capacité de stockage à court terme de l'information (STSS pour Short Term Storage Space). Cette capacité se développe à l'intérieur de l'espace total de traitement (TPS pour Total Processing Space) qui comprend aussi une capacité opératoire dédiée au contrôle du schème en cours (OS pour Operating Space). STSS augmente d'une part avec l'âge par des processus liés à la maturation du système nerveux et, d'autre part, avec l'entraînement des schèmes grâce à un gain en efficacité opérationnelle des représentations et opérations impliquées dans la structure de contrôle. Ainsi, au fur et à mesure que le sujet s'exerce, il libère de l'espace supplémentaire en mémoire à court terme pour stocker de nouveaux objectifs. Une conséquence incidente de cet aspect du modèle est qu'un environnement d'apprentissage doit favoriser cet entraînement par des exercices (sur le modèle des didacticiels classiques) au service de l'économie générale du système cognitif du sujet.

L'intégration hiérarchique de deux structures de contrôle est donc soumise à une augmentation de l'efficacité opérationnelle des schèmes qui les composent. Cette efficacité dépend étroitement du taux d'activité du sujet qui doit passer par plusieurs étapes au cours des phases d'apprentissage:
1. *Activation*: Un ensemble existant de structures de contrôle doit être activé dans une nouvelle séquence d'actions.
2. *Evaluation*: les conséquences et/ou l'utilité fonctionnelle d'une telle séquence doivent être évaluées.
3. *Etiquetage*: les points pertinents et saillants du déroulement de la séquence doivent être recodés de manière à ce qu'ils puissent être activés intentionnellement dans le futur.
4. *Consolidation*: cette nouvelle entité recodée doit être exercée afin de pouvoir être consolidée comme une nouvelle entité de traitement.

La métaphore de la pyramide et le "language shift"

Dans la conception du développement élaboré par Case, l'accent est mis sur le fait que le sujet est avant tout un "résolveur de problèmes" plus qu'un sujet qui s'approprie des "outils" comme dans les conceptions de Bruner ou de Vygotsky ou plus qu'un "scientifique" comme le perçoit implicitement Piaget. Dans cette perspective, l'enseignement doit donc faciliter le développement des capacités de résoudre les problèmes qu'un apprenant rencontre dans un environnement donné. A cette fin, un environnement d'apprentissage doit prendre en compte:

- la spécification des objectifs poursuivis par l'instruction.
- l'analyse fine des structures de contrôle de l'expert.
- l'analyse des précurseurs de ces structures de contrôle.
- la construction d'activités intégrant cette analyse.
- la mise au point de programmes, par domaines de compétences, prenant en compte ces activités dans leur globalité.

Ces principes déterminent en quelque sorte une méthodologie pour la conception d'un environnement d'apprentissage. Le système permet donc de tester à la fois un modèle du développement cognitif appliqué à l'acquisition d'une expertise complexe (le raisonnement expérimental) et une méthodologie de développement d'un logiciel. Il nous reste maintenant à construire les passerelles indispensables pour faire se rejoindre le modèle cognitif et les spécifications nécessaires à son implémentation. Pour cela nous avons utilisé une représentation intermédiaire qui joue le rôle d'"objet pour penser" le problème en termes informatiques.

La pyramide est une métaphore qui représente la structure de notre environnement: les concepts et les savoir-faire que l'apprenant doit maîtriser sont ordonnés de haut en bas en fonction de leur niveau d'abstraction et du type d'objets soumis au contrôle des sujets. Apprendre, sur ce modèle, consiste à s'élever sur l'axe vertical de la pyramide en franchissant successivement les différents niveaux. Le rôle du tuteur est de faciliter cette "élévation" en prenant en charge (temporairement) le contrôle des objectifs les plus élevés. Chaque niveau qualitatif est défini par un langage de commande, un langage de description et une classe de problèmes. Le vocabulaire du langage de commande représente l'ensemble des actions élémentaires qu'un sujet peut contrôler à un niveau quelconque de son "développement". La syntaxe du langage détermine de quelle manière le sujet peut composer les séquences élémentaires d'actions. Dans le modèle de Case, cette syntaxe correspond aux relations entre

les schèmes. Ces relations définissent l'organisation des schèmes au sein de la structure de contrôle. Le langage de description est défini comme l'ensemble des symboles et des représentations utilisées par le tuteur pour réfléchir, dans les actions du sujet, ce qui lui paraît essentiel du point de vue de la construction du plan d'expérience. Cette description a comme fonction de "réifier" les éléments abstraits de sa conduite dans le but de les rendre explicites et disponibles pour une "réflexion" ultérieure. Le sujet pourra ainsi prendre en charge une partie du contrôle qui lui échappait du fait de la surcharge cognitive provoquée par la complexité de la tâche.

Le langage de commande et le langage de description sont différents à chaque niveau de la pyramide. La hiérarchie intrinsèque que suppose cette métaphore implique que chaque niveau intègre les potentialités des niveaux qui lui sont inférieurs. Cette intégration est contenue dans les relations qui existent entre les langages utilisés à des niveaux succesifs de la pyramide: si un langage de description du niveau N est utilisé comme nouveau langage de commande au niveau N + 1, alors l'apprenti est obligé d'utiliser explicitement les concepts qui ont été "réifiés" au niveau N. Par exemple, le découpage proposé dans les connaissances enseignées dans MEMOLAB permet une intégration des savoir-faire depuis les plus élémentaires: évaluer l'effet d'une source de variation (niveau 1), jusqu'à celles qui permettent à l'apprenti-chercheur de "penser" son expérience sur un plan multidimensionnel (niveau 4). Nous avons appelé ce mécanisme la substitution de langage ("language shift" Dillenbourg, à paraître). On peut considérer ce processus comme un analogue du processus d'intégration hiérarchique du modèle de Case tel que nous l'avons décrit plus haut. Seule différence, il se présente maintenant sous une forme opérationnalisable dans un système informatique. Le langage de commande représente les schèmes "compilés" dans le programme, le langage de description, les effets spécifiques de ces mêmes schèmes sur la structure de contrôle. L'entraînement de ces schèmes associés en une nouvelle séquence par le cycle évaluation, étiquetage et automatisation permettra au sujet d'utiliser le langage de description fourni comme un nouveau langage de commande. Cette phase correspond dans le modèle de Case à la nécessaire automatisation des structures de contrôle (cycle: activation, évaluation, étiquetage et consolidation) qui permettent de libérer suffisamment d'espace en mémoire de travail pour intégrer de nouveaux objectifs.

La pyramide de MEMOLAB possède trois niveaux construits autour de la complexité croissante des plans d'expérience. D'autres connaissances indispensables, en statistique et sur le contenu des expériences, occupent dans notre modèle une place particulière. Leur maîtrise se trouvent subordonnée à la construction des plans, de la même manière que Case suppose que toute intégration de deux structures de contrôle est subordonnée à l'une d'entre elle.

Chaque niveau est donc défini par:
- une classe de problèmes de complexité analogue du point de vue de la structure du plan d'expérience (challenge).
- des objets manipulables, soumis au contrôle du sujet pour une élaboration qui le conduit à une solution (langage de commande).
- un langage de description utilisé par le tuteur pour "réfléchir" la conduite du sujet. Cette description est focalisée sur la structure du plan d'expérience.
- un ensemble de concepts et de savoir-faire subordonnés à la complexité croissante des plans d'expérience.

La structure d'un environnement d'apprentissage peut ainsi se résumer comme une séquence de quadruplets (classe de problèmes, langage d'action, langage de représentation et exercices),

séquence dans laquelle la relation entre deux ensembles est décrite par le mécanisme du "language shift".

Suivant ce modèle, il est possible de décrire maintenant les trois niveaux de la pyramide de MEMOLAB:

Niveau 0. Prérequis: utilisation de l'interface.
L'objectif est ici d'apprendre à construire un évenement et à le positionner sur le plan de travail pour le rendre actif. Chaque événement est composé de quatre éléments (les sujets, le matériel, la tâche et la mesure associée à cette tâche). La construction de cet objet-événement est supposée acquise au début d'un cycle d'apprentissage (il commence réellement au niveau 1 à construire des expériences). Le sujet doit réaliser que les événements intéressants sont ceux qui assurent une cohérence interne forte entre ses composantes. Par exemple, la tâche demandée doit être pertinente par rapport au matériel proposé. Cette étape vise à construire des familles d'événements type du genre: encodage, maintien, rappel....
La structure de contrôle secondaire subordonnée à cette acquisition concerne l'apprentissage des différentes valeurs que peuvent prendre les paramètres associés à chaque composante de l'événement: nombre de sujets, longueur des listes, valeurs affectant les mots qui constituent la liste (degré de familiarité, signification...)

Niveau 1. Des événements aux séquences: l'identification des sources de variation.
Les questions qui se posent à l'apprenant, à ce niveau, concernent l'identification des paradigmes pertinents pour l'étude de la mémoire. Les objets soumis au contrôle du sujet sont les événements, sa tâche consiste à les assembler pour former des séquences significatives. Les séquences qui ont certains traits communs appartiennent à une même famille: c'est ce que l'on appelle un paradigme. Par exemple, le paradigme du rappel libre se compose d'une phase d'encodage, suivie d'une période plus ou moins longue de maintien de l'information et enfin une phase de rappel sans contrainte d'ordre. Le rôle du système est de "réfléchir" une image des liens entre les événements par leur regroupement en séquences d'un certain type (le paradigme). Les outils de statistique descriptive (dénombrement, moyenne et variance) constituent la structure de contrôle subordonnée à la structure "paradigme".

Niveau 2. Des paradigmes à la comparaison de groupes sur un seul facteur:
A ce niveau, les objets contrôlés sont les événements, organisés en séquences, que le sujet a construits au niveau 1. Il peut donc directement invoquer et positionner sur le plan de travail un paradigme spécifique, par exemple le paradigme du rappel libre. L'apprenant instancie ensuite cette séquence en modifiant directement la valeur d'un paramètre à travers les différents événements qui composent une séquence. Le système propose quant à lui une représentation (le langage de description) qui met en évidence la nature comparative de l'expérience sur un facteur pouvant comprendre plusieurs modalités. Ce type d'expérience suppose que le sujet ne fasse varier qu'un seul paramètre à la fois pour valider les inférences qu'il pourra réaliser à l'aide des outils statistiques. Les outils statistiques associés à ce niveau permettent de réaliser des comparaisons de moyennes (en particulier le t de student).

Niveau 3. Du plan simple à l'interaction: A l'aube de ce niveau, le sujet maîtrise les comparaisons sur un facteur. Le système lui permet d'utiliser globalement ces comparaisons comme des objets instanciables. Les commandes sont celles qui permettent de réaliser cette instanciation à partir d'un objet "plan-à-deux-facteurs". L'apprenti-chercheur crée un plan d'expérience sur l'écran comme une structure de contrôle vide. La complexité des problèmes

posés suppose explicitement une analyse qui fait ressortir l'interaction entre les facteurs comme une source de variation à part entière (c.à.d. au même titre que les effets simples des facteurs). Les outils statistiques associés à ce niveau permettent des comparaisons de moyenne qui relativisent les variations inter-groupe par rapport aux variations intra-groupe (F de Snédecor, par exemple).

Le passage d'un niveau de la pyramide au niveau supérieur est lié à la substitution du langage de commande par le langage de description. Par exemple, le concept de séquence est utilisé au niveau 1 pour décrire aux apprentis leur propre travail (langage de description). L'association systématique d'une structure de comportements et de sa représentation graphique permet d'induire la signification de cette représentation. Au niveau 2, le concept de séquence et sa représentation graphique sont intégrés dans le nouveau langage de commande. De cette manière, nous contraignons l'apprenant à utiliser ce concept pour "penser" ses nouvelles expériences. Plusieurs auteurs ont insisté sur l'exploitation des réifications graphiques pour supporter la réflexion de l'apprenant. Par le dynamisme des modes d'interaction, nous souhaitons aller plus loin et contraindre le sujet à l'utiliser afin de l'exploiter comme moteur d'apprentissage.
Ce mécanisme est reproduit à chaque transition entre niveau. Les différences entre séquences sont - pour prendre un autre exemple - réifiées au niveau 2 et deviennent au niveau 3 la façon de créer un plan expérimental. A l'intérieur d'un niveau, la réalisation d'expériences de plus en plus complexes entraîne un accroissement sévère de la charge cognitive de l'apprenant. Comme le précise Case, le passage au niveau supérieur est rendu nécessaire par le caractère limité de notre mémoire de travail. Après la substitution de langage, l'apprenant dispose de nouvelles commandes qui prennent en charge la mise en oeuvre de séquence d'actions que l'apprenant devait préalablement gérer lui-même. Ceci libère les ressources cognitives nécessaires à l'élève pour se concentrer sur des aspects plus complexes de la tâche. Pour Case, cette réduction de la charge cognitive résulte d'une automatisation par l'apprenant de ses structures de contrôle. Sans sous-estimer ce phénomène, nous lui ajoutons l'automatisation informatisée comme un coup de pouce pour amener l'élève à prendre en compte des niveaux plus élevés d'abstraction.

Conclusion

La validité de la métaphore de la pyramide n'a pas à être prouvée formellement. Son intérêt réside dans sa valeur heuristique et nous avons montré qu'elle peut être utilisée comme interface entre plusieurs conceptions théoriques qui jusqu'à présent ne "parlaient" pas le même langage. Nous avons par ailleurs mis en correspondance la structure de la pyramide avec le concept de zone proximale de développement de Vygotsky (1978). Notre approche permet deux stratégies de recherche complémentaires: d'une part, nous enrichissons les recherches sur l'enseignement assisté par ordinateur d'une méthodologie de développement et d'une assise théorique qui lui faisaient souvent défaut dans les travaux issus de l'IA, d'autre part, nous proposons à la psychologie cognitive développementale un laboratoire pour éprouver ses modèles en testant leur faisabilité. Autre avantage de cette approche, un environnement intelligent d'apprentissage comme MEMOLAB prend en compte la richesse et la complexité des processus cognitifs impliqués dans l'enseignement d'une discipline. Par exemple, l'importance des mécanismes de découverte, le rôle de l'entraînement, le raisonnement par analogie et les effets du guidage de l'action sont intégrés à un environnement comme MEMOLAB, comme ils le sont dans le monde réel. Outre le micro-monde (laboratoire artificiel) et l'hypertexte, MEMOLAB comporte plusieurs composantes entre lesquelles les interventions pédagogiques ont été distribuées. Trop longtemps, la recherche en EAO a isolé

des processus d'acquisition dont la fonction complémentaire était sous-estimée. Cette ultime remarque nous amène à constater qu'une des définitions les plus classiques de l'intelligence s'appliquerait alors pleinement à notre environnement: celle qui met en avant la richesse des capacités d'adaptation d'un système à son environnement.

Références Bibliographiques

Anderson, J.R., and Reiser, B. J. - (1985) The LISP tutor, *Byte*, vol.10, no 4, 159-175.
Beauvois, J.L., Roulin, J.L., et Tiberghien, G. - (1990) *Manuel d'études pratiques de psychologie*, Paris: PUF.
Bloch, G., and Farrel, R. - (1988) Promoting creativity through argumentation, *Proceedings of ITS-88*, Montreal, 243-249.
Brown, J.S. and Burton, R.R. - (1978) Diagnostic models for procedural bugs in basis mathematical skills, *Cognitive Science*, 2, 155-191.
Brown, J.S., - (1983) Learning by doing revisited for electronic learning environments. In White , M.A. (Ed) *The future of electronic learning*, Hillsdale, New Jersey: Lawrence Erlbaum Associates.
Brown, J.S., Burton, R.R., and Bell, A.G. - (1975) SOPHIE: a step towards a reactive learning environment, *International Journal of Man-Machine Studies*, 7, 675-696.
Burton, R.R., and Brown, J.S. - (1976) A tutoring and student modeling paradigm for gaming environments. In Colman, R. and Lorton, P. Jr. (Eds) *Computer science and education*, ACM SIGCSE Bulletin, 8, 1, 236-246.
Carbonell, J.R. - (1970) AI in CAI: an artificial intelligence approach to computer-assisted instruction, *IEEE Transactions on Man-Machine Systems,* 11, 4, 190-202.
Case, R. - (1985) *Intellectual development: Birth to adulthood*, NY: Academic Press.
Clancey W. - (1979) Tutoring rules for guiding a case method dialogue, *International Journal of Man-Machine studies*, 11, 25-49.
Clancey, J.R. - (1983) GUIDON. *Journal of computer-based instruction,* 10, 1, 8-14.
Collins, A., and Brown, J.S. - (1987) The computer as tools for learning through reflection. In Mandl, H., and Lesgold, A.M. (Eds) *Learning Issues for Intelligent Tutoring Systems.* New York: Springer-Verlag.
Dillenbourg, P. - (1989) Designing a self-improving tutor: PROTO-TEG, *Instructional Science* 18, 193-216.
Gagné, R.M. - (1987) *Instructional Technology: Foundations*, Hillsdale, New Jersey: Lawrence Erlbaum Associates.
Goldstein, I.P. - (1982) WUMPUS. In Barr, A. and Feigenbaum, E.A. (Eds) *Handbook of Artificial Intelligence* (vol II), Reading, Massachussets: Addison-Wesley.
Morisson, F. - (1975) Planning a large-scale computer-assisted instruction installation - the TICCIT experience. In O. Lecarme and R. Lewis (Eds), *Computers in Education.* Amsterdam: North-Holland.
Murphy, R.T., and Appel, L.R. - (1977) Evaluation of the PLATO IV computer-based education system in the community college, *National Science Foundation Contract* No NSF-C731, Princeton, NJ: Educational Testing Service.
Park, O. and Tennyson, R.D. - (1983) Computer-based instructional systems for adaptative education : a review, *Contemporary Education Review*, 2, 121-135.
Park, O., and Tennyson, R.D. - (1987) Artificial intelligence and computer-based learning. In Gagné, R.M. (Ed)*Instructional Technology: Foundations*, Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Riesbeck, C.R., et Schank, R.C. - (1989) *Inside Case-based Reasoning*, Hillsdale, New Jersey: Lawrence Erlbaum Associates.
Self, J. - (1988) Bypassing the intractable problem of student modeling, *Proceedings of Intelligent Tutoring Systems Conference*. Montreal, 18-24.
Sleeman, D.H., and Brown, J.S. (Eds) - (1982) *Intelligent Tutoring Systems.* New York: Academic Press.
Tennyson, R.D., Christensen, D.L., and Park, S. - (1984) The Minessota Adaptative Instructional System: An intelligent CBI system, *Journal of Computer-based Instruction*, 11, 2-13.
Vygotsky, L.S. - (1978) Mind in Society. The development of higher psychological processes, M. Cole, V. John-Steiner, S. Scribner and E. Souberman (Eds), Cambridge: Harvard University Press.
Wengers, E. - (1987) *Artificial Intelligence and tutoring systems*, Los Altos CA: Morgan Kaufmann.
White, B.Y., and Frederiksen, J.R. - (1985) QUEST: qualitative understanding of electrical system troubleshooting, *ACM SIGART Newsletter*, 93, 34-37.