

POSTERS

| | |
|---|-----|
| Présentation d'une plate-forme "Hypermédia et apprentissage coopératif" | 229 |
| HYPER-ELISE : une maquette pour analyser l'interaction didactique | 233 |
| Petite Typologie des "HyperTraceurs" | 239 |
| L'articulation hypermédia / tuteur intelligent : un hypermédia éducatif | 245 |
| Utilisations éducatives des hypermédiats en démonstration : la chute des corps | 249 |
| Pour une approche signifiante de l'apprentissage à partir d'une base de données multimédia..... | 255 |
| Antibiotiques et gestion informatisée de l'antibiogramme..... | 261 |
| Pour un dispositif éducatif informatisé d'aide à la lecture critique | 269 |

**PRESENTATION D'UNE PLATE-FORME
"HYPERMEDIA ET APPRENTISSAGE COOPERATIF"**

Danièle Clément, Claude Viéville, Pierre Vilers

Université de Lille 1
Laboratoire Trigone - CUEEP
Cité scientifique, Bâtiment C6
59655 Villeneuve d'Ascq cedex
vieville@frcitl81.bitnet

Nous avons conçu un dispositif collectif de formation connecté en réseau, afin de favoriser les échanges entre participants, fonctionnant avec HyperCard et auquel nous avons ajouté un logiciel permettant les coordinations de point de vue entre apprenants.

1. Le dispositif de collectivisation d'HyperCard

A côté du projet CoCoNut qui vise la création d'une plate-forme spécialisée dans l'apprentissage coopératif, nous avons souhaité mener des expériences dans différents lieux et mobilisant un équipement standard. Pour cela nous avons mis au point un dispositif d'apprentissage coopératif du "pauvre" dans la mesure où il ne résulte que de l'assemblage de logiciels courants.

1.1. Fonctionnalités nécessaires

Pour programmer en groupe autour d'HyperCard, c'est-à-dire bâtir ensemble une architecture de pile, nous avons mis en place un dispositif matériel et logiciel relativement simple. Dans la salle de cours, nous disposons de cinq Macintosh ; nous souhaitons que chaque utilisateur puisse accéder à un espace de travail partagé où il pourra coopérer avec les autres pour programmer ; il possède aussi son espace privé mais pourra rejoindre l'espace public en activant une fenêtre représentant l'écran d'un Macintosh dédié à la coopération. Sur cette machine, nous y faisons fonctionner HyperCard qui sert à élaborer la solution négociée par les membres du groupe. Chaque utilisateur peut ainsi continuer à tester des éléments, des parties d'une pile personnelle dans une fenêtre HyperCard de sa propre station, puis a la possibilité d'envoyer ceux-ci dans la machine dédiée à la coopération. Arrivés dans cet espace, les éléments de pile pourront être manipulés par tous les membres du groupe de travail. De même, chaque utilisateur pourra recopier ces éléments dans son espace privé.

1.2. Mise en oeuvre

Les cinq appareils sont connectés par le réseau local AppleTalk qui offre le moyen d'échanger des données entre les machines et de partager des périphériques tel qu'une imprimante. Nous y avons ajouté le logiciel Timbuktu [Farallon 89] qui permet d'accéder à une station distante et de la commander à partir de son propre clavier et de sa souris. La connexion offerte par ce logiciel est dissymétrique. Un poste doit jouer le rôle de l'hôte pour le second. Et ce dernier peut contrôler l'hôte à partir de son clavier et de sa souris. Sur l'ordinateur hôte, tout se passe comme s'il avait désormais deux claviers et deux souris. Dans notre salle de classe, nous avons donc un poste hôte qui supporte l'espace public. Sur ce poste nous y faisons fonctionner HyperCard et tous les autres postes de la classe activent Timbuktu et prennent le contrôle de la station hôte. Dès cet instant une fenêtre s'ouvre sur chaque écran et on peut y voir une copie conforme de l'écran du hôte. Tous les claviers et toutes les souris de la classe peuvent contrôler le poste hôte. Le travail collectif devra donc être très organisé par un des membres de ce groupe.

Il est demandé aux participants de respecter une discipline d'intervention. Quand un utilisateur ne pilote pas l'hôte, il sort sa souris de la fenêtre pour ne pas gêner. Sur l'écran, un utilisateur peut apercevoir deux pointeurs de souris : celui de sa propre souris et le télépointeur de la machine hôte qui est de couleur différente. La désignation à distance est donc possible et autorise la personne qui a la parole à indiquer, par exemple en l'entourant, l'objet dont il parle.

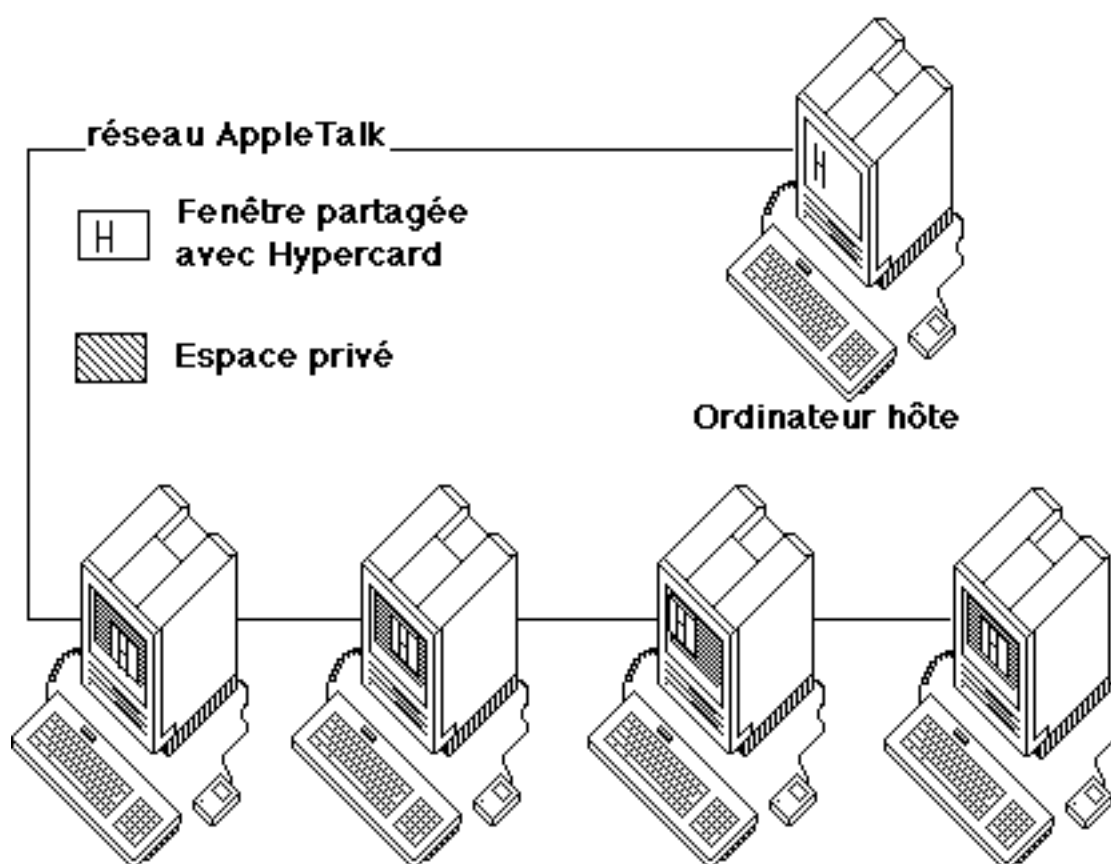


Figure 1 : Configuration utilisée

2. Les déficiences du dispositif actuel

Les déficiences de Timbuktu sont importantes. Entre autres, ce logiciel ne fait que rediriger les entrées sorties. Il ne permet pas de partager réellement les objets de l'espace public. Par exemple, il n'est pas possible de récupérer, par les opérations classiques, de copier/coller, les éléments qui se trouvent dans l'album de la machine hôte. De même les opérations de copier/coller entre les deux espaces ne sont pas possibles. En fait, la communication entre les deux espaces n'est possible qu'au travers d'un transfert de fichiers qui est heureusement une fonctionnalité offerte par Timbuktu.

Il n'existe aucun outil qui permette de gérer un travail collectif. Toute cette gestion est rejetée au niveau des utilisateurs qui pourront ainsi adopter la discipline de leur choix.

La fenêtre sur l'espace privé peut se dimensionner mais alors nous ne pouvons être certain que tous les utilisateurs voient la même chose. Par contre, Timbuktu visualise

toujours la zone où se trouve le télépointeur et la zone où se passent des modifications de l'écran.

Références

- (Derycke 90) A. Derycke, C. Vièville, P. Vilers : *Cooperation and communication in open learning : the CoCoNut project*, Proceeding of the WCCE 90, IFIP, Sydney 1990, pp 957-962
- (Derycke 91) A. Derycke : *Réseaux et apprentissage coopératif : vers l'Université Ouverte de 3ème génération*, Colloque Synapse, La Grande Motte, 30-31 Mai 1991
- (Ellis 91) C. A. Ellis, S. J. Gibbs, G. L. Rein : *Groupeware - some issues and experiences*. Communications of the ACM, January 91, Vol. 1, n° 1

**HYPER-ELISE :
UNE MAQUETTE POUR ANALYSER L'INTERACTION DIDACTIQUE**

Elisabeth Delozanne, Elisabeth Carrière

Université du Maine
Laboratoire d'Informatique
Route de Laval, BP 535
72017 Le Mans Cedex

Le projet ELISE vise à élaborer un environnement informatique à base de connaissances permettant à des étudiants de s'entraîner au calcul de primitives. Ce projet, fait suite aux travaux sur les systèmes CAMELIA (Vivet 84) et CAMELEON (Bruillard 91). Il s'appuie sur une analyse didactique du domaine qui propose d'enseigner des méthodes de résolution spécifiques au domaine pour faciliter l'apprentissage. L'article porte sur deux points d'une part la mise au point par une équipe d'informaticiens et de didacticiens des mathématiques d'une maquette sous HyperCard et, d'autre part l'observation et l'analyse du fonctionnement d'une trentaine d'étudiants pendant qu'ils travaillaient avec cette maquette (observations, trace des sessions, questionnaire, enregistrement audio, copies). Enfin nous présentons les leçons que nous avons tirées de ce test pour la construction d'ELISE, au niveau de l'ergonomie, au niveau de l'interaction didactique entre le système et les étudiants (son intérêt et ses limites).

1. Introduction

L'intégration d'un logiciel dans un réel contexte d'enseignement nécessite *dès la conception du logiciel* une étroite collaboration avec des praticiens et des théoriciens du domaine à enseigner, *et* des tests auprès des utilisateurs que sont les enseignants et les étudiants. Partant de cette hypothèse, nous avons réalisé, sous HyperCard, une maquette dont l'objectif est double. Premièrement, il s'agit de mettre au point, sous le regard des didacticiens¹, des interactions susceptibles de provoquer des apprentissages, puis tester ces interactions avec des étudiants, mais aussi auprès d'enseignants ; deuxièmement, il s'agit d'explicitier puis implémenter les connaissances mathématiques et pédagogiques nécessaires pour mettre en place ces interactions. Le travail de

¹ Quand dans cet article, nous parlons "des didacticiens", il s'agit de Marc Rogalski, professeur de Mathématiques à l'Université de Lille, mais aussi de quelques uns de ses collègues de l'IREM (Institut de Recherches sur l'Enseignement des Mathématiques) de Paris 7 (en particulier Aline Robert).

recherche que nous présentons ici, est donc du domaine de l'*étude préalable*, et nous permet de concevoir les spécifications du système ELISE.

L'objectif d'ELISE est de faire acquérir des savoir-faire à travers la résolution d'exercices, le système informatique fournissant des aides et des explications en cas de besoin. L'analyse des connaissances à acquérir est tirée d'un cours photocopié "Comment calculer une primitive ? Question de méthode..." (Rogalski 88) qui s'adresse à des étudiants de DEUG première année et se situe dans le cadre d'un *enseignement explicite de méthodes* (Robert 87, Schoenfeld 85). Par rapport à la résolution d'exercices, le logiciel a un double rôle de *correcteur* et de *donneur de leçons de méthode*. Le but des ses interventions est d'inciter les étudiants à anticiper et à développer des stratégies de contrôle de la résolution. Si les étudiants ont reçu un enseignement de méthode, il est question de les entraîner à la méthode, et, dans le cas contraire, de les aider à en élaborer une qui soit efficace.

Dans ce cadre, les *possibilités de navigation* offertes par HyperCard nous semblent intéressantes pour proposer aux étudiants *plusieurs solutions* sur chaque exercice, pour leur proposer *plusieurs modes de résolution, plusieurs niveaux d'explications*, et leur permettre de passer à leur gré de l'un à l'autre. C'est donc l'outil que nous avons retenu pour construire la maquette en vue d'étudier comment les étudiants circulent dans un tel logiciel, utilisent les diverses fonctionnalités, et dans la mesure du possible regarder la nature de leur activité lors de l'utilisation.

2. Présentation de la maquette Hyper-Elise

Deux *modes de résolution* sont proposés à l'étudiant :

- le *mode pas-à-pas* où l'étudiant choisit, dans un menu, les techniques ou transformations à appliquer et où le système se charge des calculs ; en cas d'erreur de la part de l'étudiant (choix inapplicable ou inintéressant), en retour, le système intervient pour la signaler (figure 1) ;
- le *mode plan* où l'étudiant doit donner les principales étapes de sa résolution, sous la forme de la liste des techniques à appliquer (figure 2) ; le système évalue le plan (bon, y'a de l'idée, je ne comprends pas).

Les aides et explications sont envisagées sous plusieurs formes :

- une *synthèse* des différentes résolutions "raisonnables" qui présente un arbre compacté des solutions (figure 4), et permet, de plus d'obtenir un commentaire général (objectif, démarche, justification) sur les tactiques mises en oeuvre dans les différentes solutions, d'accéder aux corrigés-types ; d'accéder au détail de la résolution pas-à-pas sur les étapes de toutes les solutions présentées ;
- des *commentaires* sur les choix effectués par les étudiants sous la forme d'une estimation sur le choix (choix optimal, raisonnable, peu prometteur, illégal) et d'une justification de ce choix ;

- des *coups de pouce* qui expliquent "comment démarrer ?" ou "comment continuer ?" au cas où l'étudiant serait bloqué; ce sont les réponses spécifiques au domaine, à quatre questions générales que l'on peut se poser face à un problème de ce type: quelles sont les caractéristiques du problème ? la méthode s'applique-t-elle ? quelles sont les techniques applicables ? à quel problème connu cela fait-il penser ?
- des *rappels de cours* sous forme Hypertexte ont été envisagés, mais non implémentés dans la maquette ; une aide en ligne rudimentaire est disponible sur chaque écran.

Problème : **Calculer** $F(x) = \int \frac{1}{2 + \operatorname{tg} x} dx$

*Considérons le **changement de variable** $t = \operatorname{tg} x$
d'où $dt = (1 + \operatorname{tg}^2 x) dx = (1 + t^2) dx$
On obtient $F(x) = \int 1 / (2 + t)(1 + t^2) dt$, avec $t = \operatorname{tg} x$
** Sous-problème: **Calculer** $G(t) = \int \frac{dt}{(2+t)(1+t^2)}$

Choisissez une technique

- Intégration par parties
- Changement de Variable
- Linéarité de l'intégrale
- Transformation
- Equation
- Identification
- Résultat Connue




Figure 1 : Résolution pas-à-pas, choix des techniques

Problème : Calculer $F(x) = \int (x^2 - 3x + 7) e^{-2x} dx$

Indiquez la technique suivante

- Intégration par parties
- Changement de Variable
- Linéarité de l'intégrale
- Transformation
- Equation
- Identification
- Résultat Connue
- Fin
- Annuler le choix précédent

Votre plan:

- 1) Transformation
- 2) Linéarité de l'intégrale
- 3) Intégration par parties
- 4) Intégration par parties




Figure 2 : Résolution en mode plan


Problème : Calculer $F(x) = \int \sin(\text{Log } x) dx$

Vous voulez:

- Etablir un plan de résolution
- Résoudre pas à pas
- Avoir une vue générale des solutions
- Vérifier votre résultat
- Voir un corrigé type

imprimer sommaire

memento



quitter aide logicielle première page vue générale coup de pouce

carte précédente

Figure 3 : Différentes fonctionnalités offertes

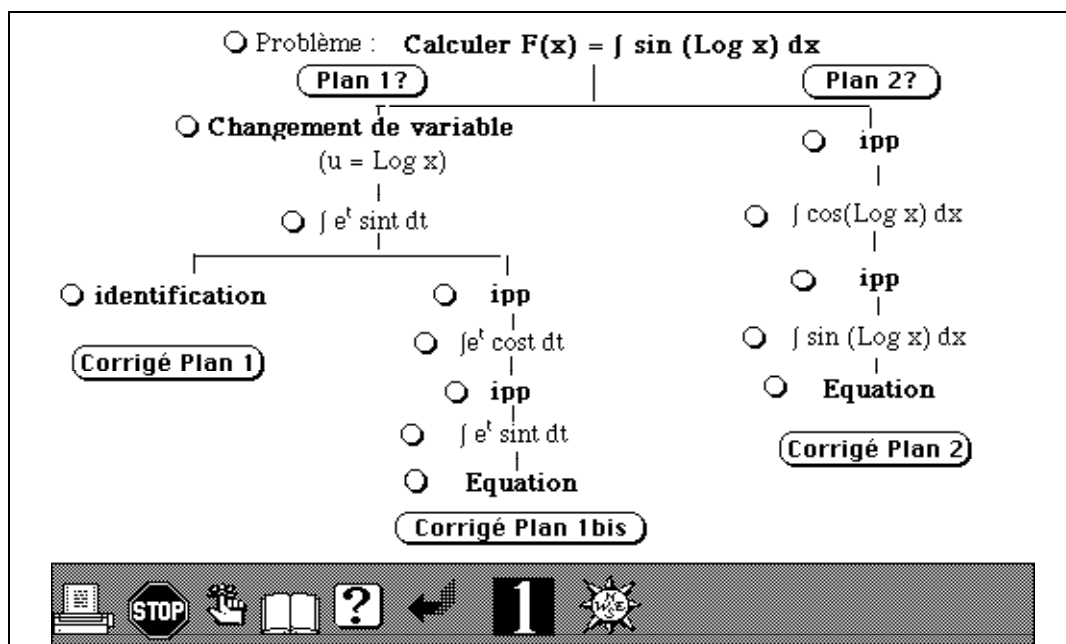


Figure 4 : Vue générale des solutions

3. Résultats des tests

Des séances de tests de la maquette ont été organisées avec deux groupes de quinze étudiants. Ils ont travaillé en binôme (par groupe de deux) pendant des séances de deux heures. Le dépouillement du mouchard, prouve une grande diversité d'utilisation selon les groupes, selon l'enseignement qui a précédé, et selon le contrat établi avec les étudiants.

A Lille, les étudiants ont préalablement suivi un enseignement de méthodes et pensaient faire une séance de travaux pratiques de mathématiques. La plupart des groupes ont commencé par établir un *plan* de résolution, et en cas d'échec l'ont essayé pas-à-pas ; trois groupes n'ont utilisé le plan que sur le premier exercice, puis n'ont utilisé que le mode de résolution pas-à-pas. Un seul groupe s'est contenté d'étudier une seule solution ; pour les autres groupes, ils sont passés par l'écran *vue générale* pour étudier au moins une autre solution. Un groupe a étudié systématiquement tous les commentaires et tous les corrigés à partir de l'écran *vue générale*. Enfin, cinq groupes ont accompagné le travail sur le logiciel par une *résolution sur papier* : deux groupes ont d'abord fait une résolution papier, utilisant le plan pour synthétiser leur raisonnement, un groupe a surtout recopié les écrans de corrigé, les autres groupes ont ponctuellement commencé un calcul, appliqué une formule ou pris des notes.

Au Mans, les étudiants n'avaient pas reçu d'enseignement de méthodes et ils savaient qu'ils participaient à une expérimentation et que l'on attendait leur avis sur le logiciel. Pour tester le logiciel, six des huit groupes ont exploré en cliquant plutôt pour voir que pour résoudre (conformément à leur contrat). Il est donc difficile de caractériser leur parcours. Sur les deux groupes restant, l'un a beaucoup *travaillé sur*

papier et passé beaucoup de temps sur les *plans* (40 minutes pour une heure-et-demi de travail), les utilisant pour *anticiper*, mais plus souvent pour *synthétiser une solution trouvée sur papier* ou présentée dans vue générale. L'autre groupe a passé beaucoup de temps sur les corrigés et n'a pas du tout apprécié le mode plan (l'observation prouve qu'ils manquaient de connaissances pour planifier).

Les questionnaires remplis par les étudiants à la fin de chacune des séances, montrent que les impressions sont généralement très favorables.

Les *points forts du logiciel* les plus souvent cités sont : sa *clarté*, la qualité des *explications* qu'il fournit ("claires", "efficaces", "méthodiques"), le caractère "*agréable*" du travail sur le logiciel (facile à manipuler, "rapide" - il fait les calculs -, "vivant", "interactivité", "le plaisir d'apprendre", "ludique", et même... "l'humour des commentaires"), la présentation de *plusieurs solutions* ; certains mentionnent aussi les *méthodes* et le fait de pouvoir aborder un exercice de plusieurs façons. Viennent ensuite : les *corrigés-type*, et la démarche ou les calculs *bien détaillés*.

Les *points faibles* signalés : le nombre trop faible d'exercices, les problèmes de vocabulaire et la lenteur du logiciel ; des étudiants signalent des problèmes de communication sans préciser pourquoi sauf un étudiant qui trouve que le logiciel "ne répond pas à toutes les questions que se posent les étudiants" ; cinq étudiants se plaignent que les résultats sont donnés trop vite et donnent un conseil : découper davantage les écrans de calcul pour laisser le temps de chercher.

Le décryptage d'une cassette audio où ont été enregistrées les réactions de deux groupes d'étudiants pendant qu'ils travaillaient sur la maquette prouvent qu'ils se sont pris au jeu (trouver ce qu'il faut cliquer, mais aussi trouver par le mode de résolution pas-à-pas ce qui n'allait pas dans un plan refusé par le système), que le jeu a donné lieu à de nombreuses discussions parfois animées, leur a donné l'occasion de revoir leur cours, de réfléchir et de faire quelques découvertes.

4. Les leçons à tirer

Au niveau logiciel ou ergonomie, des améliorations peuvent être apportées sans changements de spécifications.

Au niveau mathématique, des problèmes de vocabulaire peuvent être résolus par un memento sous-forme d'Hypertexte, mais d'autres problèmes de communication sont en grande partie incontournables, en particulier l'*hétérogénéité des choix* à faire qui peut gêner les étudiants. Ceci nous a été signalé par les ergonomes ; mais s'il est possible (et souhaitable) de remédier à l'hétérogénéité dans la *forme des choix*, ici c'est l'hétérogénéité de leur *contenu* qui est en cause, et le problème est plus difficile à régler, les moyens mathématiques pour résoudre des exercices de ce type étant de nature hétérogène. On peut réfléchir pour savoir s'il faut mettre en relief ce caractère hétérogène au lieu de le masquer comme le fait Elise. Le même problème se pose, du

point de vue de l'Intelligence Artificielle, pour le choix d'un formalisme pour l'implémentation de la base de connaissances.

Au *niveau didactique*, si les interactions semblent convenir aux étudiants qui ont reçu un enseignement sur la méthode (l'objectif d'Hyper-Elise d'entraîner à la méthode est semble-t-il atteint), elles doivent être différenciées pour les étudiants qui doivent élaborer une méthode à travers la résolution d'exercices.

Ces deux séances de tests ne prouvent pas qu'Elise a un effet bénéfique à long terme sur l'apprentissage, elles n'étaient pas conçues pour cela ; mais, ces deux séances montrent que, dans un contexte de travaux pratiques suivant un enseignement explicite de méthodes, Elise crée des situations où les étudiants sont incités à la réflexion, incités à anticiper, incités à discuter. D'autres tests seront nécessaires, pour étudier, si en modifiant ces situations, on peut élargir le contexte d'usage.

Références

- (Bruillard 91) E. Bruillard : *EIAO et Mathématiques: une vision hypertexte des environnements d'apprentissage*, Thèse de l'Université du Maine, 1991.
- (Robert et al 87) A. Robert, J. Rogalski, R. Samurçay : *Enseigner des méthodes, Cahier de didactique des mathématiques n° 38*, Irem, Université Paris 7, 1987.
- (Rogalski 88) M. Rogalski : *Comment calculer une primitive? Question de méthode*, Polycopié DEUG A1, Université des Sciences et Techniques de Lille, 1988.
- (Schoenfeld 85) A.H. Schoenfeld : *Mathematical Problem Solving*, Academic Press, 1985.
- (Vivet 84) M. Vivet : *Expertise mathématique et informatique : CAMELIA, un logiciel pour raisonner et calculer*, Thèse d'Etat, Université Paris 6, 1984.

PETITE TYPOLOGIE DES “HYPERTRACEURS”

Jean-Philippe Drouhard

Institut National de la Recherche Pédagogique

91, rue Gabriel Péri

92120 Montrouge

Nous présentons ici les premiers résultats d'une recherche en cours intitulée “Structuration des connaissances à l'aide des Hypermédias” (Cf. Beaufils). Cette recherche a entre autres pour buts de déterminer le rôle des hypermédias dans l'acquisition et la structuration des connaissances, et en particulier le rapport existant entre le parcours choisi par un élève dans un ensemble hypermédias, et le degré de connaissances acquises à la fin de ce parcours. Pour cela, il faut disposer d'un moyen de savoir ce que l'apprenant fait, et quand. Divers modes d'observation, plus ou moins indirects (caméra vidéo, observateur etc.) sont envisageables. Nous avons privilégié la voie de l'enregistrement sur l'ordinateur même des actions de l'apprenant, au moyen d'un programme défini par le terme générique d'“Hypertraceur”.

1. Les modes d'action

Avant de présenter une typologie des Hypertraceurs, nous allons tout d'abord préciser une distinction, fondamentale pour notre projet, entre les modes d'action de l'utilisateur de l'hyperbase¹.

L'utilisateur dispose d'une plus ou moins grande liberté au sein de l'hyperbase. Le logiciel HyperCard par exemple distingue cinq niveaux de liberté possibles². Quant à nous, nous distinguerons simplement entre deux modes, appelés “utilisateur” et “auteur”. Dans le mode “utilisateur”, l'utilisateur se déplace au sein d'une hyperbase déjà constituée, répond aux questions qui lui sont posées, fait des calculs, des dessins, met en œuvre des ressources extérieures (vidéodisque, autres logiciels), mais ne modifie pas la structure de l'hyperbase. Il n'ajoute ni n'enlève aucun lien ni aucun élément.

En mode “auteur” par contre, l'utilisateur modifie la structure de l'hyperbase. Il peut même la créer, en ce sens qu'il peut structurer par des liens un ensemble d'éléments existants, par exemple des pages de texte ou des images. Il est net que selon le mode choisi, les fonctionnalités d'un traceur différeront (tout autant que les caractéristiques de l'apprentissage associé d'ailleurs). Ceci dit, un certain nombre de fonctionnalités

¹ Le terme est tiré de Gaines & Vickers (Gaines 87).

² “Navigation”, “Texte”, “Dessin”, “Auteur” et “Programmation”.

des Hypertraceurs restent communes aux deux modes, et nous les exposons maintenant.

2. Les modules d'observation (Observateurs)

Nous allons considérer que les Hypertraceurs sont formés d'un certain nombre de modules, en liaison avec les fonctionnalités envisagées. Bien entendu, on ne retrouvera pas toujours ces modules aussi clairement dans les produits finis, mais ces distinctions faciliteront la présentation.

En premier lieu, nous allons distinguer l'Espionneur (module destiné à récupérer puis à mettre en forme l'ensemble des actions produites par l'utilisateur) du Représentateur (module de présentation, après coup, de la liste des actions répertoriées).

2.1. L'Espionneur

L'espionneur est formé lui-même de deux modules : le capteur, qui capte une trace de l'ensemble des actions produites par l'utilisateur, et le Rapporteur, qui établit un rapport, autrement dit qui met sous une forme symbolique uniforme (par exemple, des listes de références) les traces récupérées par le capteur. Par exemple, le début du compte-rendu d'une session apparaît dans •SPY• de la manière suivante :

```
••• Session numéro 4
20 :32 :32 openStack  Enfantillages :D-SPY-INRP :Piles-de-
démó :•Demo Graphe•
20 :32 :34 mouseUp 65,10 card button "1" of card id 2904 of
stack      "Enfantillages :D-SPY-INRP :Piles-de-démó :•Demo
Graphe•"
20 :32 :42 mouseUp 260,71 card button "7" of card id 2904 of
stack      "Enfantillages :D-SPY-INRP :Piles-de-démó :•Demo
Graphe•"
```

2.2. Les Représentateurs

Le lecteur réalise sans peine à quel point la lecture directe d'un rapport tel que celui qui précède serait fastidieuse (sans compter que rien ne nous est épargné, y compris les hésitations de l'utilisateur). Il faut donc mettre en forme ce "rapport", en faire une "représentation" (graphique ou non) aisément interprétable. C'est le rôle du Représentateur.

Il a été très vite établi qu'il n'était pas possible d'apporter sous une représentation unique l'ensemble des informations possibles, sous peine d'être totalement illisible. On peut donc faire une première distinction entre les représentations textuelles (les "Rédacteurs") et graphiques (les "Grapheurs").

2.3. Les Rédacteurs

Ce mot est une traduction approximative de “Browser”, mot anglais qu’on ne peut traduire que très malaisément : “parcoureur”, “feuilleter”, “fouineur”, voire “brouilleur” (!). En outre, ce mot (browser) désigne des applications aux fonctionnalités très différentes d’un logiciel à l’autre. Dans NoteCard (Foss 88), le Browser fournit une représentation graphique interactive de l’hyperbase, en ce sens qu’on y voit où on est, d’où l’on vient, et qu’on peut, en cliquant sur les icônes correspondantes, aller où l’on veut. Dans SmallTalk par contre, le Browser fournit une représentation structurée mais textuelle de la session en cours (l’ensemble des objets reliés à un objet donné, l’état de ces objets, etc). Nous avons décidé (parce qu’il fallait bien choisir) d’appeler “Rédacteur” une application de ce dernier type, indiquant donc de manière Hypertextuelle (et non Hypergraphique) les données pertinentes de la session dans l’hyperbase : liens empruntés, types de ces liens, ressources utilisées, etc. La présentation proprement dite peut se concevoir sous la forme d’empilements de fenêtres déroulantes.

3. Grapheurs arborescents ou à support convexe

3.1. Quelques définitions

D’un système à l’autre une grande confusion terminologique règne, et □ nous avons été amenés à préciser les termes utilisés.

Points d’ancrage : Irlér & Barbieri (1990) parlent de "link anchors" pour définir les éléments entre lesquels les liens sont établis dans l’hyperbase. On pourrait traduire ce terme par “points d’ancrage” des liens, parmi lesquels sont distingués leur origine et leur extrémité.

Noeuds : De même, Irlér & Barbieri réservent le terme de “nœud” (node) pour les documents qui contiennent plusieurs points d’ancrage, et remarquent que les piles d’HyperCard sont des suites ordonnées de nœuds. En particulier, en HyperCard, les points d’ancrage d’arrivée peuvent être les cartes toutes entières.

Lieux d’ancrage : Dans le vocabulaire d’Hypercard, un point d’ancrage s’appelle une “carte”¹. Il arrive fréquemment que plusieurs cartes défilent à l’écran sans que l’usager ait l’impression de se “déplacer” dans l’hyperbase. Il lui semble voir toujours la même carte, avec quelques modifications : apparition ou disparition d’un champ de texte, ou d’un dessin, ou encore animation d’un dessin. En fait, la substitution d’une carte à une autre, ayant le même “fond” est un des outils de base pour assurer l’interactivité sans programmation. Dans ce cas (succession de cartes ayant le même fond, en vue d’animation ou d’interactivité) on dira que l’ensemble des cartes

¹ Dans d’autres logiciels (LinkWay par exemple) les noms changent... mais le concept est le même.

successives constitue un “lieu” unique. Pour qu’une suite de points d’ancrage puisse être assimilée à un lieu d’ancrage, il est donc nécessaire (mais non suffisant) qu’ils s’enchaînent sans ramification.

Graphe : Un “graphe” (ou plan) est ici, par définition, un graphe orienté (au sens mathématique du terme). Les nœuds de ce graphe orienté correspondent aux lieux de l’hyperbase, et non à ses nœuds au sens d’Irlér & Barbieri (= cartes), sauf si lesdits lieux sont eux-mêmes réduits à des cartes uniques. Les branches représentent les liens (dits parfois “liens virtuels” pour les distinguer de ceux qui sont activés par l’usager au cours de la session).

Trace : Une trace est un ensemble de liens distingués au sein d’un plan : ceux activés par l’usager lors de son parcours.

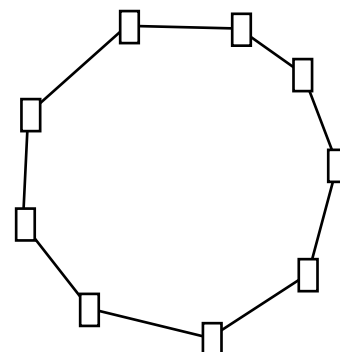
3.2. Grapheurs arborescents

Le Grapheur arborescent permet de construire le graphe (complet ou partiel) de l’hyperbase, en partant de n’importe quelle nœud, sous forme d’arbre. On peut le construire pas à pas, en l’“étendant” progressivement vers la droite à partir des icônes des nœuds qui nous intéressent. En cliquant sur un nœud quelconque, on fait apparaître vers la droite tous les chemins qui en partent et leurs destinations. Toutefois il est très difficile dans ce cas, pour des raisons d’encombrement d’écran, d’avoir une vision globale du graphe. Soit on le rétrécit pour qu’il tienne dans l’écran, et il devient illisible, soit on “déplace” l’écran (telle une fenêtre d’observation) par des bandes de défilement, mais une synthèse globale devient délicate.

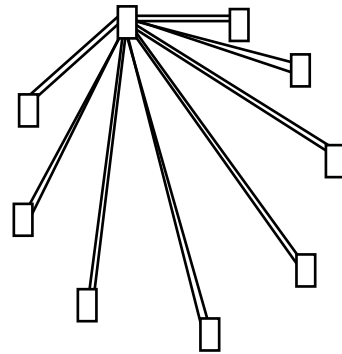
3.3. Grapheurs à support convexe

Pour remédier à ce type d’inconvénient nous avons élaboré un autre type de grapheur, représentant de manière symbolique la totalité de l’hyperbase : le Grapheur à support convexe. Celui-ci dispose les icônes des lieux (d’autant plus réduites qu’elles sont plus nombreuses) au bord d’une figure convexe (le “support”), en pratique un cercle. De cette manière, les liens forment des “cordes” de ce cercle et ne cachent pas les nœuds. Une telle représentation permet de voir immédiatement la structure de l’hyperbase.

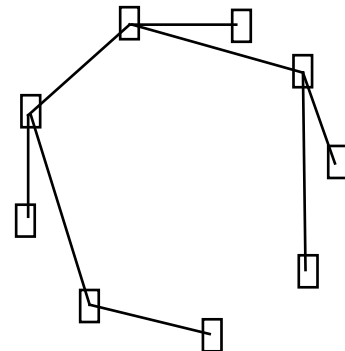
Une hyperbase “linéaire” en effet, où chaque lieu renvoie au suivant (suivant une logique de dessin animé par exemple) verra ses liens dessiner un polygone à support convexe reliant, de lieu en lieu, le premier lieu au dernier :



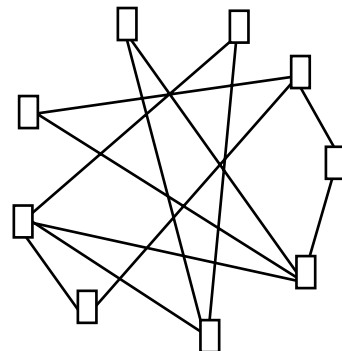
Une hyperbase “rayonnante”, où l’on passe de lieu en lieu en revenant à chaque fois à un “lieu central” (le sommet de l’arborescence) verra ses liens converger vers ce lieu :



Une hyperbase “arborescente”, généralisation de l’hyperbase “rayonnante”, dont la structure est celle d’un arbre, adoptera une représentation en “couronne de lauriers”:



Enfin, dans une hyperbase formant un réseau maillé (pleinement “hyper”) on verra les liens s’entrecroiser :



Ainsi, d’un coup d’oeil, on pourra déterminer la structure globale de l’hyperbase, les quatre types ci-dessus présentant bien entendu des cas extrêmes entre lesquels de nombreux cas intermédiaires peuvent être trouvés. Un tel grapheur présente des avantages et des inconvénients opposés à ceux du grapheur arborescent : dans un cas, (le grapheur convexe) on a une bonne vision de la structure globale de l’hyperbase, mais à cause de la petitesse des icônes représentant les lieux, on ne peut guère savoir ce qui les caractérise (sauf en cliquant dessus pour avoir des informations). On n’a donc guère d’information sur les aspects locaux de l’hyperbase. Inversement, le grapheur arborescent donne une vision exhaustive des portions de l’hyperbase qui sont à l’écran, au détriment de la structure globale. Ces deux modes de représentation sont donc complémentaires.

4. Animation

Les représentateurs dont nous venons de voir les principales fonctionnalités donnent une image statique de l'état de l'hyperbase. On peut également montrer l'évolution de l'hyperbase, par une version "dynamique" du Rédacteur ou de l'Hypergrapheur (où les représentations des objets et des liens créés apparaissent au fur et à mesure de leur ordre de création), ou encore en visionnant la suite des écrans ayant défilé sous les yeux des créateurs et/ou utilisateurs.

4.1. Le Rédacteur dynamique

Le Rédacteur dynamique est simplement une version "dynamique" du Rédacteur dans lequel les diverses fenêtres apportant des informations sur les objets et les liens créés apparaissent au fur et à mesure de leur ordre de création (sans que les intervalles temporels soient forcément respectés dans leur durée : on ne va pas attendre 40 minutes l'apparition d'une information sur un objet créé à la fin de cette durée).

4.2. L'HyperGrapheur dynamique

Comme le Rédacteur, le grapheur dynamique est une version du grapheur dans lequel images de lieux et de liens apparaissent dans l'ordre de leur création.

4.3. Visionneur

On peut prendre un parti différent et faire visionner, non les étapes successives des représentateurs au fur et à mesure de la constitution de l'hyperbase, mais plus simplement la suite des écrans ayant défilé sous les yeux des créateurs et/ou utilisateurs. Une première version d'Espionneur associé à un Visionneur a été réalisée au sein de l'équipe par D. Bertin : •SPY• version Ø.5.

Références

- Collazzo L., Mich L., Molinari A. & Castelli C. (1990) : *Measuring User Disorientation between Hypertext Structure and Cognitive Structure*, Rapport Interne -90 INF 6-, Università di Trento, Istituto di Informatica, Trento (Italie).
- De Las Heras A. (1990) : *Un système d'écriture pour l'Hypertexte*, Actes du Colloque "Communication Interactive '90 : Instruments de Communication Evolués Hypertextes, Hypermédiats", Balpe J-P. & Laufer R. (Eds.), Groupe §, Université Paris 8 et Journal de la Formation Cont. et de l'EAO, Paris.
- Foss Carolyn L. (1988) : *Effective Browsing in Hypertext Systems*, Proceedings of User-Oriented content-based text and image handling conference (RIAO 88), M.I.T., Cambridge (Ma).
- Gaines Brian R. & Vickers Joan N. (1988) : *Hypermedia Design*", Proceedings of User-Oriented content-based text and image handling conference (RIAO 88), M.I.T., Cambridge (Ma).
- Hofmann M. & Langendörfer H. (1990) : *"Browsing as Incremental Access of Information in the Hypertext System Concorde"*, Actes du Colloque "Communication Interactive '90 : Instruments de Communication Evolués

Hypertextes, Hypermédiats”, Balpe J-P. & Laufer R. (Dir.), Université Paris 8 et Journal de la Formation Continue et de l’EAO, Paris.

Irler W.J. & Barbieri G. (1990) : *Non-Intrusive Hypertext Anchors and Individual Colour Markings*, Hypertext : Concepts, Systems and Applications, Proceedings of the First European Conference on Hypertext, Sreitz & André (Eds.), Cambridge University Press.

Stotts D. & Furuta R. (1990) : *Hierarchy, Composition, Scripting Languages and Translators for Structured Hypertext*, Hypertext : Concepts, Systems and Applications, Proceedings of the First European Conference on Hypertext, Sreitz, Rizk & André (Eds.), Cambridge University Press.

L'ARTICULATION HYPERMEDIA/TUTEUR INTELLIGENT : UN HYPERMEDIA EDUCATIF

Christian Ritter, Djahanguir Djamei¹

BULL service A6
7, rue Ampère
91343 Massy Palaiseau

Nous présentons l'intégration, dans un même environnement d'apprentissage, d'un outil hypermédia (InterMed) et d'un outil de type Tuteur Intelligent (StarGuide). Nous montrons l'intérêt de cette coopération en termes de stratégies pédagogiques mais également sur le plan de l'adaptation et du guidage.

1. Les hypermédiats : des outils d'apprentissage

Dans certains contextes d'utilisation, les hypermédiats, de la même façon que les simulateurs, peuvent être considérés comme des environnements d'apprentissage au sens de S.Papert (Papert 81). La stratégie pédagogique mise en oeuvre par ce type d'environnement est axée sur la découverte. L'apprenant peut naviguer à l'intérieur d'un hyperdocument préalablement construit par un "expert" ou mieux réaliser sa propre application hypermédia. Dans les deux cas d'utilisations, l'apprenant pourra non seulement acquérir des connaissances du domaine mais également des méthodes de structuration de ces connaissances. Les expériences décrites dans (Beaufils 91) sur l'usage des hypermédiats dans des classes de collège tentent à prouver qu'il existe un lien entre des utilisations éducatives de certains logiciels informatiques et l'acquisition par les élèves de méthodologies de traitement de l'information.

2. Approche constructiviste et approche instructiviste

Les hypermédiats proposent une logique de construction des connaissances par opposition à une logique de transmission des connaissances sous-jacente à l'approche des Tuteurs Intelligents (Cf. (Nicaud 88) pour une description précise des TI). Or, comme le souligne Vivet (Vivet 91), "il y a lieu d'articuler des logiques de tutorat au sein d'environnements d'apprentissage (micro-mondes au sens de Papert). Cette articulation est nécessaire car la logique de transmission de connaissances utilisée

¹ C. Ritter et D. Djamei préparent des thèses sous les directions respectives de M. Vivet et C. Fluhr. Les travaux de recherche s'effectuent dans le service A6 dirigé par G. Claës.

seule ne fonctionne pas, pas plus d'ailleurs que la logique de construction personnelle de connaissances pour un élève largué seul dans un micro-monde."

3. Apport de l'intégration d'un hypermédia et d'un tuteur intelligent

Les approches "constructivistes" et "instructivistes" sont complémentaires. Mais la complémentarité des approches hypermédiat et des approches Tuteur Intelligent (TI) ne s'arrête pas aux stratégies pédagogiques qu'elles proposent. Les recherches sur les TI apportent :

3.1. L'adaptation

Dans un contexte d'auto-apprentissage, l'adaptation permet plusieurs utilisations d'un même environnement d'apprentissage et, utilisée à bon escient, permet de supprimer certains bruits informatiques qui parasitent l'acquisition des connaissances que l'on cherche à "faire apprendre". LOUTI (Paquette 91) permet de construire des progiciels qui sont utilisés comme outils d'apprentissage. Un progiciel ainsi construit est adapté au domaine traité.

Dans (Daniel 90), le concept d'adaptation des systèmes hypermédiat est divisé en deux, l'adaptabilité et l'adaptativité.

Un outil d'apprentissage (un hypermédia, en particulier) sera adaptable s'il peut être adapté à un utilisateur et à une utilisation donnés. Cette adaptation concerne à la fois le domaine des connaissances (choix dans la façon de présenter ces connaissances) et l'utilisation de l'outil. InterMed, l'hypermédia que nous avons développé, intègre la notion de profil : les interactions d'un apprenant avec InterMed dépendent de son profil : ses acquis, ses préférences.

Un outil d'apprentissage sera adaptatif si, par lui-même, il "s'adapte" à un type d'utilisateurs et à une situation d'apprentissage donnés. Le TI Starguide (Claës 88) que nous avons couplé à InterMed permet de réaliser une forme d'adaptativité. StarGuide gère, avec une base de règles, le profil de l'apprenant et le communique à InterMed.

3.2. Le guidage de l'apprenant en interaction avec un hypermédia

Le guidage pourra prendre différentes formes :

- il pourra se faire par contraintes sur l'environnement d'apprentissage (certains liens hypermédiat seront inactifs pour un profil d'utilisateurs et une situation pédagogique donnés). Ici, le concept de guidage rejoint celui de l'adaptation. L'attribut "sensibilité" des zones qui partagent une image InterMed permet de rendre une zone réactive (i.e. une action sur cette zone provoque l'exécution d'un lien) à un moment donné et inactive à un autre moment et ceci en fonction des objectifs pédagogiques :

- il pourra consister à fournir des aides adaptées sous forme d'explications ;
- il pourra aussi se traduire par un cours fait de démonstrations et d'exercices.

Le guidage concerne le contenu d'un hyperdocument (les données multimédias) mais également l'outil hypermédia (comment l'utiliser ?) qui permet de structurer ces données.

Starguide permet de réaliser des formes de guidage. En particulier, son système d'aides fournit à la demande des aides adaptées. Lorsque l'apprenant est en interaction avec InterMed, il peut à tout moment demander de l'aide en frappant la touche <F1>. Ces aides sont de différentes natures : suggestions sur les actions à entreprendre, reformulations de l'énoncé du problème, aide sur l'utilisation de l'outil Starguide, points sur les connaissances acquises et sur celles qu'il lui reste à acquérir pour atteindre un objectif pédagogique donné. La figure 1 donne un exemple d'un écran proposé suite à un appel à l'aide. Une liste des aides disponibles est placée en surimpression sur l'image d'une pièce de musée (Cf. (Delta 90) pour une description du prototype "musée" développé avec les outils InterMed et StarGuide).

3.3. Un modèle de l'élève

Une représentation de l'élève est indispensable pour une adaptativité d'un système d'apprentissage. La base de règles "Suiveur" de Starguide réalise un suivi de l'évolution d'un apprenant. Différentes informations sur l'apprenant sont mémorisées par le "Suiveur". Concernant les interactions avec InterMed, ces informations sont obtenues à partir de la trace du cheminement de l'apprenant à l'intérieur de l'hypermédia.

Figure 1 : Exemple de coopération InterMed/StarGuide

4. InterMed/StarGuide

Partant de ces considérations théoriques, nous avons intégré dans un même environnement d'apprentissage un Tuteur Intelligent, StarGuide (avec son moteur d'inférences et ses bases de connaissances) et un hypermédia, InterMed que nous avons développé dans une optique de collaboration avec le système tuteur.

InterMed permet de structurer des données multimédia (image, son et texte) en créant entre elles des liens dynamiques. En mode auteur, InterMed, permet de composer des images à partir d'une ou plusieurs images bit-map, de les structurer en zones sensibles et d'y associer des liens.

Le schéma de la figure 2 illustre l'environnement d'apprentissage, résultat de la coopération InterMed/Starguide.

Figure 2 : Architecture de l'environnement d'apprentissage InterMed/StarGuide

StarGuide, le Tuteur, peut solliciter InterMed, l'hypermédia, pour mettre en oeuvre une stratégie de type découverte. Il peut également paramétrer InterMed pour l'adapter à un profil et à une situation d'apprentissage donnés. InterMed peut informer Starguide de l'évolution de l'apprenant. L'élève peut interagir à la fois avec le système tuteur (pour être guidé) et le système hypermédia (navigation).

Références

- (Beaufils 91) A. Beaufils : *Construction de bases de données multimédias, exercices sur le tri, la navigation ou le raisonnement*. Technologies hypermédiats, Séminaire INSERM/CNEFEI, Juin 91, pp. 65-78.
- (Claës 88) G. CLAËS : *Contribution à l'application de l'Intelligence Artificielle pour l'EAO*, Thèse 3e cycle, Université de Paris-Sud, 17 Octobre 88.

- (Daniel 90) M.C. Daniel-Vatonne : *Hypertextes : des principes communs et des variations*. TSI vol.9, n°6, 1990.
- (Papert 81) S. Papert : *Jaillissement de l'esprit*. Flammarion, 1981.
- (Paquette 91) G. Paquette : *Le rôle des métaconnaissances dans LOUTI, un générateur d'environnements d'apprentissage*. KMET 91.
- (Nicaud 88) J.F. Nicaud, M. Vivet : *Les tuteurs intelligents : réalisations et tendances de recherche*. TSI vol.7, n° 1,1988.
- (Delta 90) Equipe DELTA TMIE : *Spécification des prototypes*, Deliverable WP9, Janvier 91.
- (Vivet 91) M.Vivet : *Expertise pédagogique et Usage des tuteurs intelligents*. Journées Francophones : Formation Intelligemment Assistée par Ordinateurs, GENEVE, 9-10 janvier 1991.