
Optimisation par « hommilière » de chemins pédagogiques pour un logiciel d'e-learning

Grégory Valigiani*, **Evelyne Lutton****, **Cyril Fonlupt*** et **Pierre Collet*****

* *Laboratoire LIL, Université du Littoral, 56 rue Ferdinand Buisson 62228 Calais.
cyril.fonlupt@lil.univ-littoral.fr*

** *Équipe APIS, INRIA Futurs, 4 rue Jacques Monod 91893 Orsay.
evelyne.lutton@inria.fr*

*** *Équipe FDBT, LSIT, Université Louis Pasteur, Bd Sébastien Brant 67412 Illkirch.
pierre.collet@lsiit.u-strasbg.fr*

RÉSUMÉ. Cet article décrit l'adaptation de techniques d'optimisation par colonie de fourmis (OCF) à un environnement d'e-learning, car le matériel pédagogique en ligne peut être organisé en graphe, en utilisant comme arcs les liens hypertextes entre les différents items. L'idée est alors de trouver des parcours dans le graphe aidant les étudiants à progresser. L'OCF est une métaphore des fourmilières naturelles, mais dans ce cas, les agents se déplaçant sur le graphe sont des élèves humains déposant inconsciemment des phéromones dans l'environnement. Cependant, des tests ont montré que les humains ne se comportent pas comme des fourmis, ce qui signifie qu'il a fallu modifier le paradigme OCF pour qu'il puisse fonctionner avec des agents humains. Un néologisme a été créé pour décrire ce nouveau paradigme : l'optimisation par «hommilière».

ABSTRACT. This paper describes experiments aimed at adapting Ant Colony Optimisation (ACO) techniques to an e-learning environment, thanks to the fact that the available online material can be organised in a graph by means of hyperlinks between educational topics. The idea is to find paths in the graph making it easier for students to improve. ACO is based on an ant-hill metaphor. In this case, however, the agents that move on the graph are students who unconsciously leave pheromones in the environment. Tests showed that humans did not behave as ants, meaning that the ACO paradigm had to be modified so that it could work with human agents. A new word has been coined to describe the new paradigm: "man-hill" optimization.

MOTS-CLÉS : Hommilière, E-Learning, Optimisation par Colonie de Fourmis.

KEYWORDS: Man-Hill, E-learning, Ant Colony Optimisation.

1. Introduction

1.1. *Environnement informatique pour l'apprentissage humain (EIAH)*

Ce projet se place dans le cadre des Environnements Informatiques pour l'Apprentissage Humain (EIAH), et plus particulièrement sur le guidage de l'apprenant parmi les différentes unités constituant le logiciel. Les EIAH sont des environnements qui intègrent des agents humains (élèves ou enseignants) et artificiels et offrent en parallèle des conditions d'interaction ainsi que des conditions d'accès à des ressources formatives (humaines et/ou médiatisées) locales ou distribuées (Balacheff *et al.*, 1997).

L'avènement des technologies de l'information et de la communication (TIC) conduit d'une part, à différents travaux centrés sur la notion de ressource pédagogique : conception industrialisée de supports pédagogiques numériques (Crozat, 2001), indexation et normalisation des objets pédagogiques (Grandbastien, 2002), diffusion de ressources *via* des plateformes spécialisées (Paquette, 2000). Ces problématiques conduisent notamment à utiliser en EIAH des travaux sur la notion d'ontologie (Mizoguchi *et al.*, 1999) ou des technologies de type XML (de la Passardière *et al.*, 2001). Les TIC proposent d'autre part, un cadre particulièrement pertinent pour les travaux liés à l'apprentissage collaboratif comme la mise en place de situations de pédagogie de projet à distance (George, 2001) et créent de nouvelles problématiques comme le suivi d'activités à distance (Després, 2001).

Ces systèmes n'embarquent cependant pas nécessairement de connaissances du domaine de l'apprentissage, sortant en cela du cadre classique de l'enseignement assisté par ordinateur (EAO).

1.2. *Cas Paraschool*

Paraschool est le leader français actuel en matière de soutien scolaire à distance avec plus de 250 000 utilisateurs, dont 5 000 utilisateurs privés et le reste dans le cadre de contrats avec des établissements scolaires. En 2001, Paraschool a voulu mettre en place un système évolutif capable de proposer des chemins pédagogiques «intelligents», adaptés à chaque élève.

Il est rapidement apparu que dans le cadre des EIAH, la technique d'optimisation par colonie de fourmis (OCF) (Moyson *et al.*, 1988; Dorigo *et al.*, 1997; Bonabeau *et al.*, 1999; Bonabeau *et al.*, 2000) était novatrice et semblait pouvoir s'appliquer directement au problème grâce au fait que les différents items pédagogiques composant le logiciel Paraschool pouvaient s'organiser sous forme d'un graphe qui serait parcouru par les milliers d'élèves inscrits chez Paraschool.

Une première version de l'algorithme a donc été mise en place sur cette base, mais les premiers tests en grandeur nature ont montré que le paradigme maintenant bien connu de l'OCF n'était pas directement transposable au système de Paraschool, car

un groupe d'humains ne présente pas les mêmes caractéristiques qu'une colonie de fourmis artificielles.

Ce document présente donc le cahier des charges de Paraschool et l'adaptation de l'optimisation par colonie de fourmis à la résolution du problème posé par Paraschool, aboutissant au nouveau concept d'«optimisation par Hommilère». Les principales différences avec les algorithmes d'OCF sont exposées, suivies d'une description des premiers tests en grandeur réelle montrant que le système présente un comportement attendu. Une conclusion récapitule les différents points apparus dans le cadre de cette étude et décrit les perspectives futures.

1.3. *Le logiciel*

Le site Paraschool est un site de soutien scolaire qui propose des compléments pédagogiques, des fiches de cours et des exercices. Il est fondé sur des principes comportementaux : le résultat de l'exercice traité est tout de suite présenté à l'élève. En cas d'erreur, un mécanisme de remédiation intervient : l'élève peut être alors redirigé vers le point du cours non assimilé. L'autre stratégie pédagogique souvent utilisée (mais moins adaptée au système de Paraschool) est le constructivisme, où le professeur est censé mettre l'élève en présence de situations et d'éléments qui doivent lui permettre de construire ses propres connaissances.

L'abonnement à Paraschool permet à l'élève d'accéder à une ou plusieurs matières et d'avoir accès à une assistance en direct, par mail ou par *chat* avec un professeur. L'essentiel du contenu concerne les élèves de collège-lycée et se concentre sur plusieurs matières telles que les mathématiques, la physique, le français, l'économie...

A la fin d'une session, les résultats sont enregistrés et quelques courbes, assorties de commentaires pédagogiques, permettent à l'élève de suivre sa propre progression.

Avant que le travail présenté dans cet article ne soit mis en œuvre, l'élève pouvait se déplacer dans le graphe des items grâce à deux types de navigation :

– *Navigation libre* Passé le choix de la matière à travailler (mathématiques par exemple), l'élève se voit proposer une liste de chapitres correspondant à son niveau, dans lesquels il trouvera trois catégories d'items représentant des entités pédagogiques :

- *points de cours* Ce sont des rappels de cours validés par des tests d'application directe du cours,

- *savoir-faire* Ces items représentent les méthodes utiles à connaître pour une bonne utilisation des notions de cours. C'est la vision pratique du cours,

- *exercices* Il faut valider les exercices pour valider le chapitre.

Chaque item se termine par une validation interactive, la plupart du temps sous forme de questionnaire à choix multiples. Le résultat est analysé et commenté. En cas d'échec, une remédiation est proposée. En cas de réussite, l'élève a la possibilité de

revenir sur les différents menus pour choisir le prochain item qu'il va effectuer, sans contrainte de thème, de cursus ou de matière,

– *Navigation guidée* l'étudiant peut également, dans un chapitre donné, se laisser guider interactivement le long d'un chemin (une suite d'items) prédéfini par l'équipe pédagogique. Cette possibilité était implémentée de façon linéaire et déterministe. C'est ici qu'intervient le système mis en place : l'hommilière aura la charge de proposer une suite «intelligente» à l'exercice qui vient d'être terminé, l'objectif étant d'optimiser l'apprentissage de chaque élève sur le chemin pédagogique proposé.

1.4. *Desirata, objectifs*

Paraschool cherchait une technique pour rendre son site plus attrayant et plus efficace du point de vue pédagogique. L'objectif de ce travail était donc triple.

1) *Emergence* Tout d'abord, le système doit permettre d'améliorer les parcours pédagogiques proposés, en faisant émerger de nouveaux chemins auxquels l'équipe pédagogique de Paraschool n'a pas pensé.

2) *Adaptabilité* Ensuite, comme le site comporte des milliers d'items, il faut que l'intégration de nouvelles unités ou la modification de contenus puisse être prise en compte de manière automatique et dynamique par le système, sans nécessiter d'intervention manuelle, qui serait trop lourde à gérer.

3) *Individualité* Pour finir, un système idéal devrait présenter un contenu adapté à chaque utilisateur mais aussi aux groupes (cas d'un professeur en salle de TD équipée d'ordinateurs, désirant que les élèves travaillent sur un point précis du programme).

Les deux premiers points peuvent être abordés par un paradigme d'Optimisation par Colonie de Fourmis. Le troisième ne peut pas l'être directement, car les fourmis (réelles ou artificielles) ont un comportement collectif sans aucune variation liée à l'individu.

2. Critères d'optimisation pour l'émergence automatique et adaptative de chemins pédagogiques

Si l'on considère la représentation du système de Paraschool comme un graphe (c'est-à-dire un groupe d'items reliés entre eux par des arcs) l'objectif est alors de trouver un «bon» chemin (succession d'items à effectuer) permettant de «maximiser» l'apprentissage de l'élève qui le parcourt.

Pour commencer, des chemins pédagogiques sont construits à partir de la navigation guidée issue de la vision des enseignants (cf. section 4.1.1), mais il semble possible que ce qui apparaît comme une suite logique d'exercices pour un enseignant, ne le soit pas pour un élève particulier, pour qui un enchaînement différent aurait été mieux adapté.

D'ailleurs, le chemin suggéré par l'enseignant est-il le plus adapté pour la majorité des élèves ? Dès lors que l'on se pose cette question, une bonne façon d'y répondre consiste à rechercher les chemins optimaux du point de vue de l'élève (et non de l'enseignant), en utilisant un paradigme spécialisé dans l'optimisation stochastique de chemins dans un graphe, à savoir l'optimisation par colonie de fourmis (OCF).

Le parcours d'un utilisateur prend alors cette forme : dans un niveau, l'élève choisit, une matière, puis un thème. Le système lui propose alors une liste d'items parmi lesquels il doit en choisir un. Une fois l'item terminé, le système prend en compte la réussite ou l'échec de l'élève sur l'item pour mettre à jour les chemins proposés, puis propose une autre liste d'items à l'élève (qu'il n'est pas contraint de suivre).

Se pose alors le problème de l'évaluation de la pertinence des items proposés à la suite d'un exercice. Ce point est très important car il détermine le type de chemins que l'on souhaite voir émerger. Dans un premier temps, l'objectif était de maximiser le succès des élèves d'une manière globale ; choix qui s'est avéré naïf.

3. Premier essai : utilisation d'un paradigme d'OCF

L'optimisation par colonie de fourmis est une technique d'optimisation inspirée par un travail de biologiste (Deneubourg *et al.*, 1983) repris par des informaticiens (Moyson *et al.*, 1988) et largement exploité et développé par Marco Dorigo dans les années 90 (Coloni *et al.*, 1991; Dorigo, 1992). L'idée consiste à imiter le comportement de fourmis réelles qui collaborent, par exemple pour la recherche de sources de nourriture en mélangeant comportement d'exploration aléatoire et suivi de traces chimiques (phéromones) laissées sur le terrain par leur consœurs. C'est cette forme de communication ainsi que l'idée de faire coopérer un grand nombre d'agents simples et localisés qui est à la base de l'heuristique développée par Dorigo.

L'optimisation par colonie de fourmis a rapidement prouvé son efficacité dans le cadre de l'optimisation combinatoire et forme aujourd'hui, avec l'optimisation par essaims particulaires, un domaine de recherche à part entière, connu sous le nom de *Swarm Intelligence* (Bonabeau *et al.*, 2000; Kennedy *et al.*, 2001).

A une échelle supérieure, les insectes sociaux sont ainsi capables de comportements intelligents : une fourmilière est capable de trouver dynamiquement en trois dimensions, malgré les obstacles et la nature variable de l'environnement, un chemin optimal vers un ou plusieurs points de nourriture. Les termites sont capables de construire des termitières de plusieurs mètres de hauteur à l'architecture quasi-optimale, et il en va de même des nids d'abeille aux alvéoles parfaitement hexagonales. Ces exemples ne sont pas des cas isolés, ce qui signifie qu'il est possible de les reproduire si l'on en a compris le mécanisme.

Un exemple typique (découverte du plus court chemin entre une fourmilière et un point de nourriture figure 1) permet de mettre en évidence les mécanismes requis pour le bon fonctionnement du paradigme.

– *Dépôt de phéromones* Les fourmis explorent aléatoirement le terrain. Lorsqu'elles trouvent une source de nourriture, elles en prélèvent un peu et tentent de retourner au nid en libérant des phéromones le long de leur chemin. Le chemin le plus court étant aussi le plus rapide, sa concentration en phéromones augmentera plus vite à nombre de fourmis égal. En suivant de préférence les traces chimiques, les fourmis seront très vite encouragées à suivre le chemin le plus court par un phénomène de renforcement.

– *Evaporation des phéromones* les phéromones s'évaporent dans le temps, rendant ainsi leur trace éphémère. Ce mécanisme permet ainsi aux chemins établis de ne pas être statiques, mais de s'adapter aux modifications de l'environnement (un chemin anciennement optimal devenu inutilisable suite à un événement extérieur finira par disparaître faute d'être utilisé).

– *Comportement aléatoire des fourmis* Ensuite, il arrive quelquefois que des fourmis étourdies se trompent et s'écartent du chemin de phéromones. Si, par chance, une fourmi égarée trouve un chemin plus court, la trace de phéromone qu'elle laissera derrière elle sera plus fraîche, indiquant aux autres fourmis qu'il existe un chemin plus court pour accéder à la nourriture. Ainsi, c'est le mécanisme d'erreur dans le suivi de trace de phéromone qui permet d'optimiser le chemin par la découverte de raccourcis, aboutissant *in fine* à l'établissement d'un chemin optimal entre fourmilière et nourriture.

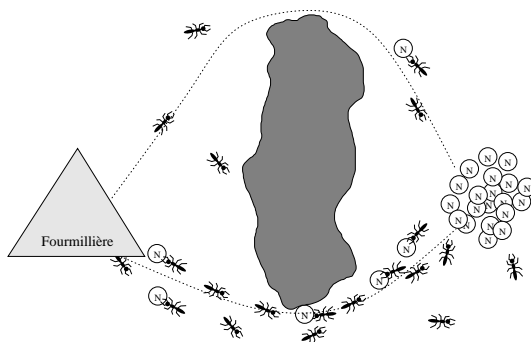


Figure 1. Un problème naturel typique : un nid, une source de nourriture et deux chemins possibles, un court en dessous de l'obstacle et un long au dessus

Enfin pour que l'algorithme puisse fonctionner, il faut un nombre suffisant de fourmis. En dessous de cette masse critique, il n'y aurait pas d'émergence d'une solution face aux comportements aléatoires des fourmis.

3.1. L'algorithme d'OCF sur Paraschool

Pour faire émerger des parcours pédagogiques adaptatifs maximisant le succès, la première approche a donc été d'implémenter un algorithme d'optimisation par colonie

de fournis sur le graphe de Paraschool, en profitant du grand nombre d'élèves inscrits pour associer un élève à une fourmi, se déplaçant dans le graphe de Paraschool et libérant des phéromones artificielles de succès ou d'échec suivant les items rencontrés, implémentant ainsi pour la première fois un algorithme d'optimisation par essaim particulière à grande échelle, utilisant des utilisateurs humains comme agents (seul un autre exemple a été trouvé d'une de fourmilière artificielle sur Internet a été trouvé, pour classifier des sessions web, mais ce n'était pas un utilisateur humain qui était associé à une fourmi, mais une session web (Labroche *et al.*, 2003b)).

– *Structure du système de Paraschool* : le logiciel Paraschool d'origine était implémenté pour une utilisation *via* Internet. Il utilisait donc des pages reliées par des liens hypertextes, qu'il a été possible de structurer en graphe pour se rapprocher des algorithmes proposés par Marco Dorigo.

– *Un comportement aléatoire possible* : la navigation libre permet à l'utilisateur de ne pas être restreint aux propositions faites par le système et donc de pouvoir choisir son propre enchaînement d'items, proposant ainsi de nouveaux chemins au système (tout comme les fourmis trouvent de nouveaux chemins dans l'environnement).

Dans les cas classiques d'OCF, on a recours à un grand nombre de fourmis artificielles dont on code le comportement suivant le résultat souhaité. Dans le cas présent, il est difficile d'infléchir le comportement de l'élève humain, et il faudra donc s'y adapter.

3.2. Adaptations par rapport à une implémentation utilisant des fourmis artificielles

Après plusieurs mois de tests sur des petites expériences fictives puis en réel sur le système de Paraschool, il s'est avéré qu'associer bijectivement une fourmi à un utilisateur demande un certain nombre de modifications par rapport au paradigme standard de l'optimisation par colonie de fourmis :

– *Comportement* : habituellement, les fourmis artificielles sont *programmées* pour résoudre un problème donné. Dans notre cas, les élèves *humains* ne suivent pas un algorithme en particulier. Ils vont là où ils veulent et les seules influences possibles passent par les suggestions faites par le système. Ce comportement ressemble plus au comportement possiblement erratique des vraies fourmis.

– *Activité* : les fourmis artificielles (ou réelles) sont constamment actives dans tout l'environnement, à la différence des élèves qui ont des périodes de repos (vacances...) et qui n'étudient les différents thèmes qu'à des moments précis de l'année. Une évaporation temporelle des phéromones effacera toute l'information portée par les arcs d'un domaine (par exemple, le thème des «vecteurs» en classe de seconde) qui n'aura pas été visité depuis l'année précédente, ce qui signifie qu'il faudrait tout reprendre à zéro l'année suivante.

– *Contraintes particulières* : de nouvelles contraintes apparaissent avec des agents humains. En effet, des fourmis artificielles (ou réelles) ne se plaindront jamais, par

exemple, de repasser plusieurs fois au même endroit. L'élève, lui, reconnaîtra tout de suite un exercice déjà proposé et risque de s'agacer si le système lui propose le même item cinq fois de suite. . .

– *Altruisme* : les colonies de fourmis naturelles et autres insectes sociaux se comportent de manière altruiste en ce qui concerne la colonie, jusqu'à se battre pour elle jusqu'à la mort. L'avenir de la colonie entière est plus importante que celui d'un seul individu, et de ce fait, certaines fourmis n'hésitent pas à aller explorer des zones lointaines à leurs risques et périls, pour le seul bien de la communauté. Ces comportements que l'on peut retranscrire dans la programmation de fourmis artificielles, sont rarement observés chez les humains. . . Il y a peu de chances qu'un élève aille jamais explorer une partie obscure du graphe Paraschool pour le seul bien de ses copains de classe.

Du fait de ces différences, il est nécessaire d'apporter plusieurs changements au paradigme original pour qu'on puisse utiliser une «colonie» d'être humains à des fins d'optimisation.

De l'évaporation à l'érosion

Le premier changement testé concerne la volatilité des phéromones. Elle joue un rôle capital car elle permet de donner à l'algorithme d'optimisation un caractère adaptatif.

Les premiers tests en vraie grandeur sur le logiciel Paraschool ont montré que l'évaporation temporelle des OCF n'était vraiment pas adaptée aux agents humains. Tout d'abord, sur quelle base évaporer les phéromones ? Sur une base journalière, hebdomadaire, mensuelle ?

Lors des premiers tests, il s'est avéré que les parties du graphe non utilisées pendant un certain temps (vacances scolaires, par exemple) perdaient rapidement l'information collectée.

Nous avons donc remplacé l'évaporation temporelle des phéromones par un mécanisme que nous avons appelé d'*érosion*, qui se produit à l'occasion du passage d'un élève sur un nœud et non plus par rapport au temps écoulé.

Ainsi, cette évaporation des phéromones n'est plus globale, mais locale : lorsqu'un élève aura validé un item A duquel sort des arcs vers une liste d'items (B, C, D) et qu'il a choisi de suivre l'arc menant vers B , ce n'est que l'ensemble des arcs sortants de l'item A (c'est-à-dire les arcs vers B, C, D) qui seront érodés, car l'élève a eu la possibilité de les emprunter. C'est donc une diminution locale de la valeur des phéromones qui est pratiquée, causée par le passage d'un étudiant (d'où le nom d'érosion). Par la suite, suivant la réussite ou l'échec de l'élève à l'item B , des phéromones de succès ou d'échec seront déposées sur l'arc $A \rightarrow B$.

Cette érosion locale amenée par le passage procure l'avantage de ne pas perdre l'information durant les périodes d'inactivité, tout en conservant un équivalent au processus d'évaporation pour les arcs non sélectionnés.

4. Application de l'OCF au système Paraschool

4.1. Paramètres globaux

Plusieurs paramètres sont nécessaires afin de formaliser correctement l'algorithme. Ces paramètres peuvent être classés suivant trois catégories symbolisant le compromis fait par le système entre la pédagogie, la collectivité et l'individu.

4.1.1. Paramètres statiques : Poids pédagogiques

Dans l'application Paraschool, le résultat souhaité est l'émergence de chemins maximisant le progrès des élèves, possiblement différents de ceux suggérés par les enseignants. Mais sans initialisation, en attendant que des chemins émergent du passage d'élèves, le système proposerait alors des enchaînements aléatoires. Pour éviter ceci (et pour tout de même conserver l'avis des enseignants), il leur a été demandé de fournir leur vision de l'enchaînement des exercices.

Pour l'intégrer au système (en attendant de récolter suffisamment de phéromones pour que l'information portée par les arcs soit significative) les arcs sont donc initialisés par l'équipe pédagogique en utilisant des poids appelés *poids pédagogiques*. Plus ce poids est important, plus l'arc concerné sera avantagé lors de la sélection de l'arc qui sera présenté à l'élève.

A chaque fois qu'un professeur crée un nouveau thème, il lui est demandé de réfléchir à la structure du graphe relatif au thème ainsi qu'aux poids pédagogiques qu'il va associer à chaque lien. En l'absence de phéromones provenant des élèves, seul ce poids pédagogique joue. Les phéromones viendront ensuite modifier cet environnement pour qu'il s'adapte aux utilisateurs.

Pour éviter des échelles de représentations de poids différentes entre les professeurs, les poids pédagogiques sont convertis en valeurs relatives par rapport à tous les arcs sortant d'un même item (cf. figure 2). Si w_1, \dots, w_n sont les poids suggérés par le professeur pour les n arcs sortant d'un item, le poids pédagogique relatif de l'arc j sera :

$$w_{rel_j} = \frac{w_j}{\sqrt{\sum_{i=1}^n w_i^2}}$$

Cette relativité pondérale donne une cohérence à tous les poids pédagogiques des arcs du graphe.

La structure du graphe créée par les professeurs n'est pas figée. En effet si un élève décide de sauter d'un item à un autre sans suivre le conseil du professeur et donc sans

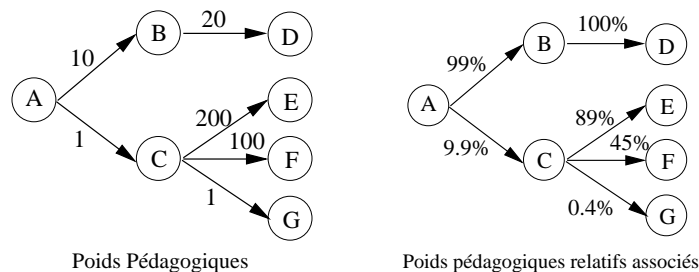


Figure 2. Exemple de calcul de poids pédagogiques relatifs

passer par un arc déjà existant, en utilisant le navigateur par exemple, un nouvel arc est créé avec un poids pédagogique par défaut de 1. Ce poids par défaut permet de donner une petite valeur pédagogique aux choix des élèves : si à partir d'un item, plusieurs élèves décident de «sauter» vers de nouveaux items, plusieurs arcs seront créés et le poids relatif pédagogique de l'arc conseillé par le professeur en sera légèrement diminué du fait de la concurrence de ces nouveaux arcs.

L'équipe pédagogique peut tirer des informations utiles de l'analyse de la création de nouveaux arcs. Par exemple, la création de plusieurs arcs entre un exercice de physique et des éléments de cours de trigonométrie peut souligner la nécessité de proposer aux élèves des rappels de cours en trigonométrie avant d'entamer l'exercice de physique concerné.

4.1.2. Paramètres dynamiques : phéromones

Les phéromones sont des informations stigmergiques¹ déposées dans le graphe à destination des autres élèves, pour les aider à se déplacer dans l'environnement pédagogique. Il y a deux types de phéromones déposées sur les arcs menant à un item : φ^+ pour les phéromones symbolisant le succès et φ^- pour l'échec. Ainsi, à la suite de la validation d'un item A par l'élève, le système peut choisir parmi les arcs sortants de cet item celui proposant le meilleur rapport succès / échec, grâce à l'information présente sur les arcs. L'information présente sur l'arc $A \rightarrow B$ est une estimation à un instant donné du taux de réussite des élèves à l'item B en provenance de l'item A .

Mais il faut bien garder en tête que la solution cherchée est un *chemin* pédagogique, donc une succession de liens visités. Pour implémenter cette notion de chemin, les phéromones (φ^+ ou φ^-) sont libérées sur les n derniers arcs de l'historique de l'élève. En effet, le succès (ou l'échec) sur un item donné est conditionné par ce qui a été vu précédemment par l'élève. Bien évidemment, cette influence diminue avec le temps et l'espace : plus le nœud est éloigné dans l'histoire de l'élève, moins il subit, en théorie, l'influence de la position actuelle de l'élève. En pratique, on ne dépose des

1. La stigmergie est l'action de communiquer avec ses semblables par dépôt d'information dans l'environnement.

phéromones que sur les trois derniers arcs visités avec un taux de diminution de $1/k$ (k représentant l'ancienneté des arcs visités) cf. figure 3. Le dépôt initial correspond à une valeur de 0,05.

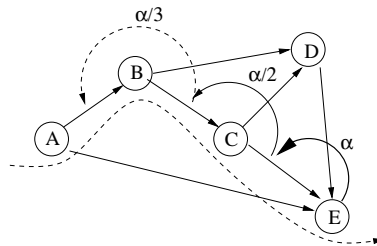


Figure 3. Dépôt effectif de phéromones après validation de l'item E, en fonction du chemin que l'élève a emprunté (en pointillé). Si l'élève a parcouru le chemin $A \rightarrow B \rightarrow C \rightarrow E$ et qu'une dose de phéromone α est libérée sur l'arc $C \rightarrow E$, une dose $\alpha/2$ sera libérée sur l'arc $B \rightarrow C$ et une dose $\alpha/3$ sera libérée sur l'arc $A \rightarrow B$

4.2. Résultats des premiers tests de l'OCF sur le système Paraschool

Les premiers résultats ont été fort concluants : des chemins sont rapidement apparus qui menaient du début à la fin d'une leçon en passant par le plus petit nombre possible d'exercices faciles.

Bien que démontrant le bon fonctionnement de l'algorithme OCF avec évaporation, cela a amené l'équipe pédagogique à s'interroger sur ce que devait être le réel objectif de l'optimisation (qui n'était clairement pas de valider les leçons en passant par les exercices les plus faciles).

Au vu de ces résultats, l'équipe pédagogique a décidé de changer d'objectif : plutôt que d'essayer de trouver des chemins maximisant le succès, il a été décidé que l'algorithme devrait servir à trouver une succession d'exercices que les élèves arriveraient à résoudre avec une probabilité de 60 % (une valeur supérieure à 50 % a été choisie pour ne pas les décourager).

5. Introduction du concept d'hommilière

Une première entorse avait été faite à l'Optimisation par Colonie de Fourmis par l'introduction du concept d'érosion, nécessité par des agents humains ne pouvant explorer continûment l'ensemble du système. Une autre contrainte devait aussi être respectée : celle de spécialiser la navigation pour chaque élève (tenir compte du fait qu'un élève avait déjà visité un item, par exemple, ou mieux, intégrer une forme de remédiation spécifique à l'élève).

Pour atteindre cet objectif, un nouveau type de phéromones a été introduit : des *phéromones individuelles multiplicatives* (PIM) venant modifier les valeurs globales trouvées sur les arcs d'après le profil de l'élève.

Si l'optimisation par colonie de fourmis s'appuie sur la collaboration d'agents identiques créés artificiellement, avec des caractéristiques altruistes et explorant l'ensemble de l'environnement en permanence, les nouveaux éléments introduits dans ce travail permettent d'utiliser des agents humains, individualisés et explorant l'environnement de manière irrégulière.

Ces modifications ouvrent la porte à quantité d'applications sur le web jusqu'alors fermées à l'OCF pour les raisons évoquées ci-dessus. Avec le concept d'*optimisation par hommilière*, en revanche, dès lors que plusieurs centaines d'utilisateurs utilisent un même site centralisé (forum, site de diffusion de média...) il devient possible d'utiliser leurs visites pour améliorer le site de manière automatique, par exemple.

5.1. Mise en œuvre des critères locaux

Les poids et phéromones évoqués jusqu'ici sont des facteurs globaux collectifs. Tout comme dans les fourmilières naturelles et artificielles, ces informations stigmergiques concernent l'ensemble des agents (élèves) et représentent les informations qui seront utilisées pour construire des cheminements optimaux.

Pour «humaniser» les propositions faites aux agents, on essaiera de tenir compte des notions d'individualité et d'*ego*, absentes des paradigmes à base de fourmis, mais chers à la psychologie humaine :

- un nouveau type de phéromone sera implémenté, à destination des seuls individus qui les ont émises (à l'opposé des phéromones globales perçues de tous),
- de plus, comme il y a besoin de prendre en charge plusieurs critères dans le profil d'un individu, plusieurs types de phéromones seront utilisées, à l'instar de ce qui se fait dans les fourmilières naturelles.

Les facteurs individuels que l'on peut prendre en compte sont alors nombreux (préférences, excellence, agenda de travail, etc.). Les informations seront déposées sur des arcs et s'évaporeront ou s'éroderont à la manière de phéromones globales. La seule différence est que l'information qu'elles portent ne sera utilisée que par l'élève qui les aura déposées (d'où leur nom de Phéromones Individuelles). Les phéromones individuelles seront utilisées comme facteurs multiplicatifs à appliquer aux phéromones globales, permettant ainsi de tenir compte des informations laissées par la communauté, tout en les biaisant pour tenir compte du profil de l'élève.

5.1.1. Phéromone individuelle multiplicative (PIM) d'historique

La PIM d'historique (φ_h) est utilisée comme un coefficient multiplicatif à appliquer sur la valeur des phéromones globales portées par l'arc sur laquelle elle a été déposée.

Initialement, la PIM historique est fixée à 1 (l'élément neutre pour la multiplication) et donc n'affecte pas le calcul de la *fitness* (voir section 6). Lorsqu'un élève réussit un exercice, la PIM Historique φ_h est divisée par 2 (et vaut donc 0.5) sur tous les arcs menant à l'exercice. La PIM agissant comme facteur multiplicatif, si l'élève a la possibilité de revenir sur l'exercice immédiatement après l'avoir résolu avec succès, la valeur de l'arc y menant est divisée par 2, (ce qui divise par 2 les chances de l'élève de se voir proposer le même exercice).

La mémoire de l'élève s'estompant avec le temps, cette PIM Historique ne s'érode pas, mais s'évapore avec le temps comme les phéromones standard de l'OCF, mais avec une petite différence : les phéromones de l'OCF étant cumulatives, elles s'évaporent en revenant vers l'élément neutre pour l'addition, c'est-à-dire 0.

La PIM Historique étant une phéromone multiplicative, elle doit s'évaporer pour revenir vers l'élément neutre de la multiplication, soit la valeur 1, pour permettre à l'arc de reprendre sa probabilité de sélection initiale.

Cette «évaporation multiplicative» (retour vers la valeur 1) est décrite par l'équation suivante :

$$\varphi_{h_{t+1}} = \varphi_{h_t} \left(1 + \frac{1 - \varphi_{h_t}}{\varphi_{h_t}} \frac{1 - e^{-\gamma t}}{1 + e^{-\gamma t}} \right)$$

où t représente le temps écoulé depuis la dernière visite et γ est une constante de temps qui règle la vitesse du phénomène. Idéalement, elle doit être réglée pour correspondre à la volatilité de la mémoire de l'élève...

5.1.2. Poids agenda

A l'inverse de la phéromone d'historique, la PIM agenda est un facteur multiplicatif poussant l'utilisateur à aller vers certains items, qu'ils aient été conseillés par un professeur, ou que le système ait jugé nécessaire d'y passer (remédiation).

Contrairement aux phéromones s'évaporant avec le temps, cette PIM ne reprendra une valeur de 1 qu'une fois l'item validé, sans quoi cela reviendrait à valider les items d'un agenda à l'ancienneté.

5.2. Conclusion

La transformation de l'OCF en optimisation par hommilière permet de répondre aux trois souhaits initiaux de Paraschool (émergence de nouveaux chemins, adaptativité et individualité).

6. Evaluation des arcs présentés à l'élève

A l'issue d'un item d'où plusieurs arcs sortent, il faut en choisir un à suggérer à l'élève. On utilisera le terme de *fitness* pour désigner la note des arcs en compétition.

L'expression de la fonction de *fitness* prend alors la forme suivante :

$$fitness_d'un_arc = \varphi_h \cdot \left(\begin{array}{ll} \left\{ \begin{array}{l} w_p \cdot (1 - w_1 \cdot (\varphi^+ + \varphi^-)) \\ 0 \end{array} \right. & \begin{array}{l} si (1 - w_1 \cdot (\varphi^+ + \varphi^-)) > 0 \\ sinon \end{array} \\ - \left\{ \begin{array}{l} w_2 \cdot \left(\frac{60}{100} - \frac{\varphi^+}{\varphi^+ + \varphi^-} \right) \\ w_3 \cdot \left(\frac{\varphi^+}{\varphi^+ + \varphi^-} - \frac{60}{100} \right) \end{array} \right. & \begin{array}{l} si \left(\frac{60}{100} \geq \frac{\varphi^+}{\varphi^+ + \varphi^-} \right) \\ si \left(\frac{60}{100} < \frac{\varphi^+}{\varphi^+ + \varphi^-} \right) \end{array} \end{array} \right)$$

où w_p représente le poids pédagogique, φ_h la phéromone d'historique, φ^+ et φ^- les phéromones de succès et d'échec et w_i les paramètres de réglage.

La première partie reflète la confrontation entre la «pédagogie» (poids pédagogique w_p) et la «collectivité» (quantité de phéromones $\varphi^+ + \varphi^-$). L'idée est de réduire le poids pédagogique lorsqu'il y a suffisamment de passage de fourmis. Ainsi, le poids pédagogique est diminué proportionnellement à la quantité de phéromones présentes sur l'arc. Au delà d'un certain nombre de passages (que le poids w_1 permet d'ajuster), le poids pédagogique n'a plus d'effet et l'adéquation au taux de réussite de 60% reste le seul objectif à atteindre. Si, en revanche, le passage des fourmis se réduit, le poids pédagogique reprend de l'importance. Cet effet peut être vu comme une force de rappel, qui aura tendance à revaloriser l'opinion de l'enseignant si la fréquentation diminue.

La deuxième partie de l'équation représente l'adéquation au taux de succès (objectif ici fixé à 60%), calculée à partir des deux types de phéromones (φ^+ et φ^-). Les poids w_2 et w_3 ont été dissociés afin d'avoir des pentes différentes au-dessus et en dessous du taux de succès de 60%, afin qu'il soit préférable de se trouver au dessus des 60% qu'en dessous.

Choix des arcs

Lorsqu'un élève valide un item, il convient de choisir parmi les arcs qui sortent de cet item celui qui sera proposé à l'élève. C'est le rôle de l'opérateur de sélection, qui utilise pour ce faire la valeur renvoyée par la fonction de *fitness* décrite ci-avant. Une autre nouveauté par rapport aux algorithmes d'OCF habituels est que la procédure de sélection est empruntée à l'évolution artificielle pour garder une composante stochastique contrôlable, afin que le système reste dynamique.

Pour autoriser les «erreurs d'aiguillage» nécessaires à l'adaptabilité du système, l'opérateur de sélection choisira un arc de façon aléatoire, mais biaisée par les valeurs de *fitness* portées par les arcs, permettant aux arcs avec une *fitness* élevée d'avoir plus de chance d'être choisis.

Cette part de hasard est une caractéristique cruciale de la procédure de sélection dont un paramètre sera la pression sélective s . Plus s est grand, plus le tirage se laisse guider par la fonction de *fitness* et plus les arcs forts auront tendance à dominer les arcs faibles.

Parmi les nombreux opérateurs de sélection utilisés dans les algorithmes évolutionnaires, le tournoi stochastique a été choisi pour ses caractéristiques. Son fonctionnement consistera à tirer 2 individus au hasard et de choisir le meilleur avec une probabilité p . On voit bien que p conditionne la pression sélective : plus p est grand, plus il est probable que le meilleur des deux arcs l'emporte. Outre son efficacité algorithmique, le tournoi stochastique permet de régler finement la pression sélective en jouant sur le paramètre p .

7. Tests de validation

7.1. Vérification de la possibilité d'inversion

Dans le but de valider le système et de l'ajuster, l'expérience utilisée pour démontrer l'adaptabilité des fourmis naturelles à trouver le chemin le plus court (Deneubourg *et al.*, 1990) a été transposée dans un environnement pédagogique (cf. figure 4). C'est un simple graphe constitué de quatre nœuds pédagogiques : un nœud de départ S , deux nœuds intermédiaires I_b et I_g et un nœud de fin E .

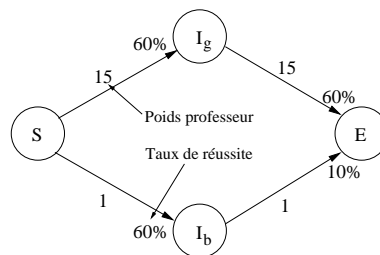


Figure 4. Vérification de la possibilité d'inversion de la valeur de fitness dans un cas simple

Supposons que l'équipe pédagogique de Paraschool pense que la meilleure façon de valider le nœud E en venant du nœud de départ S est de passer par le nœud I_b plutôt que de passer par I_g . Elle biaise donc le graphe en fixant le poids pédagogique à $15/20$ sur les arcs du haut et à $1/20$ (valeur par défaut) sur ceux du bas.

Si le professeur s'est trompé (c'est-à-dire que, pour réussir à l'item E , il vaut mieux passer par I_g que par I_b), c'est le cas typique où l'hommilière doit être capable de détecter et de corriger l'anomalie. Pour les tests en automatique, le taux de réussite a été fixé à 60% (normal) sur tous les arcs sauf sur l'arc connectant I_b à E qui est fixé à 10% (peu de chance de réussir si l'on n'est pas passé par I_b).

Cette simple expérience a été utilisée pour ajuster les paramètres afin que la valeur de la *fitness* du chemin établi par les professeurs puisse être dépassée par la *fitness* d'un meilleur chemin trouvé par les élèves (inversion de la *fitness*).

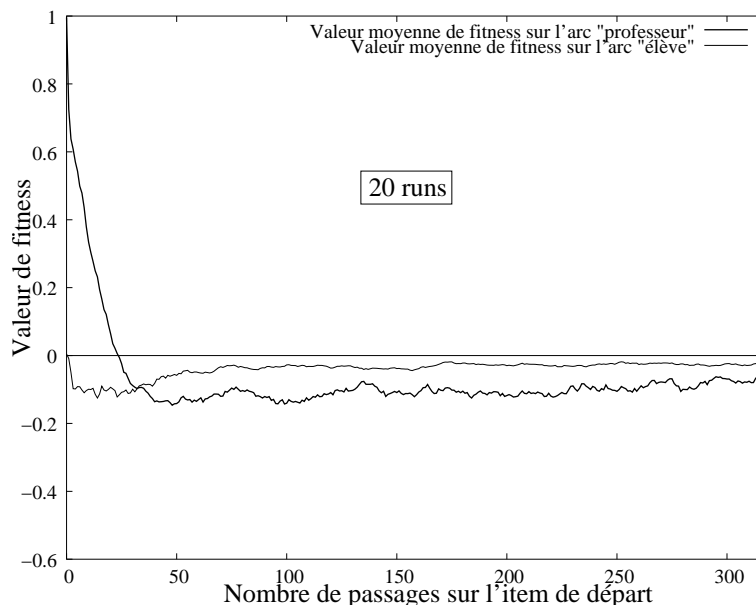


Figure 5. Valeur moyenne sur 20 essais de la valeur de fitness des arcs sortants du nœud *S* : le professeur (courbe en gras) conseille le mauvais chemin

La figure 5 montre la moyenne sur 20 tests de la valeur de la *fitness* de l'arc «professeur» (gras) et de l'arc «élève» (pointillé), lorsque le professeur conseille le mauvais arc en fixant le poids pédagogique à 15/20.

Comme prévu, la valeur de la *fitness* (en gras) diminue de façon constante pendant les 40 premiers passages jusqu'à ce que le nombre de phéromones déposés soit significatif. Alors les valeurs de *fitness* s'inversent, ce qui signifie que les élèves se verront proposer d'aller vers I_g , plutôt que d'aller vers I_b comme l'avait conseillé le professeur.

En moyenne, le système se comporte comme souhaité (l'inversion a bien lieu). De plus, même si le professeur conseille un mauvais arc, le système aura tendance à re-proposer périodiquement l'arc. C'est cette dynamique qui va empêcher une convergence trop rapide du système.

L'inversion constatée servira aussi à l'équipe pédagogique comme outil d'audit pour détecter les anomalies qui peuvent être dues à des exercices erronés (infaisables) ou trop faciles. Ainsi, l'hommière peut détecter des problèmes sémantiques dans les exercices.

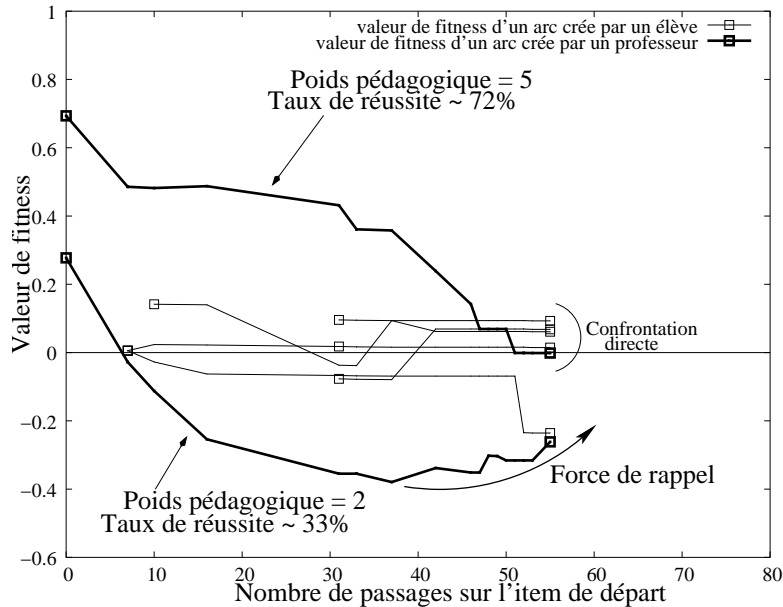


Figure 6. La courbe en gras du dessus représente un arc qui aboutit à un nœud avec un taux de succès actuel de 70 % alors que celle de dessous n'a qu'un taux de succès de 33%

7.2. Mise en production

La fonction de *fitness* ainsi que les nouveaux concepts associés à l'hommilière ont été «mis en production» depuis plusieurs mois.

La figure 6 présente l'évolution de la *fitness* des arcs sortant d'un item particulier en fonction du nombre de passages. Suite à cet item, deux arcs ont été conseillés par les professeurs (les courbes en gras), l'un avec une note de 5, (courbe du haut) et l'autre avec une note de 2 (courbe du bas). Les courbes fines représentent des arcs créés par des élèves. L'arc en gras du dessus semble avoir bien été conseillé car son taux de succès actuel avoisine les 70%, alors que le celui du dessous présente un taux de succès de 33%.

Dans le cas de l'arc du dessous, une inversion se produit assez rapidement (valeur négative après 5 passages) alors que pour le premier arc, l'influence pédagogique décroît avec le nombre de passages jusqu'à ce que sa valeur de *fitness* rencontre celle des arcs trouvés par les élèves.

Lors des 15 dernières visites, le «mauvais» arc n'est plus parcouru et les phéromones ont tendance à s'éroder du fait du passage des élèves qui choisissent les autres arcs. Comme prévu dans la fonction de *fitness*, le poids pédagogique reprend alors

de l'importance et sa valeur augmente, ce qui accroît sa chance d'être sélectionné à nouveau.

Le système réagit comme prévu, mais moins vite que dans les cas-tests. En effet, dans les cas-tests, il faut à peu près 30 passages pour que les arcs des élèves arrivent au même niveau qu'un bon arc professeur alors que dans le cas réel, il en faut une cinquantaine (arc gras à 72%). Une explication est que contrairement aux cas-tests qui sont simples, les élèves créent de nombreux arcs alternatifs ce qui diminue l'effet des phéromones.

Enfin, après analyse de plusieurs graphes pédagogiques, l'algorithme mis en place semble posséder les caractéristiques souhaitées, c'est-à-dire stabilité (en fonction du passage des élèves), robustesse (capacité à compenser des «erreurs» dans les poids statiques ou à combler certains vides pédagogiques) et surtout adaptabilité (possibilité d'inversion en cas de mauvais conseil).

7.3. Système d'audit

De plus, la présence de taux de phéromones anormalement élevés sur certains arcs a permis de mettre en évidence des problèmes pédagogiques jusqu'à présent passés inaperçus (exercices mal posés, problèmes d'évaluation...) ce qui laisse à penser que le concept d'hommilière peut aussi servir d'outil d'audit puissant, car capable de vérifier naturellement le bon comportement d'un graphe de plusieurs milliers de nœuds interconnectés entre eux, tâche quasiment impossible autrement.

Dans le cadre du développement actuel des réseaux et des Technologies de l'Information et de la Communication, les exemples se multiplient où des milliers d'utilisateurs potentiels sont mis en relation via un logiciel particulier ou un service informatique. Les résultats de ce travail suggèrent que le développement conscient et planifié d'hommilières pour des besoins d'audit, pour la remontée d'information ou pour l'amélioration de systèmes complexes puisse être une voie de recherche prometteuse.

8. Comparaison avec l'existant

8.1. Description d'une hommilière du point de vue des ITS

Dans le domaine des *Intelligent Tutoring Systems* (ITS), le système de suggestion par hommilière se place dans la sous-catégorie de l'ordonnement d'items dont le but est de proposer la suite d'items la plus adaptée à l'élève. Dans le même ordre d'idée, il existe une branche de recherche qui s'appelle le filtrage collaboratif, qui permet d'obtenir automatiquement des prédictions (filtrage) à propos des centres d'intérêt d'un utilisateur en collectant des informations de goût à partir de plusieurs utilisateurs (collaboratif). Aux cours de ses travaux de recherche, Jon Dron (Dron *et al.*, 1999; Dron *et al.*, 2000; Dron, 2002) a analysé ces systèmes et a conclu à l'exis-

tence de stigmergie résultante de la collaboration entre les utilisateurs. Il a ensuite développé un système nommé CoFIND dont le but est de regrouper et organiser des ressources pédagogiques au moyen d'actions combinées de ses utilisateurs.

Le principal avantage du paradigme hommilère (tout comme le logiciel CoFIND) est l'utilisation des données stigmergiques pour guider les élèves dans les bonnes directions pédagogiques. Le paradigme hommilère présente, en plus de CoFIND, une possibilité d'individualisation qui permet d'offrir un parcours particulier à chaque utilisateur.

Néanmoins, un des atouts de CoFIND, que ne possède pas le système de Paraschool, est la communication entre les différents utilisateurs. D'après une théorie de Moore (Moore *et al.*, 1996), la structure et le dialogue doivent être les briques élémentaires de tout système pédagogique. Mais du côté de Paraschool, l'équipe pédagogique a souvent freiné sur ce point car elle juge que les outils de dialogue sont plutôt des vecteurs de bavardage que de discussions sérieuses.

8.2. Introduction de la notation ELO dans l'hommilère de Paraschool

Pour permettre au système de tenir compte encore plus finement du profil des élèves, on aimerait aussi pouvoir guider l'élève vers des exercices d'une difficulté correspondant à son niveau.

Une notation des élèves est mise au point, en s'inspirant de la notation ELO bien connue des joueurs d'échecs (Elo, 1878). Les items et les élèves sont perçus comme des adversaires qui s'affrontent, et le système donne des points à l'item et à l'élève en fonction du résultat attendu.

Le principe de la notation ELO est de calculer la probabilité de réussite de l'élève à un exercice. Le résultat effectif est ensuite comparé à cette probabilité afin de déterminer la quantité de points échangés entre les deux joueurs. Plus le résultat est conforme au résultat attendu, plus petite sera la quantité de points échangée. Pour résumer rapidement, un élève gagnera (ou perdra) beaucoup de points s'il gagne (ou perd) contre un item de niveau ELO très supérieur (très inférieur) au sien.

Ce score permet de noter aussi bien les élèves que les items, le tout sur une même base de notation. Un effet secondaire intéressant est que l'élève peut en permanence juger son niveau par rapport au niveau de l'exercice et par rapport aux autres élèves du système.

Il y a deux méthodes pour utiliser cette notation ELO :

- la première est de présenter à l'élève le niveau de l'exercice au moment du choix. On peut alors espérer détecter des profils d'élèves différents, distinguant ceux qui préfèrent se frotter aux exercices difficiles de ceux qui, au contraire, préféreront la facilité. L'information recueillie permettra de biaiser le système en fonction de chaque élève, et de proposer des exercices différents en fonction du profil élaboré,

– comme dans la notation ELO, il est aisé de calculer la probabilité de réussite de l'élève face à un exercice, la deuxième méthode d'intégration est d'utiliser cette probabilité afin de biaiser le parcours de l'élève pour lui proposer des exercices adaptés à son niveau.

En combinant ces deux méthodes dans le paradigme d'hommilière, on peut espérer obtenir en même temps le profil de l'élève tout en adaptant son parcours à son niveau. L'implémentation de la notation automatique ELO est venue fournir un résultat inattendu sur le fonctionnement de l'hommilière : en moyenne, on observe une différence de 200 points ELO entre l'élève et l'item visité. Lorsqu'on injecte cette valeur dans la formule calculant la probabilité de réussite, cela correspond à une chance de 60% pour l'élève de réussir l'item auquel il se confronte, ce qui semble montrer que l'hommilière fonctionne bien.

REMARQUE. — La notation ELO a été mise en production, mais de manière silencieuse (les scores ne sont pas affichés à l'élève). Les poids agenda et niveau n'ont pas encore été mis en production.

8.3. *L'hommilière Paraschool et les systèmes de personnalisation de parcours*

Le nouveau paradigme développé se situe entre les algorithmes d'OCF (optimisation d'un chemin global) et les algorithmes de détection de profil (optimisation de parcours pour chaque utilisateur). Dans le cadre de Paraschool, le profil d'un élève se fonde pour l'instant sur quatre notions : le niveau, son agenda (les items qu'il doit faire suite au conseil d'un professeur ou du système), son historique (les items faits) et sa combativité (face à la difficulté d'un item).

Certaines approches se fondent sur du filtrage collaboratif de profil utilisateur (GroupLens (Konstan *et al.*, 1997; Herlocker *et al.*, 1999), Firefly (Shardanand *et al.*, 1995), SUGGEST (Baraglia *et al.*, 2004)). Pour éviter l'aspect subjectif, un certain nombre d'approches ont été développées en utilisant des techniques de fouille de données Web afin de découvrir automatiquement les profils utilisateurs. Par exemple Schechter *et al.* (Schechter *et al.*, 1998) ont développé des techniques pour prédire les futures requêtes de l'utilisateur en analysant les requêtes antérieures afin d'optimiser les mises en cache. Landrin-Schweitzer *et al.* (Landrin-Schweitzer *et al.*, 2003) ont utilisé la programmation génétique pour récrire les requêtes de l'utilisateur à partir d'un profil évolué par programmation génétique. Des colonies de fourmis artificielles (Labroche *et al.*, 2003a) ont été utilisées pour orienter les utilisateurs sur le parcours d'un site web.

Dans les algorithmes de détection de profil et donc de personnalisation de parcours, la notion d'ordre n'est souvent pas prise en compte, perdant ainsi toute notion de chemin. Cependant les schémas de navigation séquentiels (Spiliopoulou *et al.*, 1999) permettent d'améliorer l'optimisation de site surtout en ce qui concerne la personnalisation du site ou la gestion des flux sur le site. Le paradigme hommilière

possède par essence cette notion de chemin, permettant ainsi d'espérer une meilleure optimisation des parcours.

Dans un processus de personnalisation, un des inconvénients majeurs du paradigme est qu'il est difficile de juger de la satisfaction individuelle. Il faudrait donc mettre en place un système de vote qui permettrait d'évaluer cette adéquation (comme dans le système CoFIND). Le vote pourrait en plus influencer le dépôt de phéromones, ce qui orienterait les utilisateurs vers les éléments les plus satisfaisants, mais le système serait moins transparent.

9. Conclusion

L'adaptation du paradigme de l'OCF aux élèves de la société Paraschool a montré qu'on ne pouvait pas utiliser des êtres humains comme des fourmis : le principe d'évaporation des phéromones est inadapté (contrairement au principe d'érosion introduit), et il faut modifier le modèle pour tenir compte du besoin d'individualité des élèves (absent chez les fourmis), en introduisant des phéromones individuelles multiplicatives pour que le système offre à chaque élève un parcours personnalisé.

Dans le cadre des EIAH, ce paradigme d'optimisation permet d'apporter stabilité, robustesse, adaptabilité et personnalisation de la navigation. Avec l'évaluation automatique du niveau des élèves et des exercices par notation ELO, le système est très complet et réagit sagement, depuis sa mise en production en octobre 2004.

Ce travail (décrit de manière bien plus complète dans (Valigiani, 2006)) débouche donc sur l'adaptation de l'OCF à l'être humain pour créer le concept d'optimisation par hommilière. Les retombées en enseignement assisté par ordinateur sont très importantes, car le modèle est remarquablement bien adapté à cette tâche. Mais il est possible d'entrevoir de nombreuses autres applications (en audit ou en optimisation) dans d'autres domaines, du fait du développement rapide de sites web où naviguent des milliers d'internautes.

10. Bibliographie

- Balacheff N., Baron M., Desmoulins C., Grandbastien M., Vivet M., « Conception d'Environnements Interactifs d'Apprentissage avec Ordinateur. Tendances et perspectives », *Actes des journées nationales du PRC IA*, p. 315-338, 1997.
- Baraglia R., Silvestri F., « An Online Recommender System for Large Web Sites », *IEEE/WIC/ACM International Conference on Web Intelligence*, 2004.
- Bonabeau E., Dorigo M., Theraulaz G., « Swarm Intelligence : From natural to Artificial systems », 1999.
- Bonabeau E., Dorigo M., Theraulaz G., « Inspiration for optimization from social insect behaviour », *Nature*, vol. 406, p. 39-42, 2000.
- Colomi A., Dorigo M., Maniezzo V., « Distributed optimization by ant-colonies », *European Conference on Artificial Life*, MIT Press, Cambridge, p. 134-142, 1991.

- Crozat S., *Eléments pour la conception industrialisée des supports pédagogiques numériques*, PhD thesis, Université de Compiègne, 2001.
- de la Passardière B., Giroire H., « XML au service des applications pédagogiques », *Actes des journées EIAO'2001, Revue Sciences et Techniques Educatives*, vol. 8, p. 99-112, 2001.
- Deneubourg J., Aron S., Goss S., Pasteels J., « The self-organizing exploratory pattern of the Argentine ant », *Insect Behaviour*, vol. 3, p. 159-168, 1990.
- Deneubourg J., Pasteels J., Verhaeghe J., « Probabilistic behaviour in ants : a strategy of errors ? », *Theoretical Biology*, vol. 105, p. 259-271, 1983.
- Després C., *Suivi synchrone d'activités d'apprentissage à distance*, PhD thesis, Université du Maine, Lium, 2001.
- Dorigo M., *Optimization, learning and natural algorithms*, PhD thesis, politecnico di Milano, 1992.
- Dorigo M., Caro G. D., « The ant colony optimization metaheuristic », *New ideas in optimization*, 11-32, 1997.
- Dron J., *Achieving self-organisation in network-based learning environments*, PhD thesis, Brighton University, 2002.
- Dron J., Mitchell R., Boyne C., Siviter P., « Website Evaluation System : Collaboratively Discovering what makes a web site good », in G. Davies, C. Owen (eds), *WebNet 2000, AACE*, p. 685-686, 2000.
- Dron J., Mitchell R., Siviter P., Boyne C., « CoFIND - an experiment in n-dimensional collaborative filtering », in P. de Bra, J. Legget (eds), *WebNet'99, AACE*, p. 301-306, 1999.
- Elo A. E., « The rating of chessplayers, Past and Present », 1878.
- George S., *Apprentissage collectif à distance, SPLACH : un environnement informatique support d'une pédagogie de projet*, PhD thesis, Université du Maine, Lium, 2001.
- Grandbastien M., « Quelques questions à propos de l'indexation et de la recherche de ressources pédagogiques sur le Web », in B. G.-L. et Bruillard E. (ed.), *Les technologies en éducation : Perspectives de recherche et questions vives*, p. 211-220, 2002.
- Herlocker J., Konstan J., Borchers A., Riedl J., « An algorithmic framework for performing collaborative filtering », *Proceedings of the 1999 Conference on Research and Development in Information Retrieval*, 1999.
- Kennedy J., Eberhart R. C., « Swarm Intelligence », 2001.
- Konstan J., Miller B., Maltz D., Herlocker J., Gordon L., Riedl J., « GroupLens : applying collaborative filtering to usenet news », *Communications of the ACM*, vol. 40, 1997.
- Labroche N., Monmarché N., Venturini G., « AntClust : Ant Clustering and Web Usage Mining », in E. Cantu-Paz (ed.), *Genetic and Evolutionary Computation Conference, Lecture Notes in computer science 2723*, Springer Verlag Telos, p. 25-36, 2003a.
- Labroche N., Monmarché N., Venturini G., « AntClust : Ant Clustering and Web Usage Mining », in E. Cantu-Paz (ed.), *Genetic and Evolutionary Computation Conference Proceedings*, vol. 2723, Springer-Verlag Telos, Chigago, p. 25-36, July, 2003b.
- Landrin-Schweitzer Y., Collet P., Lutton E., « Interactive GP for data retrieval in medical databases », *proceedings of EuroGP 2003*, Springer Verlag, 2003.
- Mizoguchi R., Murray T., « Workshop on Ontologies for intelligent educational systems », *AI-Ed Conference*, 1999.

- Moore M. G., Kearsley G., « Distance Education : a Systems View », Wadsworth Publishing, 1996.
- Moyson F., Manderick B., « The Collective Behaviour of Ants : an Example of Self-Organisation in Massive Parallelism », *AAAI Spring Symposium on Parallel Models of Intelligence*, Stanford, California, 1988.
- Paquette G., « Construction de portails de télé-apprentissage Explor@ - Une diversité de modèles pédagogiques », *Revue Sciences et Techniques Educatives*, vol. 7, p. 207-226, 2000.
- Schechter S., Krishnan M., Smith M. D., « Using path profiles to predict HTTP requests », *Proceedings of 7th International World Wide Web Conference*, 1998.
- Shardanand U., Maes P., « Social information filtering : algorithms for automating “word of mouth” », *Proceedings of the ACM CHI Conference*, 1995.
- Spiliopoulou M., Faulstich L. C., « WUM : A Web Utilization Miner », *Proceedings of EDBT Workshop WebDB98*, Springer Verlag, 1999.
- Valigiani G., Développement d'un paradigme d'Optimisation par Hommilière et application à l'Enseignement Assisté par Ordinateur sur Internet, PhD thesis, Université du Littoral Côte d'Opale, 2006.

Article reçu le 21 mars 2005

Accepté après révisions le 22 novembre 2006

Grégory Valigiani vient de finir son doctorat en informatique au Laboratoire LIL de l'université du Littoral (Calais). Ses travaux portent sur l'évolution artificielle, notamment l'application du paradigme d'optimisation par colonie de fourmis à des problèmes concrets. Il s'intéresse plus particulièrement au nouveau paradigme d'optimisation par Hommilière et son adaptation au domaine de l'e-learning.

Evelyne Lutton est chercheur à l'INRIA, et responsable scientifique de l'équipe APIS. Ses travaux concernent l'analyse et la modélisation de signaux et de processus complexes à l'aide de techniques issues de la géométrie fractale et l'évolution artificielle. En particulier, elle s'intéresse à l'irrégularité produite par des processus élémentaires simples (croissance, cumulation, évolution) et aboutissant à des processus globaux complexes.

Cyril Fonlupt est professeur des universités à l'université du littoral - Côte d'Opale et responsable de l'équipe informatique évolutionnaire. Ses travaux portent sur l'évolution artificielle. Il s'intéresse plus particulièrement aux techniques de programmation automatique en évolution artificielle comme la programmation génétique.

Pierre Collet est professeur en informatique, au laboratoire des sciences de l'image, de l'informatique et de la télédétection de l'université Louis Pasteur à Strasbourg et chef de l'équipe FDBT (fouille de données et optimisation, bioinformatique théorique). Ses travaux de recherche portent sur l'optimisation de problèmes complexes par algorithmes stochastiques inspirés de la nature. Il est notamment spécialiste en programmation génétique, algorithmes évolutionnaires, algorithmes évolutionnaires interactifs, optimisation par colonies de fourmis. . .