



Prototypage d'un agent émotionnel d'interface pour les laboratoires virtuels

Yan Laporte, Roger Nkambou, Roland Yatchou

► **To cite this version:**

Yan Laporte, Roger Nkambou, Roland Yatchou. Prototypage d'un agent émotionnel d'interface pour les laboratoires virtuels. Technologies de l'Information et de la Connaissance dans l'Enseignement Supérieur et l'Industrie, Oct 2004, Compiègne, France. pp.198-204. edutice-00000710

HAL Id: edutice-00000710

<https://edutice.archives-ouvertes.fr/edutice-00000710>

Submitted on 16 Nov 2004

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Prototypage d'un agent émotionnel d'interface pour les laboratoires virtuels

Yan LAPORTE, Roger NKAMBOU, Roland YATCHOU

Laboratoire de Gestion, Diffusion et Acquisition des Connaissances
Université du Québec à Montréal, Canada ; Case postale 8888, succursale Centre-ville
Montréal (Québec) Canada – H3C 3P8

Tél. (514) 987-3000 ext. 8395 ; Téléc. (514) 987-8477

ylaporte @acm.org, roger.nkambou@uqam.ca, ryatchou@yahoo.fr

Résumé

Plusieurs logiciels éducatifs mettent en scène des êtres virtuels qui accompagnent l'apprenant dans son cheminement d'apprentissage. Ces assistants servent d'interface de communication par laquelle le système donne des directives à l'apprenant. Malheureusement, leurs mouvements sont construits sur un synopsis figé où, toutes les actions sont prévues d'avance. Dans cet article, nous présentons une approche qui enrichie la personnalité des agents d'interface par une gamme de comportements dynamiques en rapport avec les réalisations de chaque apprenant. Ce faisant, nous les dissocions du contenu et nous les dotons d'une couche émotionnelle, ce qui les rend réutilisables.

Mots-clés : Aide intelligente, Modèle cognitif, Réalité virtuelle.

Abstract

Several educational software implement virtual being to guide the learner in his training progress. These assistants serve as communication interface by which the system gives instructions to the learner. Unfortunately, their movements are generally built on static synopsis where all actions are determined in advance. In this article, we present an approach enriching the interface agent character with a range of dynamic behaviors displayed matching each learner accomplishment. Doing that, we dissociate them from the content and we endow them with an emotional layer, what makes them reusable.

Keywords: Intelligent assistant, Cognitive model, Virtual reality.

Introduction

Le rapprochement de l'industrie du jeu et du divertissement à celle des ressources éducatives a donné lieu à un ensemble de produits utilisant des animations, de la musique et des interactions pour permettre d'apprendre tout en s'amusant. Ces produits s'adressent parfois aux adultes, notamment pour l'apprentissage des langues mais surtout aux enfants pour lesquels ils servent de complément ou de renforcement à la formation scolaire. Plusieurs de ces produits mettent en scène des êtres virtuels qui accompagnent l'apprenant dans son cheminement d'apprentissage. Ces personnages servent d'interface

de communication par laquelle le système donne à l'apprenant des directives, des explications ainsi qu'une appréciation du travail accompli. Le comportement de ces personnages est généralement construit suivant un synopsis figé où toutes les actions (animations et paroles) sont prévues d'avance et « scriptées ».

L'appréciation donnée par ces créatures logicielles se limite soit à une réponse positive suite au bon achèvement d'une tâche ou alors à une réponse négative bien qu'encourageante en cas d'erreur. Dans certains cas, des conseils s'ajouteront lorsque l'apprenant ne parviendra pas à résoudre le problème. La réaction du personnage n'est pas en relation étroite avec le résultat vraiment réalisé par l'utilisateur. Par exemple, si l'apprenant en est à la vingtième fois où il réussit un même exercice, le personnage ne le félicitera pas moins chaudement alors que l'accomplissement réel est beaucoup moins grand qu'il ne l'a été la première fois. De même, si l'apprenant réussit à résoudre une situation après de multiples essais, son mérite devrait être moindre que s'il y était arrivé en un ou deux essais. C'est encore plus vrai dans le cas où l'apprenant a épuisé toutes les possibilités avant de réussir. Le but recherché est donc de faire que la réponse donnée à l'apprenant soit ajustée à sa performance.

Un des aspects importants des agents d'interface évoluant dans un système tutoriel est lié à leur comportement en réponse aux actions de l'étudiant. En effet, même si l'agent fait un reproche où est désolé d'un échec, une fois le commentaire passé il revient à sa posture initiale comme si rien ne s'était passé et son enthousiasme n'en est pas moins différent après un succès survenant au bout de nombreuses erreurs. De même, quand l'apprenant réussit à résoudre un problème alors qu'il était particulièrement probable qu'il ne réussira pas du premier coup, l'agent le félicite puis reprend son état premier et soulignera ses erreurs sans tenir compte de l'état précédent. Le deuxième aspect recherché est donc que l'attitude de l'agent soit constante et que les signes de son appréciation soient conséquents avec le contexte des interactions précédentes.

Finalement, chaque réponse positive ou négative donnée par l'agent est explicitement prévue, conçue dessinée et animée. Si on veut étoffer la quantité de réponses possibles ou augmenter le degré de

raffinement des réponses données à l'utilisateur, on doit multiplier le nombre de réponses et donc des efforts de production. Également, chaque réponse est habituellement attachée à un événement précis, par exemple : le fait de choisir la bonne image entraîne nécessairement la présentation d'une des différentes animations de félicitations disponibles. Ce lien doit avoir été spécifiquement identifié au moment de la conception du produit. Un autre aspect recherché est de limiter les efforts requis pour accroître la gamme des comportements exprimés par l'agent et que ceux-ci n'aient pas à être spécifiquement attachés à un événement de l'environnement d'apprentissage.

Dans cet article, nous présentons une approche permettant d'enrichir la gamme des comportements affichés par un personnage accompagnant l'apprenant dans un logiciel éducatif multimédia, afin de rendre ce personnage plus crédible. A cet effet, nous nous proposons d'introduire une couche intermédiaire entre l'environnement d'apprentissage et la représentation du personnage animé. Autrement dit, nous insérerons quelque chose qui fera le lien entre ce qui se passe dans l'environnement d'apprentissage et l'animation du personnage, tout en les gardant indépendants l'un de l'autre. Cette abstraction devra permettre :

- d'adapter le comportement de l'agent en fonction de l'accomplissement actuel et du problème à résoudre,
- de rendre le comportement de l'agent plus constant et éviter qu'il ne passe d'un extrême à l'autre.

Étant donné qu'un des objectifs visés est d'augmenter la crédibilité et la richesse du comportement de l'agent, nous explorerons les possibilités qu'offrent les modèles utilisés pour représenter les émotions humaines.

Nous présenterons tout d'abord un environnement d'apprentissage où un tel agent évoluera. Ensuite, nous examinerons plusieurs modèles disponibles pour simuler ou représenter les émotions. Nous présenterons comment certains de ces modèles ont déjà été utilisés dans le contexte de systèmes informatisés. Nous proposerons ensuite une architecture capable d'utiliser un modèle émotionnel dans le fonctionnement d'un agent d'interface. Nous présenterons et discuterons par la suite du prototype réalisé selon cette approche et nous terminerons par la suggestion des améliorations possibles pour un deuxième prototype.

Cyberscience : une Plateforme de Laboratoires Virtuels

Le cadre de travail ciblé par notre démarche est un système tutoriel intelligent comprenant une collection de laboratoires virtuels où les apprenants doivent accomplir des tâches et résoudre des problèmes dans des simulations interactives. Ce système est nommé Cyberscience.

Les usagers du système sont des étudiants de premier cycle universitaire et les activités proposées sont reliées au domaine d'étude d'un cours auquel ils sont inscrits. L'accent est mis sur la simulation d'équipements de laboratoires avec pour objectif de

permettre aux étudiants d'accomplir des tâches qui leur demanderaient normalement l'accès à un vrai laboratoire. Il est donc possible de permettre aux étudiants d'expérimenter et de manipuler les équipements en évitant les problèmes liés à la disponibilité de matériel souvent coûteux, l'achalandage des laboratoires ou alors le manque de locaux.

Cyberscience inclut des agents intelligents se voulant génériques ou adaptables aux différents micro-mondes. Ceux-ci sont des cadres d'apprentissage spécialisé où l'apprenant peut explorer et expérimenter le domaine d'apprentissage et y résoudre des problèmes. Finalement, c'est un espace qui permet d'interagir avec différents « laboratoires virtuels » qui constituent des micro-mondes qui s'insèrent à la demande dans la plateforme. Les activités du système sont des simulations en 3D d'équipements de laboratoire dans des interfaces graphiques conformes permettant de résoudre des problèmes logiques. Les informations du modèle de l'utilisateur sont aussi utilisées dans le raisonnement. Celui-ci se compose de trois parties : le modèle cognitif, le modèle affectif et un moteur d'inférence. Le détail de ces modèles de représentation du curriculum est élaboré en se basant sur des outils précédents sous l'appellation CREAM [1 ; 2].

Les agents intelligents implémentés n'ont pas nécessairement de présence physique contrairement aux agents d'interface ou aux agents personnifiés qui sont des personnages apparaissant dans l'interface utilisateur. L'étendue du monde visible des agents intelligents dans Cyberscience est limitée à la consultation de la documentation et à l'exécution des tâches requises dans une activité de laboratoire. Ils n'ont pas conscience des autres applications fonctionnant sur le poste de travail. Un aspect récurrent dans les systèmes de formation intelligents est la présence d'un ou plusieurs agents qui tracent les agissements de l'utilisateur. Certains ont pour rôle d'assister, d'encadrer (« coaching »), de donner une réaction ou d'évaluer les agissements de l'utilisateur. Notre objectif est de produire une entité visuelle émotionnelle, contrôlée par plusieurs processus logiciels, capable à la fois d'encadrer, d'évaluer et de réagir au travail de l'utilisateur.

Emotions et Applications à l'Apprentissage

Étude des Modèles d'Emotions

De nombreux modèles pour la production d'états émotionnels ont été énoncés ces dernières années. Leur représentation à l'intérieur d'agents virtuels a introduit de nouvelles caractéristiques et se conforment à deux orientations majeures soit le paradigme de Minsky [3] et la théorie des émotions de Ortony Clore et Collins [4]. Nous examinerons ici comment les modules responsables de la génération des émotions ont été réalisés.

- *Le modèle du Fungus eater.* Bien que n'ayant pas débouché sur un modèle utilisable pour la

conception d'un système capable de simuler les émotions d'un agent, la théorie du « Fungus eater » (littéralement : mangeur de moisissure) élaborée par le psychologue Masanao Toda [5] a introduit l'idée de modéliser le comportement émotionnel d'individus. Cette théorie s'articule autour d'une série d'expériences où les sujets humains étaient engagés dans un jeu où ils prenaient le rôle d'un robot-créature aux buts, perceptions et capacités simples et limitées. Un des objectifs de ces expériences était de montrer l'utilité des émotions à la survie de robots dans un environnement non contrôlé.

- *Le modèle OCC.* Ce modèle développé par Ortony, Clore et Collins (ou OCC) [4] visait précisément à fournir un modèle susceptible d'être implanté à l'aide de l'ordinateur. Il peut donc être facilement exprimé dans un langage de programmation. OCC représente les émotions comme étant la conséquence de certains processus cognitifs. Il est donc principalement constitué par la description des différents processus cognitifs à la base des émotions. Ceux-ci sont regroupés autour de trois sujets : des événements (et leurs conséquences), des actions des intervenants et des objets. Les émotions sont des réactions différentes à chacun de ces sujets et l'intensité variera en fonction du désir, du mérite et de l'attraction.
- *Le modèle de Roseman.* Le modèle de Roseman a été développé en se basant sur une série de rapports d'expérience sur les émotions [6]. Ce modèle comporte un ensemble de cinq dimensions cognitives. La première indique si le sujet a une motivation positive ou négative concernant une situation particulière. La seconde signale si une situation correspond ou non aux objectifs du sujet. La troisième décrit si un événement est considéré comme une certitude ou une simple possibilité. La quatrième indique si le sujet perçoit un événement comme mérité ou non. Finalement, la cinquième dimension décrit la source de l'événement avec trois valeurs sont possibles : les circonstances, les autres, ou soi-même. Ce modèle n'a pas permis d'obtenir les résultats escomptés par Roseman. En conséquence, il modifiera sa théorie à plusieurs reprises [7 ; 8] dans le but de le rendre plus représentatif [9].
- *Le modèle Cathexis.* Dans son approche, Velasquez présente le dynamisme naturel des phénomènes liés aux émotions tels que l'humeur et le tempérament autour de 4 sous-systèmes : le sous-système comportemental, le sous-système de la génération d'émotion, le sous-système moteur et le sous-système de stimuli interne [10]. Ces sous-systèmes sont exposés à des stimuli externes provenant de l'environnement et le système réagira par des actions sur cet environnement. Cathexis met une emphase particulière sur l'intensité de l'émotion générée. Chaque proto spécialiste est caractérisé par deux seuils d'intensité. Le premier nommé alpha, correspond au seuil où l'activation

du proto spécialiste commence, le second nommé oméga correspond au niveau de saturation.

Systèmes d'Apprentissage Générant des Emotions

Sur la base des modèles décrits, plusieurs systèmes à base d'émotions ont été réalisés. Nous présentons quelque uns de ceux qui ont influencé d'une façon quelconque nos réalisations.

- *Oz.* L'approche employée par Bates a été de construire des agents qui regroupent un ensemble de capacités et qui prennent en compte un grand nombre de facteurs sans exploiter aucun de ceux-ci de manière très approfondie. L'objectif visé par la réalisation de Oz [11] était de construire un ensemble de personnages ayant une apparence la plus vivante et autonome possible. Il constitue un monde virtuel qui se déroule comme une pièce de théâtre pour l'utilisateur qui ne tient qu'un rôle d'observateur. Les capacités de ces agents sont intégrées dans une architecture nommée Tok [12] qui regroupe un ensemble de modules réalisant des fonctions de reconnaissance et de génération de langage naturel, de mémoire, de prise de décision par rapport à un objectif, de comportement social, d'une base de connaissances sociales, etc.
- *The affective Reasoner.* The affective reasoner [13] suit la voie d'un environnement simulé dans lequel plusieurs agents évoluent et interagissent. L'objectif poursuivi par Elliot est de permettre l'étude de certaines théories de psychologie sur l'émergence des émotions ainsi que des actions quelles motivent, l'utilisation de modèles affectifs au fonctionnement de systèmes d'agents distribués ainsi que d'aider au développement d'un modèle informatique capable de comprendre les émotions pour éventuellement améliorer la qualité des interfaces personne/machine. La représentation des émotions développée par Elliot se base sur le modèle OCC. Cette approche qui ne gère pas convenablement les émotions conflictuelles a connu un certain nombre d'amélioration et a été utilisé dans de nombreux contextes [14 ; 15]. « The affective reasoner a été intégré dans plusieurs systèmes tutoriels intelligents. Son intégration dans STEVE (Soar Training Expert for Virtual Environments) visait à explorer l'application d'une personnalité et de capacité de réponse émotionnelle à un agent pédagogique. Son intégration au personnage de « Herman the bug » dans Design-à-plant [16] visait plutôt à explorer la reconnaissance et l'interprétation de l'état émotionnel de l'utilisateur.
- *Émile.* La construction d'Émile [17 ; 18] s'est inspirée de Em et de « The affective reasoner ». En effet, Émile est également basé sur la théorie de Ortony, Clore et Collins (OCC). Cependant, l'objectif de Gratch dans la construction d'Émile était plutôt de se concentrer sur la relation qui existe entre les émotions, la prise de décision et la sélection d'actions particulières. La contribution

majeure d'Émile est de décrire le raisonnement émotionnel dans le contexte d'autres formes de raisonnement, en particulier la planification. Émile a été implanté pour permettre une réponse plus réaliste des agents qui interagissent dans un champ de bataille simulé [19]. Il a également été implanté dans STEVE, où le modèle de planification inclus dans Émile viens compléter le modèle de perception, les commandes motrices, l'interface en langage naturel et la représentation physique existante [20].

Architecture de l'Agent Emotionnel d'Interface

L'architecture que nous proposons pour isoler le comportement d'un agent d'interface du domaine d'application s'appuie sur une structuration en couches ayant des responsabilités distinctes. A cet effet, nous lui intégrons un modèle d'émotions. Cette séparation est illustrée dans la figure 1 où les parties blanches sont celles qui assurent la gestion de l'agent et qui interviennent donc dans le traitement des émotions alors que les parties du haut en gris foncé sont celles qui touchent le domaine d'application. Le passage d'un micro-monde ou d'un domaine d'application à un autre revient donc à remplacer les deux premières couches.

La structure illustrée se présente comme un pipeline faisant le lien entre une action issue d'une manipulation de l'apprenant dans l'interface du micro-monde à l'apparence et au comportement de l'agent d'interface. À la base de ce processus se trouve l'action accomplie par l'utilisateur dans l'interface, cette action n'est pas nécessairement une erreur ou la complétion d'une tâche, les actions sont constituées de toute manipulation élémentaire de l'apprenant dans l'interface.

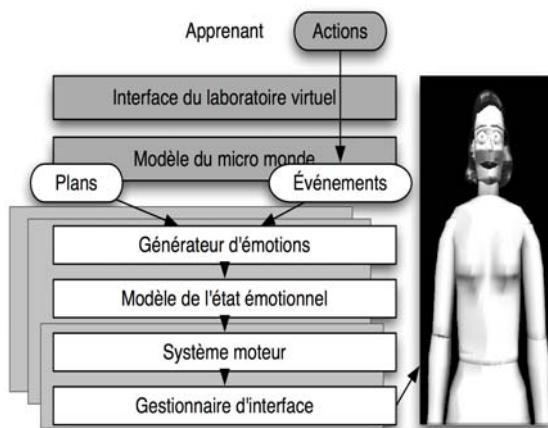


Figure 1. Architecture générale de Émile

Paramètres du Système Emotif

Étant donné que les systèmes tutoriels intelligents tendent à produire une grande quantité d'information sous diverses formes et sur différents aspects de son

opération, il est nécessaire d'identifier quels sous-ensembles d'information seront utilisés pour générer les émotions de l'agent. Bien que l'approche utilisée par Gratch avec l'utilisation de plans pour extraire des émotions soit claire, élégante et efficace, elle semble inappropriée dans le contexte d'un agent jouant le rôle de « coach » à la manière d'Émile. Nous avons donc recherché une méthode qui permettrait de générer des émotions à partir d'informations ponctuelles sans avoir à évaluer celles-ci dans un contexte.

Nous avons identifié un petit nombre d'informations d'intérêt pour le rôle de notre agent limitant ainsi la nécessité de connaissances liées au domaine d'application tout en permettant un niveau suffisant de crédibilité aux réactions de l'agent. Celles-ci ont été ramenées à deux paramètres : le degré de difficulté de l'action accomplie (où à accomplir) et le degré d'erreur de l'étudiant ou plutôt une appréciation de l'écart entre l'action de l'utilisateur et celle attendue. Le plan reçu par la couche de génération des émotions consiste en une liste de facteurs de difficulté qui devraient être rencontrés par l'utilisateur avant d'avoir complété l'activité avec succès. Cette information est utilisée pour produire les émotions d'Espoir, Peur, Satisfaction, Peur-Confirmée et Déception telles que définies par le modèle de l'état émotionnel.

Modulation du Niveau de Difficulté

Bien qu'elles ne soient pas mises à contribution lors de la génération des émotions et donc ne fassent pas partie des données utilisées par l'agent Émile, les informations provenant du modèle de l'utilisateur interviennent tout de même dans le processus de production des émotions. Le modèle de l'utilisateur utilisé contient l'information sur les concepts maîtrisés par l'utilisateur. Dans le curriculum par contre, on retrouve les prérequis et les concepts associés à une activité et, idéalement, une structure similaire pour chaque étape intermédiaire de l'activité.

Il est envisageable que des modules intermédiaires pourraient être introduits pour l'évaluation du degré de difficulté pour différents types d'activités. Ces modules ou agents seraient actifs au niveau de la couche du modèle du micro-monde qui pourrait ainsi être constitué d'une architecture tableau noir («Black Board») [21 ; 22] où chaque agent ou module apporterait sa contribution à l'interprétation de ce qui se passe dans le micro-monde.

Modélisation des Etats Emotionnels

Le modèle utilisé pour représenter l'état émotionnel dans Émile est fortement inspiré de celui proposé par Ortony Clore et Collins. Ce modèle a été préféré car il identifie facilement un grand nombre d'émotions et spécifie clairement la signification de celles-ci ainsi que la définition de chaque émotion telle que décrite par Elliot. De plus, l'état émotionnel à un moment donné peut être exprimé par une série de valeurs attribuées à chacun de ces types d'émotions.

Cette façon de représenter les émotions est particulièrement simple à implanter dans un système informatique et permet surtout de facilement spécifier la personnalité de l'agent. De plus, toutes les informations sur l'état émotionnel peuvent être centralisées à un seul endroit dans le système comparativement aux approches basées sur la contribution de plusieurs agents proto-spécialistes telle que celle de Velasquez [23] qui répartit les informations sur l'état émotionnel courant sur plusieurs agents.

Joy /Distress	Happy-for /Resentment	Sorry-for /Gloating	Hope /Fear
Satisfaction/ Disappointment	Relief/ Fears- Confirmed	Admiration /Reproach	Pride /Shame
Liking /Disliking	Gratitude /Anger	Gratification /Remorse	Love /Hate
Jealousy /-Jealousy			

Table 1. Les couples d'émotions contradictoires

La couche de l'architecture du modèle de l'état émotionnel est donc constituée d'un ensemble de couples d'émotions opposées (illustrés à la Table 1) auxquelles on assigne une valeur qui varie du positif au négatif où le pôle positif représente une des deux émotions alors que le pôle négatif représente l'émotion opposée. La modification de la valeur attribuée à une émotion modifie automatiquement la valeur associée à l'émotion opposée.

Génération des Emotions

La génération des émotions consiste en un ensemble de relations qui définissent comment les plans, les actions passées et l'événement courant doivent affecter les variations de l'état émotionnel de l'agent. Par exemple, une relation est définie entre un événement d'action positive et la montée de l'émotion « happy-for », cette relation indique également que l'importance de cette variation de l'émotion « happy-for » est directement dépendante de la valeur de la difficulté associée à cette action positive. Le fait que les relations spécifiées entraînent une variation de l'émotion plutôt que son assignation à un niveau spécifique permet à l'agent de conserver une attitude constante.

Les relations définies entre les événements et les émotions sont aussi affectées par le niveau de difficulté des actions réalisées précédemment. De plus, les plans permettent de comparer ce degré de difficulté à celui des actions à venir. Dans le même exemple, des émotions de satisfaction ou de soulagement sont plus élevées si l'étape courante est considérée plus difficile à accomplir que les actions à venir dans l'activité.

La Conversion des Emotions en Feedback Visuel

Une fois qu'un état émotionnel est disponible pour un agent, il constitue une nouvelle source d'information sur comment l'agent devrait interagir avec son

environnement. L'information sur les émotions peut influencer des décisions, introduire des variations dans la langue parlée, la musique, les couleurs ou encore plusieurs autres applications à venir. La couche du système moteur pour sa part s'occupe de définir le comportement d'un personnage en fonction de cet état émotionnel. Dans ce cas, des relations sont définies entre différentes émotions à différents niveaux et différentes caractéristiques du personnage. Autrement dit, on peut spécifier une relation entre le sourire du personnage et une combinaison des niveaux attribués aux émotions « Joy », « happy-for », « gratitude », « gratification » et « pride ». Le système moteur n'a par ailleurs aucune notion des dimensions de la bouche, de la géométrie du déplacement à appliquer sur certaines parties des lèvres du personnage, de la déformation des joues qui peut être nécessaire, etc. Il existe un certain nombre de standards [24] pour la spécification d'animation d'avatars d'apparence humaine, certains permettent même de décrire l'apparence et l'animation de formes non humaines. Ces standards sont souvent très récents et connaissent très peu d'implantations répandues.

Mise en Oeuvre de l'Agent Emotionnel

Le prototype de l'agent émotionnel nommé « Emilie » a été réalisé de manière à être utilisé par les laboratoires virtuels dans Cyberscience. Tout comme Cyberscience, ce prototype a été construit en utilisant le langage le Java Expert System Shell (JESS) [25]. JESS est une librairie qui permet de spécifier le comportement d'un logiciel sous forme de règles qui sont exécutées sur une base de faits.

– L'état émotionnel est spécifié comme une série de faits ou d'assertions qui comprennent une partie haute et une partie basse. Ces deux parties correspondent à deux émotions opposées selon le modèle défini dans l'architecture générale.

– Le générateur d'émotions et le système moteur comportent des faits et des règles. Ils sont tous deux spécifiés dans le même fichier bien qu'une séparation logique existe entre eux. Les règles du générateur d'émotions sont toujours spécifiées en utilisant l'événement courant dans le déclencheur. Chaque règle induit une variation de l'état émotionnel, jamais une valeur arbitraire, ainsi l'effet de plusieurs règles déclenchées par un même événement s'additionne. L'encadré ci-dessous présente l'exemple d'une règle qui induit l'émotion de soulagement dans le cas où l'apprenant réussit à un problème difficile après plusieurs échecs.

```
(defrule soulagementPassageDifficile
  (reussite ?difficulte :(> ?difficulte *FACILE*))
  (erreurs (scope local) (nombre ?nbErr :(> ?nbErr 0))
  (bonnesActions (scope local)
    (nombre ?nbsucces : (= ?nbSuccess 0)))
  => (changeEmotion Relief (* ?difficulte ?nbErr))
      (changeEmotion Fearr -(* ?difficulte ?nbErr))
  )
```

– Le gestionnaire d'interface du prototype Émilie gère un personnage en 3D qui est animé dans une fenêtre séparée qui flotte au dessus de la fenêtre de l'interface de l'apprentissage. Idéalement, le personnage aurait pu être dessiné directement dans la fenêtre principale de l'environnement d'apprentissage de manière à flotter au dessus de l'interface utilisateur mais des contraintes techniques ont rendu cette possibilité trop complexe. Le gestionnaire est construit en Java et le personnage 3D est rendu à l'aide de la librairie d'extension Java3D qui utilise des fonctionnalités d'accélération matérielles.

– Les règles du système moteur sont déclenchées en réponse aux variations des émotions (changement de l'état émotionnel). Un premier ensemble de règles fixe l'expression de base du personnage en fonction de son état émotionnel. Des règles d'une deuxième catégorie peuvent être déclenchées par des variations importantes d'émotion pour permettre l'exécution par le personnage de certaines mimiques et/ou de certains gestes. L'encadré ci-dessous montre un exemple de règle de cette deuxième catégorie. Cette règle est activée par le système moteur suite à la présence d'un changement au modèle émotionnel :

```
(defrule sourireDeBase
  (emotionChange Joy|Happy-for ?y )
  (emotion (higher Joy) (rate ?j )
  (emotion (higher Happy-For) (rate ?hf))
  =>
  (call ?*lecorps* setSourireDroit
    (max -100 (min 100 (+ ?j ?hf ?y)))
  (call ?*lecorps* setSourireGauche
    (max -100 (min 100 (+ ?j ?hf ?y)))
  (assert (bouche))
  )
)
```

Le système moteur a directement accès au modèle émotionnel. La communication avec le gestionnaire d'interface se fait via un mécanisme d'échange permet de partager la référence à un objet entre l'espace JESS et l'espace Java normal. En réponse aux règles du système moteur, le gestionnaire d'interface configure et ajuste les géométries nécessaires à l'affichage du personnage

La figure 2 montre le résultat des changements émotionnels dus à la complétion avec succès d'une activité évaluée comme difficile. L'expression du visage est principalement due au haut niveau des émotions « Joy » (joie) et « Happy-For » (bonheur pour quelqu'un d'autre) alors que la position des mains vient d'un geste déclenché par une variation importante du niveau des émotions « Admiration » et/ou « Relief » (soulagement).

Conclusion

Dans cet article, notre objectif consistait à explorer la possibilité d'utilisation d'un modèle intermédiaire basé sur les émotions entre les événements du système et le comportement d'un personnage virtuel. Pour ce faire, nous avons réalisé Émilie : un prototype d'agent émotif. Sa mise en œuvre nous a permis d'évaluer les différents modèles existant qui sont exploités pour la représentation de l'état émotionnel d'agents, ainsi que les différentes technologies existant pour la gestion de la représentation visuelle d'un personnage virtuel. Contrairement aux agents intégrés dans les systèmes auteurs habituels, le comportement et l'animation du personnage ne sont pas explicitement spécifiée, mais déterminé au moment de l'exécution selon un ensemble de règles paramétrées. La spécification de ces règles est indépendante de l'application ce qui permet la réutilisation du même agent. Un autre aspect visé par ce travail était de définir un mécanisme utilisant le modèle intermédiaire basé sur les émotions permettant que les réactions et le comportement de l'agent soient relatifs et conséquents au contexte plus large de l'évolution de l'apprenant plutôt qu'à l'événement initiateur de cette réaction ou comportement uniquement.

Un obstacle au travail effectué résidait dans la complexité, l'ampleur et la grande quantité de technologies intervenant dans un environnement où un tel agent peut s'intégrer. En effet, les réalisations d'une partie importante de la plateforme Cyberscience, du système de simulation des émotions et du module d'animation du personnage ont demandé des efforts significatifs. La réalisation de ce prototype a fait ressortir la complexité de l'assembler des composantes des systèmes multimédia, d'où l'intérêt d'avoir des composants réutilisables.

Malgré la simplicité relative de l'environnement d'expérimentation du prototype, la qualité et le raffinement des micro-mondes et surtout leur capacité à suivre l'apprenant est rapidement apparu comme un facteur critique au succès de système tutoriels intelligents adaptatifs. Notre contexte est particulièrement sensible car le dynamisme du comportement de l'agent est dépendant d'un flot régulier d'événements témoignant de l'avancement de l'apprenant. Indépendamment de cette limitation, les travaux réalisés ne sont qu'une étape dans la mise en œuvre des systèmes tutoriels intelligents considérant les aspects émotionnels comme part entière du développement de l'apprenant par la réalisation d'agents émotifs.

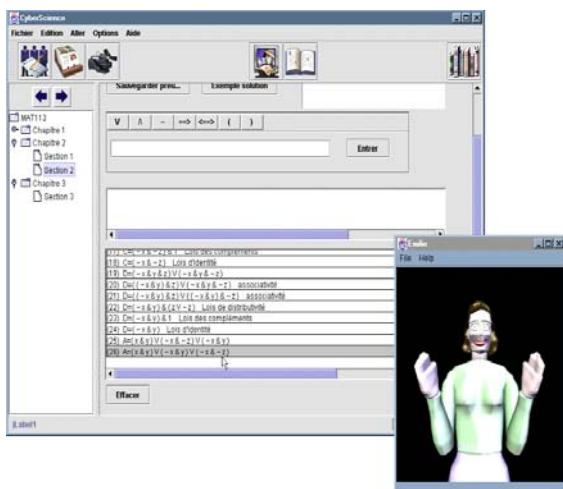


Figure. 2. L'utilisateur vient de compléter une activité difficile

Références

- [1] NKambou, R.; Frasson, C.; and Gauthier, G. 1998. A new approach to ITS - curriculum and course authoring: the authoring environment. *Computer & Education*. 31(1):105-130.
- [2] NKambou, R.; Gauthier, G.; and Frasson, C. 1996. CREAM-Tools: An authoring environment for curriculum and course building in an ITS, in *Computer aided learning and instruction in science and engineering*. Springer-Verlag: Berlin. p. 186-194.
- [3] Minsky, M. 1986. *The society of mind*. New York: Simon & Schuster.
- [4] Ortony, A.; Clore G.L.; and Collins, A. 1988. *The cognitive structure of emotions*. Cambridge (UK): Cambridge university press.
- [5] Toda, M. 1982. *Man, robot and society*. La Hague: Nijhoff.
- [6] Roseman, I.J. 1979. Cognitive aspects of emotion and emotional behavior. in 87th Annual convention, American Psychological association. New York.
- [7] Roseman, I.J. 1984. Cognitive determinants of emotions: a structural theory, in *Review of personality and social psychology*, S. P., Editor. Sage: Beverly Hills, CA.
- [8] Roseman, I.J.; Antoniou, A.A. and Jose, P.A. 1996. Appraisal determinants of emotions: Constructing a more accurate and comprehensive theory. *Cognition and emotion*. 10(3):241-277.
- [9] Picard, R.W. 1997. *Affective Computing.*, Cambridge, MA: MIT Press.
- [10] Velàsquez, J.D. 1997. Modeling emotions and other motivations in synthetic agents. in Fourteenth national conference on artificial intelligence and ninth innovative applications of artificial intelligence conference. Menlo Park.
- [11] Bates, J. 1994. *The role of emotion in believable agents*. Communications of the ACM.
- [12] Reilly, W. 1996. *Believable Social and Emotional agents*, in *School of computer science*. Carnegie Mellon University: Pittsburg.
- [13] Elliot, C. 1992. The affective reasoner : a process model of emotions in a multi-agent system, in *Technical report*, T.i.f.t.l. sciences, Editor., Northwestern university.
- [14] Elliot, C. 1997. I picked up catapia and other stories : a multimodal approach to expressivity for "emotionally intelligent" agents. in First international conference on autonomous agents.
- [15] Elliot, C.; J. Rickel and al. 1997. Integrating affective computing into animated tutoring agents. in *IJCAI Workshop on animated interface agents: making them intelligent*. Nagoya (Japon)s.
- [16] Lester, J.C.; and Stone, B.A. 1997. Increasing believability in animated pedagogical agents. in the first international conference on autonomous agents.
- [17] Gratch, J. 2000a. *Émile: Marshaling passions in training and education*. in *The fourth international conference on autonomous agents*. ACM Press.
- [18] Gratch, J. 2000b. Socially situated planning. in *AAAI Fall symposium on socially intelligent agents : the human in the loop*.
- [19] Rickel, J., et al. 2001. Steve goes to bosnia: Toward a new generation of virtual humans for interactive experiences. in *AAAI spring symposium on artificial intelligence and interactive entertainment*.
- [20] Rickel, J.; and Johnson, L. 1998. Animated agents for procedural training in virtual reality: perception, cognition and motor control. *Applied artificial intelligence (13)*
- [21] Corkill, D.D. 1991. BlackBoard systems. *AI expert*. 6(9): p. 40-47.
- [22] Gelernter, D. 1993. *Mirror worlds*. New York: Oxford University Press.
- [23] Velàsquez, J.D. 1998. A computational framework for emotion-based control. in 5th international conference of the society for adaptive behavior. Zürich.
- [24] ISO/IEC, 2001. Coding of audio-visual objects -- Part 1 : Systems, in ISO/IEC 14496-1.
- [25] Friedman-Hill, E. 2001. *JESS: The Java Expert System Shell*. Sandia national laboratories.