



HAL
open science

Aider à l'analyse du comportement d'un apprenant dans les jeux sérieux

Amel Yessad, Mathieu Muratet, Thibault Carron

► **To cite this version:**

Amel Yessad, Mathieu Muratet, Thibault Carron. Aider à l'analyse du comportement d'un apprenant dans les jeux sérieux. EIAH 2017, Jun 2017, Strasbourg, France. hal-01515748

HAL Id: hal-01515748

<https://hal.science/hal-01515748>

Submitted on 8 Feb 2022

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.

Aider à l'analyse du comportement d'un apprenant dans les jeux sérieux

Amel Yessad¹, Mathieu Muratet^{1,2}, Thibault Carron^{1,3}

¹ Sorbonne Universités, UPMC Paris 6, CNRS, LIP6 UMR 7606, Paris, France

² INS HEA, 58-60 Avenue des Landes, 92150 Suresnes, France

³ Université Savoie Mont Blanc, 73376 Le Bourget du Lac, France
{amel.yessad, mathieu.muratet, thibault.carron}@lip6.fr

Résumé. L'analyse des traces des apprenants dans les jeux sérieux est un défi important. En particulier, lorsque le jeu sérieux est caractérisé par un grand espace d'états et une grande liberté d'interaction de l'apprenant, l'analyse de ces interactions devient non seulement difficile d'un point de vue calculatoire mais également difficile à interpréter par les enseignants. Dans cet article, nous présentons un algorithme qui analyse les traces d'apprenants et produit un étiquetage sémantique et pédagogique de leurs résolutions du jeu sérieux. Cet algorithme exploite un modèle des solutions « correctes » fournies par les experts du domaine. Nous avons ensuite mené une étude expérimentale pour évaluer l'utilité et l'acceptation des étiquettes produites par l'algorithme proposé ainsi que le niveau de compréhension de ces mêmes étiquettes par les enseignants.

Mots-clés. Suivi de l'apprenant, jeu sérieux, retours aux enseignants, modèle comportemental, réseau de Petri, étiquettes pédagogiques

Abstract. Understanding play traces resulting from the learner's activity in serious games is a challenged research area. Especially, when the serious game is characterized by a large state space and a large amount of free interactions, the play traces become complex and thus hard to analyze and to interpret by teachers. In this paper, we present a framework that assists designers to build a model of an expert's solving process semi-automatically. Based on this model, we propose an algorithm that analyzes player's traces in order to generate pedagogical labels about the learner's behavior. We carried out an experimental study which aimed to evaluate the effectiveness of the labeling algorithm. From a usability point of view, we also use the experiment to validate the acceptance and readability of pedagogical labels by the teachers.

Keywords. Learner assessment, serious game, feedback, behavioral model, Petri net, pedagogical labels

1 Introduction

Le suivi des apprenants est une problématique importante du domaine des Environnements Informatiques pour l'Apprentissage Humain (EIAH). En particulier, les traces produites par les apprenants dans les environnements de formation interactifs tels que les jeux sérieux sont complexes et difficiles à analyser. Il est donc utile de doter

les enseignants d'outils capables d'analyser ces traces et de leur fournir un retour efficace et pertinent leur permettant de mieux interpréter le comportement des apprenants pour éventuellement leur apporter une aide adaptée. Notre approche vise à fournir un ensemble d'informations sémantiques et pédagogiques sur le comportement d'un apprenant évoluant dans un environnement complexe. Ces informations sont calculées en comparant la trace de l'apprenant avec un modèle des solutions fournies par les experts.

Nous axons notre recherche sur les jeux sérieux qui simulent des processus métiers complexes (physiques, industriels, économiques, etc.). Ces processus présentent souvent une explosion combinatoire de leurs espaces d'états avec une grande liberté d'interaction des utilisateurs, ce qui rend compliqué le suivi par les enseignants. Ainsi, notre objectif est de générer une description lisible du comportement des apprenants à destination des enseignants ou tout autre personne désirant analyser leur comportement (les tuteurs, les concepteurs de jeux, les joueurs eux-mêmes, etc.).

L'architecture globale est composée de deux parties : le processus permettant la construction semi-automatique du modèle intégrant les solutions référencées par les experts (que nous appelons dans la suite *solutions expertes*) et le système Laalys V2 (Learner Activity AnaLYSer) permettant le suivi des apprenants à partir de l'analyse de leurs traces. Cet article traite la deuxième partie et en particulier l'algorithme qui associe des étiquettes sémantiques aux actions d'un apprenant et calcule un score à partir de ces étiquettes. Ces dernières renseignent les utilisateurs et notamment les enseignants sur le comportement d'un apprenant en qualifiant par exemple les actions comme correctes, erronées, tardives ou prématurées. Le principe de cet algorithme d'étiquetage est de fournir des informations sémantiques sur les écarts détectés entre la résolution de l'apprenant et celles des experts. Nous avons mené une expérimentation avec des étudiants et enseignants de collège dans le but d'évaluer l'efficacité de l'algorithme d'étiquetage pour caractériser le comportement d'un apprenant et l'utilité des étiquettes produites pour les enseignants.

Dans la section suivante, nous dressons un panorama des recherches récentes en lien avec notre travail. Ensuite, nous présentons l'environnement qui permet d'assister les experts dans le processus de formalisation des solutions expertes. Dans la section 4, nous présentons de manière détaillée la contribution principale de cet article, à savoir, l'algorithme d'étiquetage des actions de l'apprenant. Dans la section 5, nous décrivons l'expérimentation menée et les résultats obtenus. Enfin, nous concluons l'article en mettant en avant quelques perspectives que nous considérons pertinentes pour la suite de ce travail.

2 Travaux similaires

Le travail de recherche présenté dans cet article est une contribution à la problématique du suivi des apprenants dans les jeux sérieux. La fouille de données et les techniques de l'intelligence artificielle ont déjà donné lieu à de nombreux travaux sur l'analyse des traces laissées par les apprenants en vue de fournir un feedback aux enseignants [1-4]. Or, peu de travaux ont été réalisés dans le cadre de systèmes intégrant de la simulation tels que les jeux sérieux qui nous intéressent.

Pour de tels systèmes, des recherches ont été menées dans le domaine de la visualisation de l'information pour fournir aux enseignants les moyens d'en explorer des représentations graphiques pour identifier quelles informations sont nécessaires à leurs activités pédagogiques. Par exemple, les travaux présentés dans [5, 6] se sont focalisés sur des indicateurs visuels de jeu et consistent à collecter de manière extensive les données des joueurs pendant le jeu pour guider la prise de décision. Ils visent à fournir à la fois une analyse du *gameplay* et des traces d'exécution des joueurs en s'appuyant sur le temps et le séquençement. Les techniques proposées ne se sont toutefois pas intéressées à définir des indicateurs liés au comportement des joueurs dans le domaine de l'éducation ce qui est justement le point clé de notre approche. Dans notre perspective, le système prend des décisions sur le comportement des apprenants en étiquetant leurs actions avec des métadonnées pédagogiques. L'objectif est de fournir un retour efficace et pertinent aux enseignants sur le comportement des apprenants.

Des recherches similaires aux nôtres ont été proposées par [7]. Les auteurs proposent une méthodologie pour extraire des caractéristiques conceptuelles des traces des apprenants et les utilisent pour classer les apprenants dans des groupes. Notre approche est différente de ces travaux car nous considérons toute la résolution de l'apprenant et pas seulement ses résultats. De plus, leur approche est basée sur de l'apprentissage machine alors que la nôtre utilise un modèle construit par les experts. Enfin ils traitent essentiellement des jeux sérieux de type puzzles tels que *RumbleBlocks*¹ ou *Refraction*² : leur approche n'est donc pas bien adaptée aux jeux sérieux à base de simulation caractérisés par de grands espaces d'états. Dans les travaux de [8], les auteurs associent au jeu une machine à états qui représente les situations que les enseignants souhaitent observer. Ils sélectionnent a priori les états qu'ils souhaitent tracer. Cette approche est très intéressante mais nécessite a posteriori un effort important de la part des enseignants d'interprétation des indicateurs remontés par le système.

D'autres recherches s'appuient sur les réseaux de Petri pour décrire la résolution provenant des experts et proposent aussi un algorithme pour étiqueter les actions des apprenants [9, 10]. Cependant, cet algorithme est seulement adapté aux jeux de type « étude de cas » et ne convient absolument pas pour des jeux sérieux impliquant de très grands espaces d'états et une grande liberté d'interaction des utilisateurs. Nos travaux s'inscrivent dans la continuité de ces recherches mais visent à proposer un environnement plus générique et capable de supporter le passage à l'échelle en vue de fournir des informations d'ordre pédagogique sur le comportement de l'apprenant. Ces étiquettes pédagogiques sont ainsi plus sémantiques et sont basées sur la comparaison entre le comportement des apprenants et celui de l'expert résolvant un niveau de jeu. Une autre contribution importante de notre approche est la construction semi-automatisée du modèle de résolution de l'expert à partir d'une description déclarative et de haut niveau des niveaux de jeu.

¹ RumbleBlocks : <http://rumbleblocks.etc.cmu.edu/> accédé le 4 janvier 2017

² Refraction : http://play.centerforgamescience.org/refraction/_remix/cgs/, accédé le 4 janvier 2017

3 Processus d'aide à la construction du modèle des solutions expertes

Le travail décrit dans cet article a pour objectif de présenter une approche d'analyse du comportement d'un apprenant au sein d'un jeu sérieux. Le système de suivi que nous proposons est basé sur la comparaison entre les traces de l'apprenant et la (les) solution(s) proposée(s) par des experts et modélisée(s) par un réseau de Petri. Un cadre méthodologique a été mis en place pour aider à la construction de ce réseau.

3.1 Brève présentation du formalisme des réseaux de Petri

Formellement, un réseau de Petri est un graphe biparti et valué, composé de deux types de nœuds : des places et des transitions. Chaque arc permet de connecter une place à une transition ou une transition à une place. Les places sont marquées et contiennent un nombre de jetons positif ou nul. Le vecteur qui associe à chaque place son marquage est appelé marquage du réseau de Petri et représente son état à l'instant t . Une transition du réseau de Petri peut être exécutée si elle est sensibilisée, c'est-à-dire, si chacune de ses places d'entrée contient au moins un nombre de jetons supérieur ou égal au poids de l'arc qui la relie à la transition en question. Lorsqu'une transition est exécutée, des jetons sont consommés des places d'entrée et d'autres jetons sont ajoutés aux places de sortie. L'ensemble des marquages atteignables à partir du marquage initial du réseau de Petri en exécutant au fur et à mesure les transitions sensibilisées représente l'ensemble des états du système et est appelé le graphe d'accessibilité. Pour plus de détails, nous invitons le lecteur à consulter la très abondante littérature sur les réseaux de Petri [11].

Le formalisme des réseaux de Petri est très puissant et correspond à l'association à la fois d'une théorie mathématique très bien définie et d'une représentation graphique du comportement du système modélisé. Les réseaux de Petri ont montré depuis longtemps leur capacité à modéliser des systèmes extrêmement complexes avec de grands espaces d'états. Pour toutes ces raisons, nous avons choisi de représenter les jeux sérieux qui nous intéressent avec des réseaux de Petri. En particulier, les places représentent les états des objets du jeu et les transitions représentent les actions que l'apprenant peut exécuter dans le jeu.

3.2 Construction semi-automatique des réseaux de Petri complets et filtrés

Dans le but de mettre en place un système de suivi automatique des apprenants dans les jeux sérieux, nous avons fait le choix d'assister semi automatiquement les concepteurs du jeu dans la construction des réseaux de Petri. En effet, il faut reconnaître que la construction manuelle des réseaux de Petri est problématique à cause de la complexité du formalisme et de la complexité des systèmes qui nous intéressent : cela se traduit par des réseaux de Petri de grandes tailles.

Pour atteindre notre but, nous construisons deux réseaux de Petri. Le premier réseau de Petri modélise toute la simulation du jeu incluant toutes les actions que l'apprenant peut exécuter dans le jeu et est appelé « Réseau de Petri complet » (RdpComple). Le deuxième réseau de Petri appelé « Réseau de Petri filtré » (RdpFiltré) est une partie du

RdpComplet incluant uniquement les actions que les experts ont utilisées pour résoudre le niveau du jeu. Des détails supplémentaires sur la construction semi-automatique de ces deux réseaux de Petri sont présentés dans [12].

3.3 Vue générale du cadre méthodologique

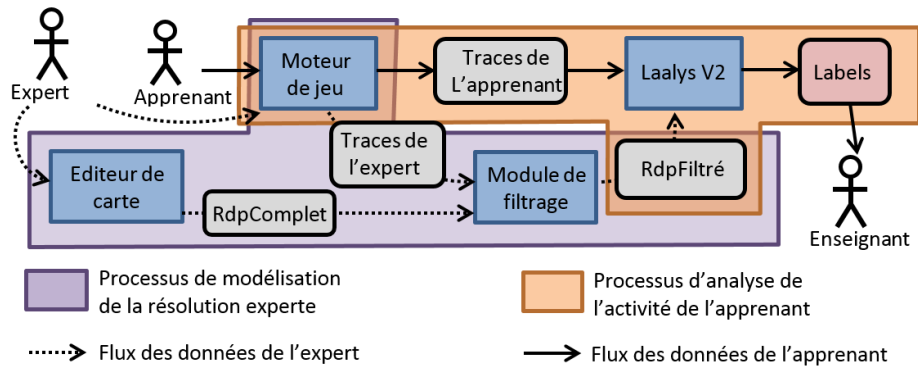


Fig. 1. Architecture générale des processus de traitement.

La Fig. 1 illustre les différents processus permettant de mettre en place le suivi des apprenants. Les concepteurs disposent d'un système auteur graphique qui leur permet de décrire les entités du jeu et leurs propriétés (par exemple une entité « porte » peut être ouverte ou fermée, une flaque d'eau peut se transformer en glace ou s'évaporer). Une fois, les entités du jeu décrites, le RdpComplet est généré automatiquement. En parallèle, le ou les experts peuvent jouer le niveau plusieurs fois (si plusieurs solutions « recommandées » existent pour un même niveau de jeu) et leurs actions sont alors tracées par le moteur du jeu. Ces traces sont utilisées pour filtrer le RdpComplet et générer ainsi le réseau de Petri filtré (RdpFiltré). Dans ce dernier, nous ne gardons que les transitions qui apparaissent dans les traces des experts, toute transition n'apparaissant pas dans les traces des experts est considérée comme non critique pour la résolution du niveau. Toutefois, ce processus de filtrage a deux conséquences : (1) Exclure des solutions intéressantes du RdpFiltré. Dans ce cas, des traces non-expertes pourraient également être utilisées pour filtrer le RdpComplet : par exemple, une solution originale et correcte faite par un apprenant et évaluée positivement par l'enseignant peut être ajoutée aux traces expertes pour élargir l'éventail de solutions présentes dans le RdpFiltré ; et (2) laisser des solutions qui ne sont pas « recommandées » par les experts (en effet toute combinaison des actions de jeu utilisées par l'expert ne constituent pas des solutions valides du problème posé). Dans ce cas le RdpComplet et le RdpFiltré doivent être vérifiés manuellement par les experts/concepteurs afin d'inclure des contraintes supplémentaires (c'est pour cette raison que nous parlons de construction semi-automatique). L'effort fourni lors de cette vérification est nettement inférieur à l'effort de la construction manuelle du réseau de Petri du jeu. Une fois le RdpFiltré validé, il est utilisé pour étiqueter les actions des apprenants d'un point de vue pédagogique.

4 Analyse du comportement de l'apprenant

Comme défini dans la section précédente, deux réseaux de Petri sont exploités : le premier (RdpComplet) modélise la simulation complète d'un niveau de jeu alors que le second (RdpFiltré) modélise uniquement les actions recommandées par les experts pour résoudre un niveau de jeu.

4.1 Algorithme d'étiquetage

Les entrées de l'algorithme d'étiquetage sont le RdpFiltré et la trace d'un apprenant. L'algorithme d'étiquetage s'appuie sur l'identification des 3 cas suivants : le premier cas correspond aux situations dans lesquelles l'apprenant a réalisé des actions qui sont refusées par le jeu et donc qui ne modifient pas l'état du jeu ; le deuxième cas correspond aux situations où l'apprenant a réalisé des actions qui sont acceptées par le jeu et donc en modifient l'état ; le troisième et dernier cas correspond à l'identification des actions manquantes alors que le jeu est terminé.

Chacun de ces cas mène à une description qui permet de caractériser l'action de l'apprenant au moyen d'une (et d'une seule) étiquette pédagogique.

Cas 1 - actions qui ne changent pas l'état du jeu

Ce cas apparaît lorsque l'apprenant essaie de réaliser une action qui n'est pas autorisée dans l'état actuel du jeu. Par exemple, si le joueur tente d'ouvrir une porte alors qu'il n'a pas la bonne clé. Nous avons identifié quatre étiquettes spécifiques pour caractériser ces actions de manière plus précise : (1) l'action était disponible dans le graphe d'accessibilité du RdpFiltré, l'étiquette générée pour étiqueter cette action est « *Tardive* », (2) l'action sera disponible dans le graphe d'accessibilité du RdpFiltré, l'étiquette générée est « *Prématurée* », (3) l'action a été disponible et le sera à nouveau plus tard dans le RdpFiltré, l'étiquette générée dans ce cas est « *Non synchronisée* », ou enfin (4) cette action n'est jamais disponible dans le graphe d'accessibilité du RdpFiltré, i.e. elle n'est jamais réalisée par les experts dans la résolution du niveau, l'étiquette générée dans ce quatrième cas est « *Erronée* ».

Cas 2 - actions qui changent l'état du jeu

Cela correspond au cas le plus courant, mais toutefois toutes les actions ne sont pas forcément correctes en comparaison de la résolution des experts. En effet, la plupart des jeux pour apprendre permettent à l'apprenant de faire des erreurs. Ce principe d'essai/erreur est au cœur des jeux vidéo. L'opportunité de réaliser des séquences d'actions inutiles, redondantes ou incorrectes dans un jeu sérieux [13] est fondamentale afin de capter l'attention des apprenants et tenter d'atteindre les objectifs pédagogiques visés. L'algorithme évalue donc si le nouvel état courant du jeu quitte l'*espace d'état filtré* (que nous appellerons f-space dans la suite de l'article et représente le graphe d'accessibilité du RdpFiltré), revient dans le f-space, se déplace à l'intérieur du f-space ou se déplace en dehors du f-space.

Cas 2.1 - sortie du f-space : ce cas peut apparaître quand l'apprenant joue une action qui n'a pas été réalisée par les experts. Si l'état du jeu est en-dehors du f-space, cela

signifie que l'état du jeu généré ne correspond à aucun des états présents dans le f-space. L'étiquette utilisée pour étiqueter cette action est « *Erronée* ».

Cas 2.2 - retour vers le f-space : cela correspond au moment où l'apprenant corrige de précédentes erreurs et retourne dans le f-space. Dans ce cas, l'algorithme (1) vérifie si l'état du jeu résultant de l'action de l'apprenant est inclus dans l'historique des états déjà atteints par l'apprenant, (2) calcule la longueur du plus court chemin entre ce nouvel état et un état final et (3) évalue la distance minimale (plus court chemin) entre chaque état de l'historique et un état final. En fonction des résultats sur ces trois critères, les étiquettes suivantes sont générées pour l'action de l'apprenant : « *Rattrapage* » ($1 \wedge (2 = 3)$), « *Retour arrière* » ($1 \wedge (2 > 3)$), « *Bond en avant* » ($\neg 1 \wedge (2 < 3)$) ou « *Non optimale* » ($\neg 1 \wedge (2 \geq 3)$).

Cas 2.3 - déplacement dans le f-space : dans ce cas, l'algorithme vérifie si l'apprenant progresse vers un état final ou tend à s'en éloigner. Ainsi l'algorithme (1) calcule la longueur du plus petit chemin entre le nouvel état et un état final, (2) évalue la longueur du plus court chemin entre l'état précédent et un état final, (3) vérifie si ces deux chemins sont différents, (4) vérifie si l'action effectuée est une action experte et (5) vérifie si le nouvel état est un successeur direct de l'état précédent dans le f-space. En fonction des résultats sur ces cinq critères, les étiquettes suivantes sont générées pour l'action de l'apprenant : « *Correcte* » ($((1 < 2) \wedge 4)$), « *Equivalente* » ($((1 < 2) \wedge \neg 4 \wedge 5)$), « *Bond en avant* » ($((1 < 2) \wedge \neg 4 \wedge \neg 5)$), « *Inutile* » ($((1 = 2) \wedge \neg 3)$), « *Non-optimale* » ($((1 = 2) \wedge 3)$), « *Régression* » ($1 > 2$).

Cas 2.4 - déplacement en dehors du f-space : ce comportement correspond au cas où l'apprenant réalise une action qui conserve le jeu dans un état en-dehors du f-space. Dans ce cas, l'algorithme essaie de calculer un nouveau f-space en initialisant le RdpFiltré à l'état courant du jeu. Si un état final est atteignable à partir de l'état courant, l'algorithme traite cette action comme dans le cas 2.3. Sinon, l'algorithme détermine si le nouvel état est plus proche (l'étiquette générée est « *Rapprochement* »), plus loin (l'étiquette générée est « *Eloignement* ») ou équidistant (l'étiquette générée est « *Stagnation* ») d'un état final du niveau que l'état qui le précède.

Cas 3 - actions manquantes lorsque le jeu est terminé

Si la dernière action de l'apprenant ne lui permet pas d'atteindre un état final du jeu, l'algorithme recherche le dernier état dans le f-space atteint par l'apprenant et calcule le chemin le plus court pour atteindre la fin du niveau à partir de cet état. Ensuite, l'algorithme étiquette les actions de ce chemin comme « *Manquantes* ».

4.2 Calcul du Score

L'algorithme d'étiquetage, présenté ci-dessus, produit 16 étiquettes pédagogiques. Nous associons un coefficient pour chaque étiquette afin d'être en mesure de proposer un score (voir Table 1). Ce score permettra, dans la phase expérimentation, de comparer l'évaluation produite par l'algorithme d'étiquetage avec les données des enseignants.

Tout d'abord, nous définissons comme des étiquettes positives, celles qui garantissent une progression vers un état final. Nous définissons les coefficients associés à ces étiquettes dans l'intervalle]0 ; 1]. Pour cette classe, nous retenons les étiquettes suivantes : « *Correcte* » dénote un comportement satisfaisant par rapport à la solution

des experts ($coeff = 1$); « Equivalente » est similaire à une action « Correcte », mais fait référence à une action non-experte (d'où un coefficient plus bas : 0,8); « Bond en avant » signifie que l'apprenant suit un processus de résolution non-expert mais se rapproche de la solution ($coeff = 0,6$); et « Rapprochement » est associée au plus petit coefficient positif car cela arrive lorsque l'apprenant se déplace en dehors du f-space ($coeff = 0,4$).

Table 1. Coefficients des étiquettes positives et négatives.

Etiquettes positives	Coeff.	Etiquettes négatives	Coeff.
Correcte	1	Inutile	-0,1
Equivalente	0,8	Eloignement	-0,2
Bond en avant	0,6	Manquante	-0,3
Rapprochement	0,4	Erronée	-0,5

Ensuite, nous définissons comme étiquettes négatives, celles qui représentent un écart sûr par rapport à la solution des experts. Afin de donner plus de poids aux étiquettes positives dans le calcul du score, nous définissons les coefficients négatifs dans l'intervalle $[-0,5; 0[$. Pour cette classe, nous retenons les étiquettes suivantes : « Erronée » est l'opposée de « Correcte » ($coeff = -0,5$); « Manquante » apparaît quand les apprenants abandonnent le niveau, nous choisissons de pondérer fortement cette étiquette pour dégrader le score des étudiants dans le cas d'un abandon ($coeff = -0,3$); « Eloignement » est l'opposée de « Rapprochement » ($coeff = -0,2$); et « Inutile » ne caractérise pas un erreur majeure et a donc une influence mineure sur le score ($coeff = -0,1$).

Finalement, les étiquettes restantes (Non-optimale, Rétablissement, Retour arrière, Stagnation, Prématurées, Tardives, Régression et Non synchronisée) sont classées comme étiquettes neutres. Même si leurs coefficients sont mis à 0, elles contribuent au calcul du score en augmentant la taille de la trace et dégradent ainsi le score de l'apprenant.

Dans l'expérimentation que nous avons menée, le score d'une trace d'apprenant a été calculé par la formule (1) et est défini dans l'intervalle $[-0,5; 1]$. Pour une trace, nous notons $L = \{l_1, l_2, \dots, l_{16}\}$ l'ensemble incluant le nombre de chaque sorte d'étiquette et $C = \{c_1, c_2, \dots, c_{16}\}$ l'ensemble définissant le coefficient de chaque sorte d'étiquette.

$$score = \frac{\sum_{i=1}^{16} l_i * c_i}{\sum_{i=1}^{16} l_i} \quad (1)$$

Naturellement, la pondération des étiquettes et la méthode de calcul de score peuvent être adaptées en fonction des besoins, objectifs et des pratiques pédagogiques des enseignants.

5. Etude expérimentale

L'architecture globale a été évaluée sur un prototype de jeu sérieux appelé « Les Cristaux d'Ehere », conçu pour enseigner les concepts de la physique à des élèves de

5^{ème} au collège. L'objectif pour chaque niveau est de résoudre des problèmes mettant en jeu des compétences relatives à la compréhension de processus de changements d'états de l'eau. Les apprenants doivent déplacer un avatar pour interagir avec des objets du jeu et atteindre une solution mettant en œuvre des compétences en physique. Tous les niveaux du jeu (18) ont été conçus en suivant le cadre méthodologique que nous avons présenté dans cet article.



Fig. 2. Copie d'écran du niveau « Le mur de glace »

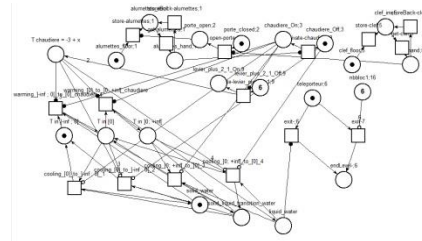


Fig. 3. Réseau de Petri filtré (construit automatiquement) du niveau « Le mur de glace »

Nous présentons ici les résultats concernant deux niveaux : « Le thermomètre » et « Le mur de glace ». Dans le premier, « Le thermomètre », l'apprenant travaille sur la compétence « Lire la température précisément sur un thermomètre analogique » et dans le second, « Le mur de glace », le travail repose sur des compétences relatives à la compréhension de la fonte de l'eau à l'état solide. Les Fig. 2 et 3 montrent une copie d'écran et le réseau de Petri filtré (construit automatiquement) du niveau « Le mur de glace ».

5.1 Protocole expérimental

L'objectif de cette expérimentation est de déterminer si les étiquettes et le score calculé par l'algorithme donnent une idée satisfaisante du comportement d'un apprenant et sont cohérents avec les retours (scores) proposés par les enseignants.

Nous avons mené une expérimentation de type qualitative avec neuf élèves et quatre enseignants de physique de collège. Un protocole expérimental a été suivi afin d'analyser l'efficacité de l'algorithme d'étiquetage auprès d'enseignants. Tout d'abord, nous avons filmé les élèves pendant les sessions de jeu. Ensuite, nous avons présenté les niveaux de jeu aux enseignants.

D'une part, nous avons demandé à trois enseignants sur les quatre de visualiser les vidéos des apprenants résolvant les niveaux « Le thermomètre » et « Le mur de glace ». Nous leur avons demandé d'évaluer le comportement des apprenants avec un score global sur une échelle de 1 à 3 : 1 pour un élève qui a échoué ou rencontré de sérieuses difficultés à résoudre le niveau, 2 pour un élève qui rencontre quelques difficultés et 3 pour un élève qui a montré une bonne compréhension des concepts du domaine sous-jacents à la résolution du niveau. Ensuite, nous avons fourni la liste des étiquettes pédagogiques et leur signification. Nous avons demandé à ces mêmes enseignants d'étiqueter les actions des élèves manuellement en utilisant la liste des étiquettes. Une

fois qu'ils avaient étiqueté les actions des élèves, nous leur avons présenté les étiquettes générées par l'algorithme d'étiquetage. Enfin, nous avons soumis aux enseignants un questionnaire afin d'évaluer la qualité des étiquettes générées automatiquement.

D'autre part, nous avons demandé au quatrième enseignant de visualiser les vidéos de tous les apprenants pendant les deux niveaux et nous lui avons demandé d'évaluer le comportement des apprenants avec la même échelle de score que celle des trois autres enseignants. Finalement, nous avons comparé le score calculé automatiquement par LaaLys V2 avec les notes données par cet enseignant.

5.2 Résultats et discussion

L'avis des enseignants sur les résultats de l'algorithme d'étiquetage

Les trois enseignants ont été capables d'étiqueter les traces des élèves en 3 à 10 minutes, en fonction de la longueur de la trace. Ils considèrent que les étiquettes ont un sens pédagogique et apportent une bonne compréhension du comportement des élèves.

Quand nous avons comparé l'étiquetage automatique avec ceux réalisés manuellement par les enseignants, nous avons noté une grande similarité. Néanmoins, les enseignants ont souvent confondu les étiquettes « Rattrapage » et « Retour arrière ». Nous pensons que ces deux étiquettes sont en effet très similaires mais restent sémantiquement différentes et nécessiteraient plus d'explication. Un professeur n'a pas compris la signification de l'étiquette « Non synchronisée » et par conséquent, ne l'a pas utilisée. Après discussion avec cet enseignant, nous pensons que cette étiquette peut être assimilée à l'étiquette « Prématurée » car le fait qu'une action fut disponible dans le passé n'est pas important et que la caractéristique la plus utile pour l'enseignant est surtout de savoir si une action va être disponible dans le futur.

Finalement, les trois enseignants ont estimé que les étiquettes choisies par l'algorithme étaient plus précises que leurs propres choix et que cela représente une économie en temps substantielle comparé à la production et la visualisation des vidéos. Ils proposent dans le même esprit d'extraire automatiquement des motifs plus abstraits de regroupement d'étiquettes : par exemple, des séquences systématiques dans une trace de plusieurs actions « Prématurée » ou « Tardive » pourraient montrer qu'un apprenant ne comprend pas les contraintes temporelles des processus simulés par le jeu. Cela permettrait de monter la sémantique au bon niveau de compréhension : celui attendu par les enseignants.

Table 2. Comparaison des scores de chaque élève pour les niveaux « Le thermomètre » et « Le mur de glace ».

	Le thermomètre									Le mur de glace								
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S1	S2	S3	S4	S5	S6	S7	S8	S9
Score de Laalys V2	2	1	1	3	1	1	1	1	1	3	1	2	3	1	3	1	3	1
Score des enseignants	2	1	1	3	1	3	1	2	1	3	2	2	3	1	2	1	3	1
Longueur de la trace	8	27	18	6	23	18	33	35	32	9	22	13	13	20	12	13	9	42

Comparer les scores de Laalys V2 avec les notes données par les enseignants

Le tableau 2 montre les données expérimentales sur les niveaux « Le thermomètre » et « Le mur de glace ». Pour ces deux niveaux, nous présentons les données pour les neuf élèves (S1 à S9). La première ligne donne pour chaque élève le score automatique fourni par l'algorithme (rapporté et discrétisé sur l'ensemble des valeurs [1 ; 2 ; 3]) et la deuxième ligne le score des enseignants. La dernière ligne donne la longueur de la trace. Le coefficient k de Kappa ($k=0,64$) permet de conclure à un accord fort entre les scores des enseignants et ceux de Laalys V2. Toutefois, nous notons une divergence entre les scores des deux élèves S6 et S8 pour « Le thermomètre » et entre les scores des deux élèves S2 et S6 pour « Le mur de glace » :

- Thermomètre S6 : Ce cas est le plus grand écart entre la note donnée par l'enseignant (3) et le score calculé automatiquement (1). L'enseignant a noté que l'élève a tâtonné pour atteindre la solution, a essayé un grand nombre de choses et a vérifié la température après chaque action. L'algorithme a détecté ce comportement en étiquetant plusieurs actions avec l'étiquette « Stagnation » dans la trace de l'élève. Cependant, l'enseignant a identifié un bug dans le jeu, l'outil d'aide (apparition d'une loupe facilitant la lecture de la température) n'a pas été activé pendant l'activité de l'élève S6. L'enseignant a pris en compte ce problème et cela explique son indulgence ;
- Thermomètre S8 : L'enseignant note que cet élève actionne les leviers contrôlant la température sans tester la sortie à chaque fois. Selon l'enseignant, c'est un bon comportement mais le moteur du jeu ne trace pas l'information de tester la sortie. Si c'était le cas, le score calculé aurait été augmenté par rapport aux autres élèves qui n'avaient pas ce comportement ;
- Le mur de glace S2 : L'enseignant justifie sa note avec les mêmes explications que pour l'élève S5 mais n'a pourtant pas donné la même note. Du point de vue de l'analyse automatique, les deux traces sont très similaires et donc le score l'est aussi. Cela semble montrer un résultat plus cohérent que l'évaluation de l'enseignant ;
- Le mur de glace S6 : L'enseignant a commenté le comportement de l'élève comme celui des élèves qui ont obtenu la note 3 (S1, S4 et S8). Or la note est différente (2). Il n'a pas expliqué les différences entre ces élèves. Là encore l'analyse automatique semble plus cohérente.

6 Conclusion et travaux futurs

Le travail présenté dans cet article concerne l'évaluation du comportement des apprenants dans des jeux sérieux avec de grands espaces d'états. La contribution principale est un algorithme qui permet d'étiqueter les actions de l'apprenant au moyen d'un ensemble d'étiquettes pédagogiques. Un cadre méthodologique a été suivi pour concevoir les 18 niveaux du jeu sérieux « Les Cristaux d'Ehère » et a permis de produire les réseaux de Petri (complets et filtrés) de ces niveaux automatiquement. L'algorithme d'étiquetage a été expérimenté lors d'une étude qualitative avec neuf élèves et quatre enseignants de collège. L'évaluation a donné des résultats positifs à la fois sur l'utilité des étiquettes et sur la concordance des scores calculés par le système avec les notes

des enseignants. Ce travail de recherche a également permis d'envisager plusieurs nouvelles pistes : (1) utiliser l'analyse du comportement de l'apprenant et son score pour par exemple l'affecter à un groupe de niveau, (2) utiliser les étiquettes afin d'implémenter un retour (feedback) en ligne et adapté à chaque élève, (3) implémenter la propagation des étiquettes dans un modèle d'apprenant et (4) comparer les réseaux de Petri (RdpFiltré) construits par filtrage automatique des réseaux complets (RdpComplet) avec les réseaux de Petri élaborés manuellement par les experts.

Références

1. Essam Kosba, Vania Dimitrova, and Roger Boyle. Adaptive feedback generation to support teachers in web-based distance education. *User Modeling and User-Adapted Interaction*, 17(4):379–413, 2007.
2. Brijesh Kumar Baradwaj and Saurabh Pal. Mining educational data to analyze students' performance. CoRR, abs/1201.3417, 2012.
3. Charoula Angeli and Nicos Valanides. Using educational data mining methods to assess field-dependent and field-independent learners' complex problem solving. *Educational Technology Research and Development*, 61(3):521–548, 2013.
4. Roger Azevedo, Jason Harley, Gregory Trevors, Melissa Duffy, Reza Feyzi-Behnagh, François Bouchet, and Ronald Landis. Using trace data to examine the complex roles of cognitive, metacognitive, and emotional self-regulatory processes during learning with multi-agent systems. In *International handbook of metacognition and learning technologies*, pages 427–449. Springer, 2013.
5. Ben Medler, Michael John, and Jeff Lane. Data cracker: Developing a visual game analytic tool for analyzing online gameplay. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 2365–2374, New York, NY, USA, 2011. ACM.
6. G. Wallner and S. Kriglstein. Technical section: Plato: A visual analytics system for gameplay data. *Comput. Graph.*, 38:341–356, February 2014.
7. Erik Harpstead, Christopher J. MacLellan, Kenneth R. Koedinger, Vincent Aleven, Steven P. Dow, and Brad A. Myers. Investigating the solution space of an open-ended educational game using conceptual feature extraction. In *Proceedings of the 6th International Conference on Educational Data Mining, Memphis, Tennessee, USA, July 6-9, 2013*, pages 51–58, 2013.
8. Ángel del Blanco, Javier Torrente, Eugenio J. Marchiori, Iván Martínez-Ortiz, Pablo Moreno-Ger, and Baltasar Fernández-Manjón. Easing assessment of game-based learning with e-adventure and lams. In *Proceedings of the Second ACM International Workshop on Multimedia Technologies for Distance Learning*, MTDL '10, pages 25–30, New York, NY, USA, 2010. ACM.
9. Amel Yessad, Pradeepa Thomas, Bruno Capdevila Ibáñez, and Jean-Marc Labat. Using the petri nets for the learner assessment in serious games. In ICWL, pages 339–348, 2010.
10. Pradeepa Thomas, Jean-Marc Labat, Mathieu Muratet, and Amel Yessad. How to evaluate competencies in game-based learning systems automatically? In *Intelligent Tutoring Systems*, pages 168–173. Springer Berlin Heidelberg, 2012.
11. J. L. Peterson. *Petri Net Theory and Modeling of Systems*. Prentice Hall, Reading, Mass, 1981.
12. En raison de la contrainte d'anonymat, cette référence sera intégrée dans la version finale.
13. Pradeepa Thomas, Amel Yessad, and Jean-Marc Labat. Petri nets and ontologies: Tools for the "learning player" assessment in serious games. In *ICALT*, pages 415–419, 2011.