



HAL
open science

Personnalisation interactive de parcours pédagogiques par une méthode de décision multicritère à base de traces

Hoang Nam Ho, Mourad Rabah, Samuel Nowakowski, Pascal Estrailier

► **To cite this version:**

Hoang Nam Ho, Mourad Rabah, Samuel Nowakowski, Pascal Estrailier. Personnalisation interactive de parcours pédagogiques par une méthode de décision multicritère à base de traces. 7ème Conférence sur les Environnements Informatiques pour l'Apprentissage Humain (EIAH 2015), Jun 2015, Agadir, Maroc. pp.429-431. <hal-01405981>

HAL Id: hal-01405981

<https://hal.science/hal-01405981v1>

Submitted on 30 Nov 2016

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.



HAL Authorization

Personnalisation interactive de parcours pédagogiques par une méthode de décision multicritère à base de traces

Hoang Nam Ho¹, Mourad Rabah¹, Samuel Nowakowski², Pascal Estrailier¹

¹ Laboratoire L3i, Université de La Rochelle, La Rochelle, France
{hoang_nam.ho, mourad.rabah, pascal.estrailier}@univ-lr.fr

² LORIA, UMR 7503, Université de Lorraine, Nancy, France
samuel.nowakowski@loria.fr

Résumé. Les travaux que nous menons pour l'e-Éducation portent sur les architectures logicielles des applications interactives à exécution adaptative. Nous nous appuyons sur une hypothèse de structuration de l'application au moyen de *situations* (ensemble d'interactions contextualisées). Nous introduisons une approche qui étend la méthode de décision multicritère traditionnelle PROMETHEE II par l'évaluation des situations selon différents critères à partir des traces des exécutions précédentes en appliquant la méthode kNN (k plus proches voisins). Nous évaluons la pertinence de notre approche par une étude de cas basée sur le système du Tamagotchi en déduisant l'applicabilité au e-Learning.

Mots-clés. décision multicritère, PROMETHEE II, traces, e-Learning

Abstract. Our work deals with software architectures for interactive applications with adaptive execution. The article is based on the assumption that the application is structured with contextual interaction sequences called *situations*. When one situation ends, the system chooses the best following one. We introduce an approach that overcomes a drawback of the traditional multicriteria decision method PROMETHEE II by selecting the next situation according to different criteria applied on system's traces using the kNN (k nearest neighbours) method. We demonstrate our approach on a case study based on the Tamagotchi system in deducing the applicability in e-Learning.

Keywords. multicriteria decision, PROMETHEE II, traces, e-Learning

1 Introduction

Les travaux décrits dans cette communication détaillent une approche basée sur la modélisation d'un dispositif de e-Éducation intégrant dans l'environnement d'apprentissage des applications interactives à exécution adaptative. Pour cela, nous nous appuyons sur la notion de *situation* [1] pour structurer le scénario pédagogique. Une situation est un composant où les acteurs interagissent en utilisant des ressources locales associées à un contexte commun en vue d'atteindre un ou plusieurs objectifs identifiés. Ainsi, l'utilisateur d'une application interactive exécute et participe à des situations successives jusqu'à atteindre un objectif défini par le concepteur. Notre objectif est alors d'intégrer dans un environnement numérique d'apprentissage un dispositif permettant d'optimiser le choix des situations en vue d'atteindre, par un

chemin personnalisé, un objectif pédagogique donné. Dans nos travaux précédentes, nous avons proposé un processus de décision multicritère [2] et nous nous plaçons ici à la dernière étape de ce processus, celle de la prise de décision. Dans cet article, le choix s'appuiera sur une heuristique de décision multicritère [3]. Nous nous sommes plus particulièrement intéressés à la méthode PROMETHEE II [4] que nous avons enrichie en intégrant un système d'analyse des traces générées lors de l'exécution [5], [6] pour améliorer le processus de décision. Nous avons validé notre approche sur une application de type jeu : le Tamagotchi. Nous avons conçu un prototype de ce jeu comme un dispositif d'apprentissage afin de pouvoir y valider les approches que nous intégrerons dans l'EIAH en cours de développement dans notre laboratoire.

2 Mise en œuvre de PROMETHEE II à base de traces

Nous définissons les concepts de traces et de modèle de traces [5], [6]. Une *Trace* est une information ou une séquence d'informations générées par toute action relative à un objet ou un événement. Chaque trace peut être associée à un modèle, appelé *modèle de trace* et représentant la trace d'une manière formelle. Une *trace modélisée* ou *m-trace* est une trace effective associée au modèle correspondant. Un système qui gère les m-traces est appelé *système à base de traces modélisées*. Dans nos travaux, nous n'utilisons que les trois composants les plus courants d'un SBT :

Système de collecte : détermine les informations à tracer en se basant sur des observateurs. Il récupère les données pendant l'exécution de l'application. Ces données sont des *traces premières*, noté $P = (U, S, E, \Omega, C)$ où : $U = (user_id, date)$ décrit l'utilisateur, $S = \{s_i\}$ est l'ensemble des situations exécutées, $E \subseteq S \times S$ est l'ensemble de transitions entre situations, $\Omega \subseteq V \times S$ est l'ensemble des propriétés du vecteur d'état et la situation exécutée correspondante, C est l'ensemble des critères.

Système de transformation : permet d'extraire des traces premières les données dont nous avons besoin, dites *traces transformées*. Dans notre contexte, nous extrayons uniquement Ω pour obtenir un ensemble d'enregistrements notés T .

Système d'analyse : permet d'analyser les enregistrements issus du système précédent. Nous l'intégrons comme support de mise en œuvre dans PROMETHEE II. À chaque instant de l'exécution, nous disposons d'un vecteur d'état global $V = \{pro_k^i\}$ contenant p propriétés qui indiquent la valeur de l'attribut i contribuant au critère c de l'application. PROMETHEE II analyse le vecteur V selon m critères sur l'ensemble A de n alternatives pour choisir la meilleure solution combinant ces m critères. Nous avons une base de traces transformées T obtenue par le SBT. Chaque enregistrement dans T a le format : $(V, situation_choisie)$. De T , nous extrayons q enregistrements comportant une des n situations alternatives : on choisit q tel que $situation_choisie_q \in A$. Puis, avec la base extraite, nous créons m ensembles fonction des m critères. Comme un critère est évalué par une ou plusieurs propriétés dans le vecteur d'états, les enregistrements de chaque ensemble auront un format différent. Plus généralement, chaque enregistrement dans l'ensemble du critère c est évalué par w propriétés et est représenté par : $eng = (pro_c^1, pro_c^2, \dots, pro_c^w, situation_choisie)$.

Enfin, nous décomposons le vecteur d'états V en m sous-vecteurs selon les m critères. Le sous-vecteur du critère c est décrit par $v^c = (pro_c^1, pro_c^2, \dots, pro_c^w)$. Pour chaque ensemble obtenu dans la phase précédente, nous calculons la distance euclidienne $dis_c^q(alternative_i)$ qui représente la distance de l'alternative i entre v^c et eng_q . Nous obtenons donc pour chaque critère une matrice de taille $q \times 1$ contenant toutes les distances entre le sous-vecteur d'état actuel et les q enregistrements. À partir des m matrices précédentes, nous choisissons k plus proches voisins [7] pour évaluer l'alternative i sur le critère c , notée $E_c(alternative_i)$:

$$E_c(alternative_i) = \frac{e_c(alternative_i)}{\sum_{i=1}^n e_c(alternative_i)}, \text{ avec } e_c(alternative_i) = \sum_{j=1}^k dis_c^j(alternative_i)$$

Une fois les calculs effectués pour chaque critère, PROMETHEE II enchaîne avec les étapes restantes pour construire une liste prioritaire des alternatives, dont la première situation de cette liste sera la situation à exécuter.

3 Résultat et Conclusion

Nous avons testé notre approche sur une mise en œuvre du jeu Tamagotchi sur l'entretien d'un animal virtuel. Nos premiers résultats nous permettent de montrer la performance, en termes d'évolution des critères et de temps d'exécution total de notre approche, est meilleure que celle de PROMETHEE II classique. Les principes abordés sont aisément transposables au domaine de l'apprentissage en ligne où la notion de parcours pédagogique est une succession de situations d'apprentissage contextualisées (cours, travail individuel, tests, etc.). Nos travaux actuels s'orientent donc sur les applications de cette approche dans le contexte des EIAH.

Références

1. Pham, P.T., Rabah, M., Estrailier, P.: Agent-based architecture and situation-based scenario for consistency management. 2013 Federated Conference on Computer Science and Information Systems. pp. 1041–1046 (2013).
2. Ho, H.N., Rabah, M., Nowakowski, S., Estrailier, P.: Trace-Based Weighting Approach for Multiple Criteria Decision Making. Journal of Software. Vol. 9, 2180–2187 (2014).
3. Köksalan, M., Wallenius, J., Zionts, S.: Multiple Criteria Decision Making: From Early History to the 21st Century. World Scientific (2011).
4. Taillandier, P., Stinckwich, S.: Using the PROMETHEE multi-criteria decision making method to define new exploration strategies for rescue robots. IEEE International Workshop on Safety, Security, and Rescue Robotics. pp. 193–202 (2011).
5. Laflaquière, J., Settouti, L.S., Prié, Y., Mille, A.: Trace-Based Framework for Experience Management and Engineering. 10th International Conference, KES 2006. pp. 1171–1178. , Bournemouth, UK (2006).
6. Cordier, A., Lefevre, M., Champin, P.-A., Georgeon, O.L., Mille, A.: Trace-Based Reasoning - Modeling Interaction Traces for Reasoning on Experiences. In: Boonthum-Denecke, C. and Youngblood, G.M. (eds.) FLAIRS Conference. AAAI Press (2013).
7. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2011).