

Modélisation Sémantique de l'Utilisateur

Charles Tijus, Sébastien Poitrenaud
Jean-François Richard

Laboratoire Cognition & Usages
Université Paris 8, 2 rue de la Liberté, F-93526 St Denis cedex 02
tijus@univ-paris8.fr

Résumé. Notre approche « sémantique de l'utilisabilité », basée sur la catégorisation, correspond à un mode de représentation des connaissances, sous la forme d'un treillis de Galois qui permet de modéliser et simuler les procédures utilisateurs sur un dispositif technique. Cette approche, qui diffère de celles qu'on trouve avec SOAR ou ACT, associe les actions et les procédures aux catégories d'objets, comme propriétés de ces catégories (Poitrenaud, Richard & Tijus, sous presse). L'accès aux actions et procédures a lieu à partir des catégories d'objets. Dans le cadre de cette approche, les erreurs relèvent de méprises catégorielles et l'analogie relève des processus de reconnaissance qui ont lieu lors de la catégorisation. La modélisation et la simulation dans le cadre de cette approche se réalisent avec les formalismes développés par Poitrenaud (1995): ProcOpe et STONE.

1 La sémantique de l'utilisabilité

On peut avoir deux points de vue sur l'utilisateur. Un premier point de vue est celui de la conception de dispositifs qui désire connaître et modéliser ses utilisateurs cible, pour savoir comment personnaliser l'interface, quels liens recommander selon le profil utilisateur, etc. Un deuxième point de vue est celui de l'utilisateur lui-même qui peut se demander pourquoi l'interface ne lui convient pas et pourquoi il n'a pas perçu la signification du lien qui correspondait à sa recherche. Ces deux points de vue sont complémentaires et leur intégration devrait donner lieu à une modélisation plus complète de l'utilisateur. L'approche de la sémantique de l'utilisabilité correspond au second point de vue. Elle découle des recherches menées dans notre laboratoire sur la résolution de problème (Richard, Poitrenaud, & Tijus, 1993 ; Richard, Clément & Tijus, 2002). Ces recherches montrent que ce n'est pas l'organisation des actions qui freine une planification réussie, mais la conceptualisation des objets de l'action. D'où la grande différence de difficulté trouvée entre des problèmes isomorphes, de la Tour de Hanoi par exemple, qui ont des habillages différents. Ainsi, le problème de la Tour de Hanoi qui consiste à changer la place de 3 disques de taille différente se résout en moyenne au 11.4 coups alors que l'isomorphe qui consiste à changer la taille de 3 disques de place différente se résout en moyenne au bout de 35.5 coups, alors que 7 coups suffisent dans les deux cas. On mesure toute l'importance de la sémantique, c'est-à-dire de la signification accordée aux objets qu'on manipule. Ainsi, le paradigme de la Tour de Hanoi, problème apparemment trivial pour l'informaticien, doit être considéré sérieusement, lorsqu'on découvre les grandes différences de comportement lorsque change l'habillage du problème, c'est-à-dire sa sémantique. C'est aussi un paradigme qui sert de base à la modélisation de l'utilisateur et sert à valider les propositions de modèles.

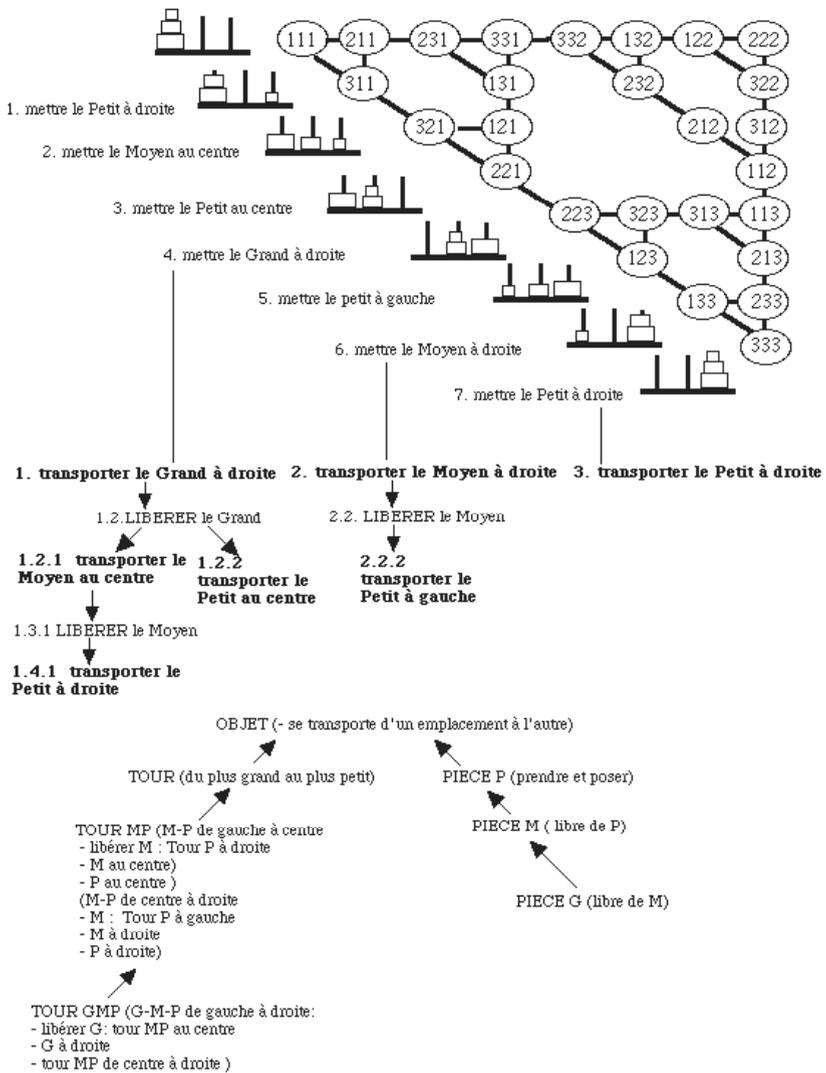


FIG. 1 – Quatre descriptions de la procédure de résolution de la Tour de Hanoi pour cheminer dans le graphe des états de l'état 111, où les 3 disques, petit, moyen et grand, sont à l'emplacement de droite codé 1 (111) jusqu'à l'état où ils sont à l'emplacement de gauche codé 3 (333) par le chemin le plus court (7 coups) ; ceci en respectant les règles (R1) de ne déplacer qu'un disque à la fois (R2) qui ne peut pas être pris s'il y a un plus petit au même endroit et (R3) qui ne peut être posé à un emplacement où il y a un plus petit.

La première description de la figure 1 correspond au cheminement dans le graphe d'états, avec l'automate correspondant, et est utilisée avec les réseaux de Pétri (par exemple AMME, pour Automatic Mental Model Evaluator, Fjeld, Schlupe & Rauterberg, 1999). La seconde description est une suite d'actions primitives. Ce mode de description est utilisé dans le modèle Network de Mannes (Mannes & Kintsch, 1991) où un réseau connexionniste associe par apprentissage les réponses que sont les actions, d'une part aux états rencontrés et d'autre part aux actions qui précèdent. Ce modèle qui a servi à simuler les tâches d'édition avec un traitement de texte simule le comportement expert, et est basé sur le passé. Il est utilisé également dans le modèle ACT d'Anderson (1993), sous forme de règles de productions qui sont sélectionnées lorsque les conditions des actions correspondantes sont satisfaites, les conditions étant les règles du problème, et par une mesure de l'écart au but : entre plusieurs actions candidates est sélectionnée celle qui permet de se rapprocher le plus du but. On a plutôt ici une primauté de l'avenir, le but, sur le comportement. La troisième description a la forme d'une structure de buts (Nguyen & Hoc, 1987). Son intérêt est de fournir une planification hiérarchique et une sémantique à l'action : la justification d'une action est sa participation à la réalisation de l'action superordonnée. La quatrième description est une représentation orientée-objet du savoir de l'utilisateur (Barcenilla, Leproux, Poitrenaud, 2002 ; Poitrenaud, 1995 ; Richard, Clément & Tijus, 2002) qui intègre la structure de buts à une représentation des catégories d'objets: des pièces qu'on manipule pour déplacer des tours. Ici, la sémantique de l'action comprend la justification des actions par leur participation à la réalisation des actions qui leur sont superordonnées, mais aussi un savoir sur les objets de l'action, qui correspond aux propriétés structurales de ces objets qui justifient les actions.

2 Evaluation des modèles utilisateurs

Les modèles utilisateurs doivent pouvoir être évalués. Parmi les catégories de critères d'évaluation, il y a ceux qui relèvent des catégories suivantes et qui concernent les prédictions qui peuvent être faites quant aux interactions humain-machines, aux remédiations à apporter aux interfaces, et aux comparaisons ergonomiques entre interfaces alternatives : (1) la représentation et la simulation du fonctionnement, (2) l'apprenabilité, et (3) l'utilisabilité.

2.1 Représentation et simulation du fonctionnement du dispositif

Le modèle utilisateur doit pouvoir simuler le fonctionnement du dispositif pour disposer de ses réponses suite aux actions de l'utilisateur, ne serait-ce que pour pister l'évolution de l'interaction humain-machine, mais surtout pour la simuler. Le fonctionnement qui doit être simulé comprend le fonctionnement sous l'action de l'utilisateur, mais aussi le fonctionnement indépendant de l'utilisateur, lorsque la machine « prend la main » (pour des alertes, par exemple), ou suite à l'intervention d'autres opérateurs (avec les dispositifs de communication, par exemple). Enfin, ce fonctionnement simulé doit être le fonctionnement lors de l'application de procédures licites, mais aussi lors de l'application de procédures illicites.

Cette composante du modèle utilisateur est lourde et difficile à mettre en œuvre puisque le concepteur lui-même peut ne pas avoir vu les « bugs » générés par le comportement attendu et le comportement inattendu de l'utilisateur. Elle permet toutefois des simulations en ligne rapides, des tests et l'évaluation ergonomique. On peut éviter l'intégration de cette composante en interfaçant le modèle utilisateur directement avec le dispositif. Toutefois, dans ce cas, la simulation risque d'être dans des impasses lorsque l'action du modèle de l'utilisateur n'est pas comprise par le dispositif ou lorsque la réponse du dispositif ne correspond pas à l'attente du modèle. Une autre solution est celle qui consisterait à mimer l'interaction humain-machine. Il s'agit d'une méthode pratique pour l'élaboration du modèle, mais qui n'est pas méthodologiquement valide dans la mesure où le concepteur du modèle est amené à le modifier en cours de route. Le modèle doit être complètement finalisé au début de la simulation.

Cette composante peut avoir la forme du graphe des états de la figure 1, ou y être équivalent. Elle permet des tests, mais aussi des comparaisons intra-dispositif entre procédures alternatives, de prévoir les procédures erronées, et des comparaisons inter-dispositifs aux mêmes fonctionnalités. La taille de l'espace des états est par exemple un indicateur. Un problème concret est, par exemple, l'approximation de la taille de l'espace sans avoir à le construire.

2.2 L'apprenabilité

Idéalement, l'apprenabilité d'un dispositif devrait pouvoir se mesurer par la taille de l'espace des états et être, par exemple, proportionnelle au nombre d'états. C'est loin d'être le cas, comme le montre n'importe quel « casse-tête », mais aussi les études en résolution de problème. Savoir utiliser un dispositif relève de l'apprentissage. La somme des connaissances à acquérir et le temps d'apprentissage sont des indicateurs à relativiser avec le gain apporté par l'apprentissage. Il peut être économique d'apprendre beaucoup pour obtenir beaucoup. Cet apprentissage peut déjà avoir été fait sur d'autres dispositifs. Pour un utilisateur donné, on peut être informé sur ce qu'il connaît déjà de l'utilisation d'autres dispositifs, pour mesurer ce qui lui reste à apprendre. On se base ainsi sur un transfert analogique positif du savoir relatif à des dispositifs « source », de référence, connus de l'utilisateur, vers le nouveau dispositif « cible » qu'il aura à utiliser. Il faut toutefois que l'utilisateur reconnaisse dans le nouveau dispositif ce qu'il connaît déjà. Cette connaissance des dispositifs « source » permet aussi de prédire le transfert « négatif » à partir de fausses reconnaissances, c'est-à-dire lorsque l'utilisateur reconnaît dans le dispositif cible des éléments de dispositifs source qui n'y sont pas.

Une autre difficulté à mesurer l'apprenabilité provient des contraintes implicites définies par Richard, Poitrenaud & Tijus (1993). Pour un état du dispositif, il y a les actions illicites et les actions licites. Parmi les actions illicites, il y a celles que l'utilisateur juge illicites et celles qu'il juge licites. Ces dernières sont facilement décelables, voire prédictibles à partir d'hypothèses basées sur transfert analogique négatif. Parmi les actions licites, il y a celles que l'utilisateur juge licites et celles qu'il juge illicites. Ces dernières sont difficilement décelables car il est difficile de savoir si une action licite qui n'est pas faite, n'est pas faite intentionnellement ou si c'est parce que l'utilisateur pense qu'il s'agit d'une erreur à éviter. Il s'agit d'un véritable frein à l'apprentissage puisque l'utilisateur évitera d'explorer toute une partie de l'espace des états. Un second problème est le diagnostic de ces contraintes implicites, à base du modèle des contraintes de Richard (Richard et al., 1993).

2.3 L'utilisabilité

Pour un état d'apprentissage donné, l'utilisabilité relève de la plus ou moins grande facilité d'application des procédures connues. L'utilisabilité dépend d'un certain nombre de facteurs internes et externes : retrouver la procédure en mémoire à long terme, la maintenir en mémoire à court-terme dans le focus attentionnel, non-concurrence en tâches pour éviter les situations de doubles tâches, processus de décision quand plusieurs actions ou procédures sont possibles, complexité de la procédure, etc.

L'utilisabilité est mesurable expérimentalement en recueillant le temps de réalisation de tâches et les erreurs commises, ceci pour un ensemble de tâches qui couvre la mise en œuvre des fonctionnalités du dispositif. Elle devrait pouvoir être prédite en comparaison à une interaction « idéale », sans erreur, c'est-à-dire une exécution réalisée par un "expert idéal", une notion qu'on trouve avec le modèle CAMERA (Leonardo, Tabachneck, & Simon, 1995).

3 Conclusion : apports de la modélisation sémantique

Notre approche « sémantique de l'utilisabilité », basée sur la catégorisation, qui correspond au quatrième mode de représentation des connaissances, sous la forme d'un treillis de Galois permet de modéliser et simuler les procédures utilisateurs sur un dispositif technique et fournit un ensemble de mesures telles que (i) la complexité d'un dispositif, définie par le produit du nombre de catégories par le nombre de procédures : $COMP.dev = N_c \times N_p$, (ii) l'applicabilité d'une procédure, définie par sa portée ou extension, ce qui veut dire par le nombre de catégories auxquelles elle s'applique : $EXT.proc = N_c.proc$, (iii) la puissance d'une procédure, définie par le rapport entre son extension et le nombre total de catégories : $POW.proc = EXT.proc / N_c$, (iv) la puissance du dispositif, définie comme la somme de la puissance de chaque procédure : $POW.dev = \sum POW.proc$, et (v) L'efficacité du dispositif, définie comme le rapport entre puissance et complexité, $EFF.dev = POW.dev / COMP.dev$.

Enfin, comme l'apprentissage et l'utilisation ne sont pas dissociés, une version dynamique permet le déplacement dans le graphe d'états à partir d'un treillis de Galois dynamique qui décrit la représentation évolutive de la situation, elle-même évolutive. Le treillis est construit en considérant les propriétés des objets dans l'état antécédent, qu'on vient de quitter, dans l'état courant, où l'on est et dans l'état but, où l'on veut être. $P(abc)$ représente les propriétés communes aux objets dans les trois états, $P(ab)$ celles qui sont communes aux états antécédent et but, $P(ac)$ aux états antécédent et courant, $P(a)$ la catégorie des propriétés spécifiques des objets de l'état antécédent, $P(b)$ celles qui sont spécifiques à l'état but et $P(c)$ à l'état courant. Les catégories dans le réseau correspondent respectivement : $P(a)$ et $P(ac)$: aux propriétés qui ne doivent pas être obtenues, $P(c)$: aux résultats obtenus, de nouvelles propriétés différentes de celles du but, $P(bc)$: aux propriétés communes à l'état courant et au but, $P(b)$: aux propriétés propres au but qui diminuent en nombre au fur et à mesure de la résolution.

Le modèle permet ainsi de simuler l'utilisation par les actions liées à la catégorie $P(bc)$, dont l'affordance lorsqu'une seule action est attachée à $P(bc)$, mais aussi l'apprentissage à travers l'exploration qui consiste ici à se débarrasser des propriétés indésirables liées à $P(c)$ et $P(ac)$.

Références

- Anderson, J. R. (1983). *The architecture of cognition*. Cambridge, MA: Harvard University Press.
- Barcenilla, J., Leproux, C., & Poitrenaud, S. (2002). *Analogie et Ergonomie Cognitive : Expérimentation and Simulation. Métaphore et Analogie. Traité des Sciences Cognitives*. Paris: Hermes
- Fjeld, L. , Schlupe, s. & Rauterberg, M. (1999): Action-driven quantification of task-solving behaviour. In D. Harris (ed.) *Engineering Psychology and Cognitive Ergonomics*, Vol. 4. Hampshire: Ashgate, pp. 253-261.
- Leonardo, A., Tabachneck, H., & Simon, H. A; (1995). Computational Model of Diagram Reading and Reasoning. Proceedings of the 17th Annual Conference of the Cognitive Science Society. July 1995.
- Mannes, S. M., & Kintsch, W. (1991) Routine computing tasks: planning as understanding. *Cognitive Science*, 15, 305-342.
- Nguyen-Xuan, A., & Hoc, J.M. (1987). Learning to use a command device. *Cahiers de Psychologie Cognitive*, 7, 5-31.
- Poitrenaud, S. (1995). The Procope Semantic Network: an alternative to action grammars. *International Journal of Human-Computer Studies*, 42, 31-69.
- Poitrenaud, S., Richard, J.F., & Tijus, C.A., (sous presse). Properties, Categories and Categorization. *Thinking and Reasoning*
- Richard, J. F., Poitrenaud, S., & Tijus, C. (1993). Problem solving restructuration : Elimination of implicit constraints. *Cognitive Sciences*, 17, 497-529.
- Richard, J.F., Clément, & Tijus, C.A. (2002). Les différences de difficulté dans la résolution des problèmes isomorphes comme révélatrices des composantes sémantiques dans la construction de la représentation du problème. In Pitrat, J. (Ed.), *hommage à H. Simon*. *Revue d'Intelligence Artificielle*, 16, 191-219
- Tijus, C.A., Poitrenaud, S., & Richard, J.F. (1996). Propriétés, Objets, Procédures: les Réseaux Sémantiques d'Action appliqué à la Représentation des Dispositifs Techniques. *Le Travail Humain*, 59, 209-229

Summary

Semantics of usability is a categorization based approach using Galois Lattices for modeling user procedures in human-computer interaction. This approach differs from operator's model based on rules production, such as Soar or Act, by associating actions and procedures to device objects categories as functional properties. Accessing actions and procedures is done while accessing objects categories. In the model, errors are semantic mistakes and the analogy concerns the processes of recognition that take place at the time of categorization. Modeling and simulation within the framework of this approach are carried out with the formalisms developed by Poitrenaud (1995): ProcOpe and STONE.