

**VERS UNE ÉVALUATION ANALYTIQUE DES INTERFACES HOMME
MACHINE DÉVELOPPÉES DANS LE CONTEXTE DES HABITATS
INTELLIGENTS**

par

Belkacem Chikhaoui

**mémoire présenté au Département d'informatique
en vue de l'obtention du grade de maître ès sciences (M.Sc.)**

**FACULTÉ DES SCIENCES
UNIVERSITÉ DE SHERBROOKE**

Sherbrooke, Québec, Canada, mai 2009



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-53152-5
Our file Notre référence
ISBN: 978-0-494-53152-5

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

Le 8 juillet 2009

le jury a accepté le mémoire de M. Belkacem Chikhaoui dans sa version finale.

Membres du jury

Mme Hélène Pigot
Directrice
Département d'informatique

M. Yves Bouchard
Membre
- Faculté de théologie, d'éthique et de philosophie

M. Shengrui Wang
Président-rapporteur
Département d'informatique

Sommaire

Concevoir des interfaces homme-machine en respectant les normes ergonomiques et suivant des démarches rigoureuses et systématiques constitue une préoccupation majeure pour les concepteurs des systèmes informatiques. Le besoin accru des interfaces accessibles et facilement utilisables a poussé les chercheurs dans ce domaine à créer des méthodes et des modèles qui permettent d'évaluer ces interfaces tout en mettant l'accent sur les aspects d'utilité et d'utilisabilité. Deux approches différentes sont actuellement utilisées pour évaluer les interfaces homme machine, des approches empiriques qui nécessitent l'association de l'utilisateur dans tout le processus de développement de l'interface et des approches analytiques qui ne font pas nécessairement appel à l'utilisateur pendant le processus de développement de l'interface. L'objectif de ce projet de maîtrise est d'évaluer analytiquement et simuler l'interface homme machine d'un assistant contextuel domiciliaire (ACD), développé pour assister les personnes atteintes de troubles cognitifs à réaliser les tâches de la vie quotidienne. Cette évaluation est basée sur trois méthodes analytiques largement utilisées dans ce domaine, qui sont : l'architecture cognitive ACT-R, le modèle GOMS et la loi de Fitts. Ces méthodes nous permettront d'évaluer des interfaces en mettant l'accent d'une part, sur l'aptitude de ces méthodes à prédire le temps d'exécution des tâches pour atteindre des buts spécifiés, et d'autre part, sur l'analyse et la description des tâches impliquées dans la réalisation de ces buts.

Afin de valider les trois modèles développés, les résultats obtenus ont été comparés avec des données provenant d'une étude expérimentale menée au sein du laboratoire DOMUS.

Remerciements

Je tiens à remercier ma directrice de recherche la professeure H  l  ne Pigot pour son immense patience, pour son appui financier et sa disponibilit   tout au long de la p  riode de r  alisation de ce travail. Elle demeure une professeure envers qui j'ai beaucoup de respect.

Je remercie   galement les professeurs du d  partement d'informatique avec qui j'ai pu apprendre les concepts qui m'ont permis d'entreprendre ce travail avec confiance.

Je ne peux oublier de remercier ma famille et mes amis pour leur soutien et leurs encouragements durant toute cette p  riode.

Je tiens    remercier toute l'  quipe du laboratoire DOMUS, professeur(e)s et   tudiant(e)s, pour les discussions enrichissantes des midi-DOMUS.

Je veux aussi exprimer ma gratitude    tout le personnel de la facult   des sciences et du d  partement d'informatique de l'Universit   de Sherbrooke.

Table des matières

Sommaire	ii
Remerciements.....	iii
Table des matières	iv
Liste des abréviations.....	vi
Liste des tableaux.....	vii
Liste des figures	viii
Introduction.....	1
Problématique	1
Objectifs.....	10
Méthodologie.....	12
Structure du mémoire.....	14
Chapitre 1 Aperçu sur les méthodes d'évaluation d'interfaces homme machine	15
1.1 Historique d'interaction homme machine.....	15
1.1.1 Définition du terme <i>interface</i>	19
1.2 Évaluation d'interfaces homme machine.....	20
1.2.1 Utilité et utilisabilité	21
1.3 Approches d'évaluation d'interfaces homme machine.....	23
1.3.1 Approches empiriques	23
1.3.2 Approches analytiques.....	24
1.4 Choix des méthodes d'évaluation d'interfaces	25

1.4.1	Architectures cognitives.....	27
Chapitre 2 Évaluation des interfaces de l'ACD utilisant ACT-R et la loi de Fitts		
.....		30
2.1	Avant-propos.....	30
2.2	Résumé de l'article	30
Chapitre 3 Évaluation des interfaces de l'ACD utilisant ACT-R et GOMS		40
3.1	Expérimentation.....	40
3.2	Ajustement du modèle ACT-R	41
3.3	Le modèle GOMS	42
3.4	Avant-propos de l'article	43
3.5	Résumé de l'article	44
Chapitre 4 Résultats et discussion		56
4.1	Résultats.....	56
4.2	Discussion.....	59
4.2.1	Prédiction au niveau des objets.....	59
4.2.2	Prédiction au niveau des tâches	60
Conclusion		63
Bibliographie.....		65

Liste des abréviations

ACT-R	Adaptative Control of Thought, Rational
ACD	Assistant Contextuel domiciliaire
ACM	Association for Computing Machinery
AVQ	Activités de la Vie Quotidienne
CAPS	Collaborative Activation-based Production System
CCT	Cognitive Complexity Theory
DI	Déficienne Intellectuelle
DOMUS	DOMotique et informatique Mobile de l'Université de Sherbrooke
EPIC	Executive Process Interactive Control
GOMS	Goals, Operators, Methods and Selection rules
HMI	Human Machine Interaction
ISO	International Organization for Standardization
KLM	Keystroke Level Model
LICAI	The LInked model of Comprehension-based Action planning and Instruction taking
PDA	Personal Digital Assistant
SOAR	State, Operator And Result

Liste des tableaux

Tableau 1 - Étude comparative entre les différentes versions de GOMS	43
Tableau 2 - Résultats obtenus par l'étude expérimentale et les trois modèles.....	56
Tableau 3 - Prédiction du temps d'exécution des tâches dans chaque modèle.....	60

Liste des figures

Figure 1 - Modèle d'acceptabilité du système selon Nielsen	21
Figure 2 - Dimension de l'évaluation de l'IHM selon Senach	23
Figure 3 - Exemple de déroulement des expérimentations.....	41
Figure 4 - Prédiction de temps de sélection des objets de la première tâche.....	58
Figure 5 - Prédiction de temps de sélection des objets de la deuxième tâche	58
Figure 6 - Prédiction du temps d'exécution des tâches dans chaque modèle	61
Figure 7 - Exemple d'un objet difficilement identifiable	62

Introduction

Problématique

Dans les pays occidentaux, la baisse de la natalité et l'augmentation de l'espérance de vie se traduisent par la modification de la pyramide des âges et une augmentation de la proportion des personnes âgées. Selon le rapport de la deuxième assemblée mondiale sur le vieillissement qui s'est déroulée à Madrid en avril 2002, le nombre des personnes âgées devrait d'ici 2050 dépasser celui des jeunes et ce pour la première fois dans l'histoire de l'humanité (Radio des Nations Unies, 2008).

Nous savons, d'après les statistiques, qu'il existe aujourd'hui plus de 629 millions de personnes âgées de plus de 60 ans; et que, selon les prévisions, leur nombre dépassera les 2 milliards en 2050. La proportion des personnes âgées n'a cessé de croître durant le 20^{ème} siècle, et l'on prévoit une poursuite de cette tendance. Elle est passée de 8 % en 1950 à 10 % en 2000 et devrait atteindre 21 % en 2050 (Radio des Nations Unies, 2008). Au Canada, les données de recensement provenant de Statistique Canada montrent d'importants changements dans la répartition de la population canadienne selon l'âge en raison du vieillissement démographique. Selon les données issues du recensement de 2006, le nombre de personnes âgées de 65 ans et plus a dépassé pour la première fois la barre des 4 millions. Ainsi, entre 2001 et 2006, la proportion des personnes âgées est passée de 13 % à 13,7 % (Statistique Canada, 2006).

Pour l'individu comme pour la société, l'une des principales préoccupations associées à la vieillesse est le risque de dépendance associé à la maladie. Cependant, en raison du problème grandissant posé par les soins de plus en plus coûteux, et des besoins de plus en plus accrus en terme d'assistance, on craint qu'une population vieillissante n'entraîne une charge supplémentaire pour les services de santé déjà surchargés. En réalité, seulement une faible

proportion de personnes âgées environ 20 % a très souvent recours aux services de santé officiels bien qu'un grand nombre de canadiens âgés souffrent d'une maladie chronique. Les enquêtes démontrent que moins de la moitié de ces personnes souffrent d'une incapacité physique ou intellectuelle qui limite leurs activités quotidiennes, et moins du tiers ont des problèmes de santé nécessitant des soins médicaux (Réalités Canadiennes, 2008).

Outre les maladies chroniques, c'est sans doute l'affaiblissement intellectuel que les personnes âgées craignent le plus. Ceci engendre une perte d'autonomie dans l'accomplissement des activités de la vie quotidiennes, qui entraîne une augmentation progressive du niveau de dépendance. Des statistiques récentes montrent qu'environ 450 000 canadiens âgés de plus de 65 ans sont atteints de la maladie d'Alzheimer et d'affections connexes. En 2008, on dénombre 300 000 Canadiens atteints de la maladie d'Alzheimer, et on prévoit qu'environ 750 000 canadiens seront atteints de la maladie d'Alzheimer et d'affections connexes en 2031 (Société Alzheimer, 2006). La maladie d'Alzheimer est une maladie neurodégénérative qui se caractérise par un certain nombre de symptômes comme une détérioration progressive et continue de la mémoire, l'incapacité d'accomplir des tâches familières, la modification du jugement et du raisonnement ou encore des changements d'humeur et du comportement. Elle est la forme la plus commune de maladies neurodégénératives liées à l'âge et est responsable de 64 % de tous les cas de maladie neurodégénérative au Canada (Société Alzheimer, 2006).

Selon les statistiques de l'institut de vieillissement du Canada¹, il est estimé que 16 % des personnes âgées de plus de 65 ans souffrent de troubles cognitifs, 8 % d'entre elles sont atteintes de maladies neurodégénératives comme la démence. Cette prévalence augmente de façon exponentielle avec l'âge, atteignant 30 % (troubles cognitifs) et 35 % (démence) chez les personnes de plus de 85 ans. De tels problèmes menacent la qualité de vie de ces personnes.

¹ <http://www.cihr-irsc.gc.ca/f/10332.html>

Au Canada, cependant, les services de santé sont essentiellement conçus pour le traitement à court terme des maladies aiguës. Cela pose une question très importante : qui s'occupe des personnes âgées et plus précisément des personnes présentant des troubles cognitifs ?

Malgré l'évolution récente de la famille canadienne, on estime que les proches parents et les amis apportent environ 80 % de tous les soins aux personnes âgées. Cette aide prend diverses formes : travaux ménagers, préparation des repas, transport, etc. Même les personnes placées dans des établissements de soins de longue durée continuent généralement à bénéficier de l'aide de leurs familles et amis (Statistique Canada, 2006). La dépendance est la première conséquence de la vieillesse. Physique et/ou mentale, la dépendance empêche la personne âgée de vivre seule dans un logement ordinaire. Elle se traduit soit par le placement de la personne en institution, soit par le besoin d'un soutien ou aide d'un proche.

Durant les années à venir, l'augmentation du nombre de personnes souffrant de troubles cognitifs ou d'autres maladies liées à l'âge va aggraver la surcharge de travail et la pénurie de personnels soignants (infirmières, médecins, spécialistes, assistants sociaux etc.) dans les établissements médico-sociaux.

Pour faire face au problème de surcharge des établissements et pour répondre aux souhaits des malades de rester chez eux, le maintien à domicile paraît comme une solution socio-économique en faveur de ce type de population. Toutefois, pour rester plus longtemps à domicile dans de bonnes conditions, quatre critères doivent être réunis. Ils concernent la santé, l'environnement familial, le niveau de ressources et l'habitat (Simon et Fronteau, 1996).

- **L'état de santé :** c'est le critère le plus important dans le choix du maintien à domicile. En effet, la dégradation de l'état de santé, entraînant une incapacité totale ou partielle, implique un placement en institution. Les troubles physiques ne sont pas toujours un obstacle au maintien à domicile. Or, les troubles mentaux sont très difficiles à prendre en charge.

- **La famille** : elle constitue la clé du maintien à domicile malgré le niveau de dépendance. Pour favoriser le maintien à domicile, les professionnels viennent compléter l'aide familiale sans pour autant s'y substituer.
- **Le niveau de ressources (coûts)** : le maintien à domicile n'est pas toujours possible en raison du coût d'une aide professionnelle rémunérée. En France par exemple, au moment de l'évaluation de la dépendance (dans le cadre de la Prestation Expérimentale Dépendance), les coûts pour les cas les plus lourds ont été estimés à plus de 12 000 F par mois) (Simon et Fronteau, 1996).
- **L'habitat** : s'il est inadapté, il peut accélérer le placement en institution. L'adaptation du logement permet à la personne âgée de conserver une partie de son autonomie.

Au début, l'idée du maintien à domicile supposait un déplacement régulier du personnel médical pour fournir aux malades des soins médicaux ou paramédicaux de qualité similaires à ceux donnés dans les institutions de soins. Toutefois, cela exige un soutien très fort de la part de la famille. Une solution consiste alors à impliquer l'environnement physique pour qu'il assiste la personne. Cette solution introduit un nouveau système de maintien à domicile dans lequel l'environnement joue un rôle important dans le système de soins. Grâce aux nouvelles technologies de l'information et de la télécommunication, l'environnement est devenu l'interlocuteur entre la personne à son domicile et ses soignants qui se trouvent sur des sites différents, voire un acteur dans le système de soins. Cette tendance peut être garantie par la mise en place d'un système informatique, qui surveille la personne et n'alerte le personnel médical qu'en cas d'urgence. Les soins fournis à domicile constituent une alternative à l'hospitalisation et au recours aux établissements d'hébergement de longue durée. Mais le problème des personnes atteintes de troubles cognitifs n'est pas seulement de leur offrir des soins médicaux, mais plutôt de les surveiller et de les assister dans les activités de la vie quotidienne. Cette problématique a donné naissance au concept d'habitat intelligent.

Un habitat intelligent est défini par sa capacité à réagir à ce qui se passe dans son environnement. Il est capable d'identifier toute situation inhabituelle ou inadéquate et de

fournir l'aide appropriée à l'occupant en cas de besoin. En d'autres termes, un habitat intelligent peut s'adapter aux besoins de l'occupant. Plusieurs projets de recherche sur les habitats intelligents sont lancés à travers le monde afin de favoriser le maintien à domicile.

La télémédecine ou la télésanté est l'exemple le plus connu de l'utilisation des technologies de l'information pour offrir et gérer des soins à distance. Plus spécifiquement, l'avènement des nouvelles technologies de l'information et de télécommunications permet d'envisager de nouvelles façons d'exercer la médecine et d'offrir des services médicaux spécialisés. Selon le rapport publié par le Collège des Médecins du Québec, la télémédecine est définie comme étant « *l'exercice de la médecine à distance à l'aide de moyens de télécommunications* » (Direction de l'amélioration de l'exercice, 2000).

Au Québec, la pratique de la télémédecine est présente et expérimentale. Par exemple, au centre hospitalier Îles-de-la-Madeline, un médecin et son patient peuvent consulter en temps réel, un dermatologue à Gaspé, un cardiologue à Rimouski ou encore un neurochirurgien à Québec. Ils sont reliés par un réseau de communications socio-sanitaires qui relie tous les hôpitaux au Québec (Radio-Canada, 2002). Selon (Bajolle, 2002), les principales applications de la télémédecine sont :

- Télé-consultation et télédiagnostic : consultation médicale à distance
- Télésurveillance ou surveillance à distance d'un patient
- Télé-expertise : avis donné à distance par un expert ou un médecin
- Télé-formation : consultation des informations médicales (bases de données, imageries, cours de formation, etc.)
- Télé-chirurgie : permet de manipuler de matériel médical à distance et d'avoir une action directe du praticien sur le patient.

La télésurveillance est une branche centrale de la télémédecine. Elle permet le suivi d'une personne âgée vivant seule dans son milieu de vie sans perturber ses habitudes de vie en

équipant son habitat par des capteurs (détecteurs de mouvement, de présence, d'ouverture, etc.). Il ne s'agit pas systématiquement de vidéo de surveillance ou de transmission d'images. Ainsi, avec l'évolution fulgurante de la technologie, l'habitat peut être équipé par différents types de capteurs tels que les capteurs sonores et médicaux (tensiomètre, oxymètre, balance, etc.) en vue de la détection d'une situation de détresse (Istrate, 2003). Les données issues des capteurs sont fusionnées pour diagnostiquer toute situation anormale. Grâce aux données provenant des capteurs médicaux, le traitement des données recueillies permet ensuite de mesurer d'autres paramètres physiologiques et comportementaux afin d'identifier le mode de vie de la personne.

Avec la télésurveillance médicale, les personnes âgées conservent une large autonomie dans leur environnement privé et social tout en bénéficiant des services de santé. Ainsi, la télésurveillance médicale des personnes au domicile représente une alternative à l'hospitalisation et au recours aux établissements d'hébergement de longue durée.

Dans les systèmes de télémédecine en général, l'aide fournie aux personnes provient toujours de professionnels de santé qui sont à l'extérieur du domicile. Par contre, l'assistance provient de l'intérieur de l'habitat en analysant et interprétant les données issues des différents capteurs par un système informatique intelligent. En effet, l'habitat intelligent perçoit les actions de la personne et analyse l'adéquation de ce qui a été fait et ce qui devrait l'être. On distingue deux types d'assistance selon la nature du déficit de la personne. Une assistance physique et une assistance cognitive.

Assistance physique : l'assistance physique consiste à compenser le déficit physique (handicap) de la personne en utilisant des moyens appropriés (commande à distance, chaise roulante, robot mobile, bras manipulateur, etc.).

Assistance cognitive : l'assistance cognitive consiste à fournir de l'aide à la personne pour pallier aux déficits cognitifs causés par les maladies telles que l'Alzheimer, la schizophrénie, les traumatismes crâniens et la déficience intellectuelle. Le but fondamental de l'assistance cognitive est donc de promouvoir l'autonomie de ces personnes par des méthodes

compensatoires. L'assistance cognitive permet de rappeler au besoin, les activités à réaliser et les procédures pour les faire. Toutefois, dans des situations délicates, une aide extérieure est nécessaire particulièrement si les risques sont trop élevés (Pigot et al., 2007). Le système d'assistance cognitive doit être en mesure d'analyser les différentes situations, ce qui peut être réalisé par l'intégration de mécanismes de reconnaissance d'activités, d'apprentissage et de raisonnement très pertinents.

Les habitats intelligents sont basés sur le paradigme de l'informatique diffuse, ou ubiquitaire, introduite par (Weiser, 1991). Dans ce paradigme, la technologie devient invisible à la personne avec laquelle elle entretient des interactions permanentes (Weiser, 1993). Ainsi, en éliminant les interfaces homme machine trop complexes pour des personnes ayant des troubles cognitifs, l'informatique diffuse permet des interactions directes entre la personne et son environnement par l'intermédiaire d'objets interactifs (Pigot et al., 2007).

Le laboratoire DOMUS (DOMotique et informatique Mobile de l'Université de Sherbrooke)² est un laboratoire multidisciplinaire dédié à la recherche en domotique et en informatique mobile. Les problématiques de recherche attachées au laboratoire DOMUS sont complexes et diversifiées tant au plan informatique (systèmes répartis, personnalisation, informatique diffuse, informatique omniprésente, modélisation cognitive, interaction homme machine, etc.) qu'au plan multidisciplinaire (ergothérapie, psychologie, psychoéducation, éthique, santé, etc.).

Le laboratoire DOMUS dispose d'un appartement intelligent à la fine pointe de la technologie où l'on retrouve un équipement domiciliaire de base. Il s'agit d'un 4^{1/2} construit à l'intérieur des murs de l'université, pouvant loger une personne seule. Cet appartement est équipé d'une technologie de pointe dispersée dans l'environnement. Les capteurs, haut-parleurs, écrans d'ordinateurs et de télévision, les réseaux filaires et non filaires et les serveurs transforment cet appartement en un environnement intelligent capable de s'adapter aux actions de l'occupant. Ce type d'appartement s'adresserait à des personnes présentant des

² <http://domus.usherbrooke.ca>

troubles cognitifs, qui les empêchent à mener à bien les activités de la vie quotidienne. Par conséquent, les projets du laboratoire DOMUS s'adresseraient à une population bien ciblée comme les personnes atteintes de démence de type Alzheimer, les clientèles de type schizophrène, traumatisé crânien, etc.

L'objectif du laboratoire DOMUS est double : d'une part, aider les personnes atteintes des troubles cognitifs à vivre normalement en palliant leurs déficiences et, d'autre part, surveiller à long terme l'évolution de l'état de santé de la personne et prévenir rapidement le personnel qualifié en cas de problèmes (feu, dégâts d'eau, chute, malaise, etc.). Pour répondre à ces objectifs, le système d'appartement intelligent doit comporter deux principaux modules : le module d'assistance cognitive, chargé de l'aide à la réalisation des activités de la vie quotidienne (AVQ), et le module de télésurveillance, chargé de suivre l'état de santé de l'occupant (Pigot et al., 2003).

En ce qui concerne la partie d'assistance cognitive, il s'agit de pallier les troubles cognitifs de l'occupant, de manière à ce qu'il puisse réaliser l'ensemble des tâches de la vie quotidienne. Cette aide nécessite une bonne connaissance des habitudes de vie de la personne et l'élaboration d'un modèle de ses compétences en fonction de sa maladie, afin que l'aide fournie soit appropriée et adaptée. Le système d'assistance cognitive est conçu de telle sorte que la personne exerce ses fonctions cognitives d'une manière autonome, et que le système n'intervienne qu'en cas de besoin, afin de laisser l'initiative à la personne et d'éviter d'aggraver ses pertes cognitives.

Pour favoriser l'autonomie et le maintien à domicile des personnes avec des déficits cognitifs, un assistant contextuel domiciliaire (ACD) est développé au laboratoire DOMUS. L'ACD est une application développée pour assister les personnes à réaliser leurs tâches de la vie quotidienne en leur rappelant la localisation des objets et les étapes à faire en cas de besoin. Les tâches de préparation des repas ont été choisies pour plusieurs raisons. D'abord il s'agit d'une composante centrale dans la vie résidentielle. C'est une activité complexe qui exige une multitude de compétences et d'habiletés spécifiques (planification, résolution de problèmes, raisonnement). L'ACD est intégré dans l'appartement intelligent qui recueille de

l'information sur les actions de l'utilisateur par l'entremise de capteurs et renvoie l'information par des écrans tactiles et des hauts parleurs. Il a été montré que l'ACD aidait les personnes avec déficience intellectuelle (DI) à cuisiner des recettes complexes. Toutes les informations recueillies par les capteurs sont analysées par l'ACD pour pouvoir déduire par un système de reconnaissance d'activités si la personne a réalisé correctement ses activités ou s'il a besoin d'assistance (Lussier-Desrochers et al., 2007; Pigot et al., 2007).

L'assistance est disponible dans l'ACD selon plusieurs modes : aide pas à pas présentée sous forme d'images et de séquences vidéos ou aide déclenchée par le système lorsque celui-ci détecte qu'une activité est réalisée de façon inadéquate. Si la personne réalise ses activités correctement, le système n'interviendra en aucun moment. L'assistance fournie par l'ACD permet de pallier les troubles de la personne en l'aidant pas à pas à la planification des tâches et d'étendre cela à d'autres tâches liées à la vie de la personne.

Une composante importante de l'ACD est l'assistance cognitive dispensée sur un écran tactile déposé sur une surface de travail bien choisie dans la cuisine. Dans l'ACD, l'écran tactile constitue le seul moyen de communication directe avec le système. Étant donné son rôle stratégique dans l'assistance et la population ciblée, il s'avérait important de concevoir des interfaces simples et conviviales, qui permettent de fournir des informations pertinentes à la personne. Les interfaces de l'ACD ont été conçues suivant les recommandations des chercheurs et spécialistes en lien avec la création d'interfaces utilisateurs spécifiquement destinées à des personnes présentant des déficits cognitifs (Davies et al., 2001). Il a été recommandé l'utilisation d'interfaces simples comprenant des boutons surdimensionnés pour faciliter la navigation à l'aide de l'écran tactile.

Afin que les interfaces de l'ACD soient accessibles, facilement utilisables par la population ciblée, efficaces et efficientes, on doit procéder à leur évaluation en utilisant des méthodes et des techniques qui s'adaptent avec le contexte pour lequel ces interfaces ont été conçues.

L'évaluation des interfaces homme-machine constitue un problème épineux pour les concepteurs et les designers d'interfaces. Elle est le plus souvent jugée trop coûteuse. Ceci

dépend de la méthode ou l'approche choisie pour mener à bien le processus d'évaluation. La classification la plus fréquente des approches d'évaluation d'interfaces est fondée sur la distinction entre : approches empiriques basées sur les tests d'utilisateurs et des approches analytiques basées sur des méthodes d'analyse rigoureuses (Nielsen et Molich, 1990; Senach, 1990).

Afin de mettre en œuvre une évaluation empirique, plusieurs paramètres doivent être réunis : un nombre suffisant d'utilisateurs (représentatifs des utilisateurs finaux) avec lesquels se déroulent les expérimentations, un environnement reproduisant l'environnement réel des utilisateurs et une infrastructure matérielle et logicielle. L'évaluation empirique requiert par la suite beaucoup de temps ainsi qu'un budget suffisamment important pour le bon déroulement des expérimentations. Contrairement à l'évaluation empirique, l'évaluation analytique permet de surmonter les problèmes liés à l'évaluation empirique en utilisant des méthodes, des outils et des modèles suffisamment sophistiqués qui permettent de minimiser considérablement les coûts et le temps d'évaluation.

Dans le contexte d'évaluation d'interfaces homme machine conçues dans le cadre des habitats intelligents, la problématique de notre projet consiste à évaluer analytiquement les interfaces de l'ACD en mettant l'accent sur le temps d'exécution des tâches requises lors de l'interaction de l'utilisateur avec ces interfaces. La démarche et les solutions proposées doivent permettre de dessiner un cadre théorique pour cette problématique, et de mettre une base solide pour un processus complexe d'évaluation d'interfaces de l'ACD en prenant en compte : l'environnement de l'utilisateur et les erreurs que ce dernier peut commettre lors de la réalisation de ses tâches en interagissant avec l'interface de l'ACD.

Objectifs

Pour fournir une assistance cognitive adéquate, le laboratoire DOMUS dispose des moyens matériels et logiciels qui permettent de détecter et de recueillir des informations sur l'activité en cours de réalisation, en prenant en compte les actions et déplacements de l'utilisateur dans

l'habitat intelligent. Cette infrastructure matérielle et logicielle permet également de fournir une aide appropriée lorsqu'un comportement inadéquat est détecté.

Afin d'assurer un bon déroulement et suivi des activités, et que l'aide appropriée soit fournie, le système doit également être en mesure de fournir des moyens et des outils qui facilitent l'interaction de l'utilisateur avec le système. Pour permettre une interaction simple et efficace avec le système, les interfaces homme machine développées et intégrées dans l'ACD, doivent être accessibles et utilisables sur le plan ergonomique et conceptuel.

Pour répondre à la problématique d'accessibilité et d'utilisabilité des interfaces, on procède à une évaluation analytique et rigoureuse de ces interfaces. Cette évaluation permettra d'améliorer entre autres le design, l'accessibilité, l'adaptabilité et l'utilisabilité de ces interfaces. En conséquence, le but de ce travail est d'évaluer analytiquement l'interface de l'ACD en prenant en compte les composantes perceptuelles, cognitives et motrices impliquées lors de l'interaction avec cette interface.

Notre évaluation est basée principalement sur trois méthodes analytiques : ACT-R, GOMS et la loi de Fitts. Ces méthodes requièrent une approche rigoureuse d'évaluation de l'interaction avec l'IHM. Cela peut être réalisé à l'aide d'une simulation de cette interaction. Par conséquent, trois objectifs sont donc poursuivis dans cette recherche :

- 1- Simuler l'interface de l'ACD et l'interaction de l'utilisateur avec cette interface.
- 2- Évaluer analytiquement cette interface en utilisant les trois méthodes mentionnées précédemment.
- 3- Valider les modèles analytiques développés en comparant les résultats simulés avec des résultats expérimentaux réalisés avec des humains.

Méthodologie

Afin de répondre à la problématique d'évaluation analytique des interfaces homme machine, on procède en premier lieu à une revue de littérature sur le domaine d'évaluation des interfaces homme machine et les méthodes analytiques utilisées dans l'évaluation de l'interaction avec ces interfaces. Cela nous amènera à choisir convenablement les trois méthodes analytiques utilisées dans notre projet de recherche.

Les interfaces de l'ACD sont évaluées durant la réalisation d'une activité particulière de la vie quotidienne. Le choix de l'activité est arrêté sur la préparation d'une recette complexe de cuisine, qui est la « préparation des spaghetti ». Cette recette est incluse dans le système d'assistance cognitive de l'ACD. Cette activité est choisie à cause de sa complexité de réalisation, et du nombre d'interfaces impliquées durant sa réalisation. Cette complexité entraîne l'augmentation du nombre d'erreurs commises par l'utilisateur lors de la réalisation de cette activité, et par conséquent, l'augmentation du nombre d'interventions de l'ACD pour remédier à ces erreurs.

La recette de préparation des spaghetti est composée de plusieurs étapes, les deux premières étapes consistent à faire sortir les ustensiles et les ingrédients respectivement, nécessaires pour la réalisation de l'activité, et les autres étapes expliquent la procédure de préparation en utilisant des photos et des vidéos explicatives.

Dans notre projet, l'évaluation est limitée aux interfaces requises pour la réalisation des deux premières étapes de l'activité, « Sortir tous les ustensiles » et « Sortir tous les ingrédients ».

Pour répondre au premier objectif, on procède à une description et analyse approfondie des tâches requises pour l'accomplissement des deux premières étapes de la recette. La phase d'analyse et de description des tâches est une phase primordiale dans le processus d'évaluation d'interfaces. Elle permet de simplifier l'évaluation en découpant les tâches en sous-tâches, et les buts en sous-buts, suivant une décomposition hiérarchique. Un but ne peut être atteint, que si tous ses sous-buts le sont.

La démarche suivie peut être résumée dans la section suivante :

Pour répondre au premier objectif, nous allons faire :

- 1- Revue de littérature sur le domaine d'évaluation d'interfaces homme machine, afin de dessiner les contours de notre projet et d'avoir une idée claire sur ce domaine.
- 2- Choix des méthodes analytiques utilisées dans notre projet de recherche : ACT-R, GOMS et la loi de Fitts.

Après avoir complété cette partie, on procède à la simulation de l'interface de l'ACD. La simulation de l'interface se fait uniquement avec la méthode ACT-R, les deux autres méthodes ne font pas appel à la simulation d'interface.

Pour répondre au deuxième objectif, on procède à une analyse détaillée de l'interface, afin de faciliter le processus d'évaluation. L'évaluation selon chaque méthode sera présentée.

Enfin, pour répondre au troisième objectif, on présente l'étude expérimentale réalisée au laboratoire DOMUS, et on effectue une comparaison entre les résultats simulés et les résultats réels, afin de valider nos résultats.

Les deux premiers modèles font partie des modèles cognitifs, et sont basés sur l'architecture cognitive ACT-R et le modèle GOMS. Le troisième modèle est basé sur la loi de Fitts, cette loi est basée sur un modèle mathématique linéaire.

Les expérimentations ont été menées au laboratoire DOMUS sur des sujets sains en manipulant l'interface de l'ACD. Chaque sujet réalise les deux premières étapes de la recette spécifiée. Le temps d'exécution des tâches est enregistré et récupéré à la fin de chaque expérimentation. Les données obtenues sont analysées et par la suite, comparées aux données simulées obtenues par les trois modèles développés.

Structure du mémoire

Après une présentation des objectifs du présent projet de recherche et la méthodologie adoptée, le chapitre 1 présente une introduction générale sur l'évaluation des interfaces homme machine, les approches utilisées dans l'évaluation ainsi que les principales méthodes utilisées dans la littérature. Le deuxième chapitre aborde l'évaluation des interfaces de l'ACD en utilisant deux méthodes analytiques : ACT-R et la loi de Fitts. Ce chapitre est représenté par un article scientifique publié dans le journal IJHIS (International Journal of Hybrid Intelligent Systems). Le troisième chapitre abordera l'évaluation des interfaces de l'ACD avec l'ajout d'un troisième modèle d'évaluation basé sur le modèle GOMS, et la réalisation d'une étude expérimentale au sein du laboratoire DOMUS afin de comparer les données simulées aux données expérimentales. Ce chapitre est représenté par un article scientifique publié dans le journal IJIT (International Journal of Information Technology). Enfin, le dernier chapitre présente une discussion générale avec une conclusion et les améliorations futures des modèles développés.

Chapitre 1

Aperçu sur les méthodes d'évaluation d'interfaces homme machine

Ce chapitre donne un aperçu du domaine de l'interaction homme machine et des différentes méthodes d'évaluation d'interfaces. Tout d'abord, l'historique de l'interaction homme machine est brièvement présenté et quelques définitions de bases sont introduites avant de présenter les différentes approches d'évaluation et les principales méthodes utilisées dans la littérature.

1.1 Historique d'interaction homme machine

Depuis qu'existent les ordinateurs, la question de l'interface avec l'ordinateur s'est posée. En 40 ans, l'interaction homme machine a rendu l'informatique accessible à un plus grand nombre de personnes, d'une façon de nul ne pouvait prédire. Que serait le visage de l'informatique aujourd'hui sans les interfaces graphiques ?

Si on considère que la programmation aux clés des premiers ordinateurs pouvait être qualifiée de « manipulation directe » au sens littéral du terme, et si l'on peut considérer l'invention des langages de programmation comme un moyen pour faciliter l'interaction avec les machines informatiques, ce sont bien les travaux de Ivan Sutherland sur SketchPad au début des années 1960 qui marquent le début de l'histoire de l'interaction homme machine.

SketchPad, développé par Ivan Sutherland au début des années 1960 et publié dans sa thèse de doctorat en 1963 (Sutherland, 1963), est considéré comme la première interface graphique.

SketchPad, développé au MIT Lincoln Laboratory³, est le premier système qui a utilisé un écran cathodique et un crayon optique pour permettre l'édition graphique de dessins techniques. Bien plus tard, en 1983, Ben Shneiderman appellera ce type d'interaction avec des objets représentés à l'écran « manipulation directe », par contraste avec l'utilisation systématique, jusqu'au début des années 1980, de langages de commandes obligeant à mémoriser les noms des commandes et des objets (Myers, 1998).

De nombreux concepts fondamentaux des interfaces graphiques ont pour origine SketchPad : désignation directe des objets à l'écran, retour d'information immédiat sous forme de lignes, placement des segments d'une figure par contraintes (parallèle, angle droit, etc.), zoom avant et arrière sur le dessin, etc. même au niveau de la mise en œuvre, les concepts sont modernes comme la représentation des objets graphiques en mémoire (Myers, 1998).

Sutherland développe SketchPad sur le TX-2, l'un des rares ordinateurs de l'époque utilisable en ligne : jusqu'à la fin des années 1970, la grande majorité des ordinateurs sont utilisés de façon non interactive, en traitement par lots (batch). Le TX-2 a 320 Ko de mémoire, deux fois plus performant que les plus gros ordinateurs commerciaux de l'époque. À la même époque, d'autres chercheurs utilisent le TX-2 pour réaliser d'autres interfaces, comme Genesys, le premier système d'animation de l'histoire créé par Ronald Baecker (Baecker, 1969).

Peu après, Ivan Sutherland devient l'un des innovateurs de l'infographie et de la réalité virtuelle. En 1967, alors qu'il est professeur à Harvard, il crée avec son étudiant Bob Sproull le premier casque de réalité virtuelle affichant des images de synthèse. Plus tard encore, il s'intéresse à la robotique, et crée l'entreprise Evans & Sutherland, célèbre dans les années 1980 pour ses systèmes graphiques haut de gamme (Carlson, 2003).

Parallèlement à la création d'interfaces homme machine, et en partant du fait que l'interface représente ce que les utilisateurs voient du système, l'évaluation d'interfaces a vu le jour avec l'apparition d'interfaces homme machine. Par contre, le paradigme d'évaluation et le système

³ <http://www.ll.mit.edu/>

à évaluer à cette époque sont totalement différents de la vision actuelle et future des interfaces homme machine. En effet, à travers les quatre dernières décennies, l'évolution de l'évaluation de systèmes en général passait par plusieurs périodes qui peuvent être distinguées selon le type d'utilisateur, le type d'évaluateur et le facteur limitant dans l'évaluation (Kaye et Sengers, 2007). Les principales périodes selon leur apparition dans le processus de développement de l'informatique sont présentées dans la section suivante.

1- Évaluation par des ingénieurs

- Les utilisateurs : des informaticiens et des mathématiciens
- Les évaluateurs : des ingénieurs
- Facteur limitant : la fiabilité du système

2- Évaluation par des informaticiens

- Les utilisateurs : des programmeurs
- Les évaluateurs : des programmeurs
- Facteur limitant : la vitesse de la machine

3- Évaluation par des psychologues expérimentés et des cognitiens

- Les utilisateurs : des vrais utilisateurs, l'ordinateur est un moyen et non pas une fin.
- Les évaluateurs : des psychologues expérimentés et des cognitiens
- Facteur limitant : ce que l'homme peut faire

4- Évaluation par les professionnels d'Interaction Homme Machine

Dans cette catégorie, l'évaluation est assurée par des professionnels d'utilisabilité qui croient en leur expertise. Les professionnels d'utilisabilité ont pris une décision de se concentrer sur le fait d'avoir des meilleurs résultats, indépendamment de savoir s'ils ont été prouvés expérimentalement ou non.

Les applications développées actuellement ont consacré une moyenne de 48% de code pour l'interface utilisateur (Myers et Rosson, 1992; Nielsen, 1993). Il semblerait donc justifié d'allouer une proportion raisonnable d'effort de développement d'applications pour assurer l'utilisabilité de ces interfaces.

Beaucoup de recherches sont actuellement menées dans le domaine des interactions homme machine sur des sujets aussi divers que la conception des systèmes de gestion des applications interactives, les interfaces auto-adaptatives et l'évaluation de la qualité des interfaces.

Il est tout à fait important de prendre connaissance de tous les concepts couverts par le terme « Interaction Homme Machine ». A cet effet, la définition donnée par l'ACM semblerait recouvrir tous les aspects du domaine de l'interaction homme machine. L'ACM définit le domaine de l'interaction homme machine comme «une discipline qui concerne la conception, l'évaluation et l'implémentation de systèmes interactifs de traitement de l'information destinés à être utilisés par des humains et avec l'étude des principaux phénomènes qui les entourent»⁴ (ACM., 1992). Dans cette définition, tout le processus de création des systèmes interactifs est représenté par les termes *conception*, *évaluation* et *implémentation*. Par conception, on entend les différentes approches comme le prototypage ou la conception itérative. L'évaluation concerne tout processus qui permet de déterminer dans quelle mesure un système interactif recouvre certains critères comme la facilité d'utilisation et d'apprentissage. En fin, l'implémentation consiste à étudier tous les outils nécessaires pour mettre en œuvre un système interactif.

⁴ Cette définition est une traduction de celle présentée dans le rapport de l'ACM.

La définition de l'ACM spécifie clairement la nature des systèmes étudiés. Il s'agit bien de systèmes principalement interactifs. Ces systèmes sont destinés à être utilisés par des humains, d'où l'importance accordée au concept d'interface homme machine.

La personne est vue dans le domaine d'interaction homme machine comme un système de traitement de l'information. L'information est acquise au moyen de dispositifs sensoriels et communiquée par des dispositifs comme la parole ou le toucher. Les aspects ergonomiques, comme la fatigue au travail, les limites cognitives et physiques de l'humain, sont étudiés de même que la conception du système.

Avant d'introduire la notion d'évaluation d'interfaces homme machine, il est tout à fait intéressant de donner quelques définitions de base qui permettent de donner une idée claire sur le terme d'évaluation d'interfaces.

1.1.1 Définition du terme *interface*

De façon générale, l'interface se définit comme un ensemble de dispositifs permettant la communication bidirectionnelle entre l'humain et un système d'une manière générale. Un système englobe par exemple une automobile, une radio, un pupitre, un tableau de bord ou une application informatique. De manière plus détaillée, Ravden et Johnson (1989) définissent l'interface utilisateur de la façon suivante :

“The user interface generally consists of information displayed to the user and facilities which allow the user to enter information into the computer, to manipulate information which is displayed, and to take control actions. It enables the end-user to access and make use of the facilities and functions which the system provides, and to carry out the tasks for which it has been designed. It provides the user with information about the system, about what it does, and about what the user can and should do. It enables the user to learn about the system and to build an understanding of how it works”.

Dans le domaine de l'informatique, l'interface est le point central dans la communication entre l'humain et l'application informatique. En plus de donner accès aux fonctionnalités

d'une application informatique, elle permet à l'utilisateur d'entrer et de manipuler de l'information, de prendre des actions et d'interpréter les informations fournies par l'application.

1.2 Évaluation d'interfaces homme machine

La facilité d'utilisation des logiciels et applications pose des défis intéressants à relever. Le besoin accru des applications utiles et utilisables, justifie l'apparition d'une nouvelle dimension dans la production des applications. Cette dimension est basée sur l'évaluation ergonomique qui est le plus souvent jugée coûteuse en termes de temps et de ressources.

Selon (Senach, 1990), l'évaluation est réalisée pour satisfaire quatre buts :

- 1- pour établir un diagnostic d'usage du système existant,
- 2- pour assurer la qualité de la conception d'interfaces homme machine,
- 3- pour comparer les avantages et les inconvénients de versions de logiciels,
- 4- pour contrôler a priori la qualité ergonomique d'un produit.

Chaque contexte présente des contraintes spécifiques nécessitant la mise en jeu de techniques bien adaptées (Senach, 1990). La section suivante propose une revue des pratiques actuelles d'évaluation d'interfaces homme machine.

Les nouvelles technologies (interfaces graphiques, animation, langages évolués, outils de design, etc.) apportent des nouveaux outils et moyens qui facilitent le développement et la production des applications et améliorent sensiblement l'homogénéité des interfaces homme machine. Ces outils ne garantissent cependant pas encore à l'utilisateur final l'aisance et la facilité d'utilisation et d'apprentissage. Pour faire face à cette problématique, et dans le but d'améliorer les performances de l'interaction homme machine, une évaluation ergonomique est estimée nécessaire afin de remédier aux problèmes sus évoqués et découvrir les autres problèmes qui pourraient empêcher les utilisateurs d'accomplir leurs tâches.

Pour que cette évaluation soit réalisée suivant les contraintes ergonomiques, on doit être en mesure de répondre à la question que mesure-t-on?

Une interface homme machine est évaluée en fonction de son utilité et de son utilisabilité (Senach, 1990). Ces deux concepts sont définis dans la section suivante.

1.2.1 Utilité et utilisabilité

L'utilité et l'utilisabilité sont deux concepts intégrés dans une problématique plus large relative à l'acceptabilité des systèmes en général, qui est principalement la question de savoir si le système est assez bon pour satisfaire tous les besoins et les exigences des utilisateurs et d'autres parties prenantes comme les gestionnaires et les clients. L'acceptabilité des systèmes informatiques peut être scindée selon deux dimensions : l'acceptabilité sociale, s'intéressant au contexte d'utilisation du système, et l'acceptabilité pratique, considérant l'intention que le système permet d'atteindre (Nielsen, 1993). L'arbre de l'acceptabilité des systèmes selon (Nielsen, 1993) est présenté dans la figure suivante.

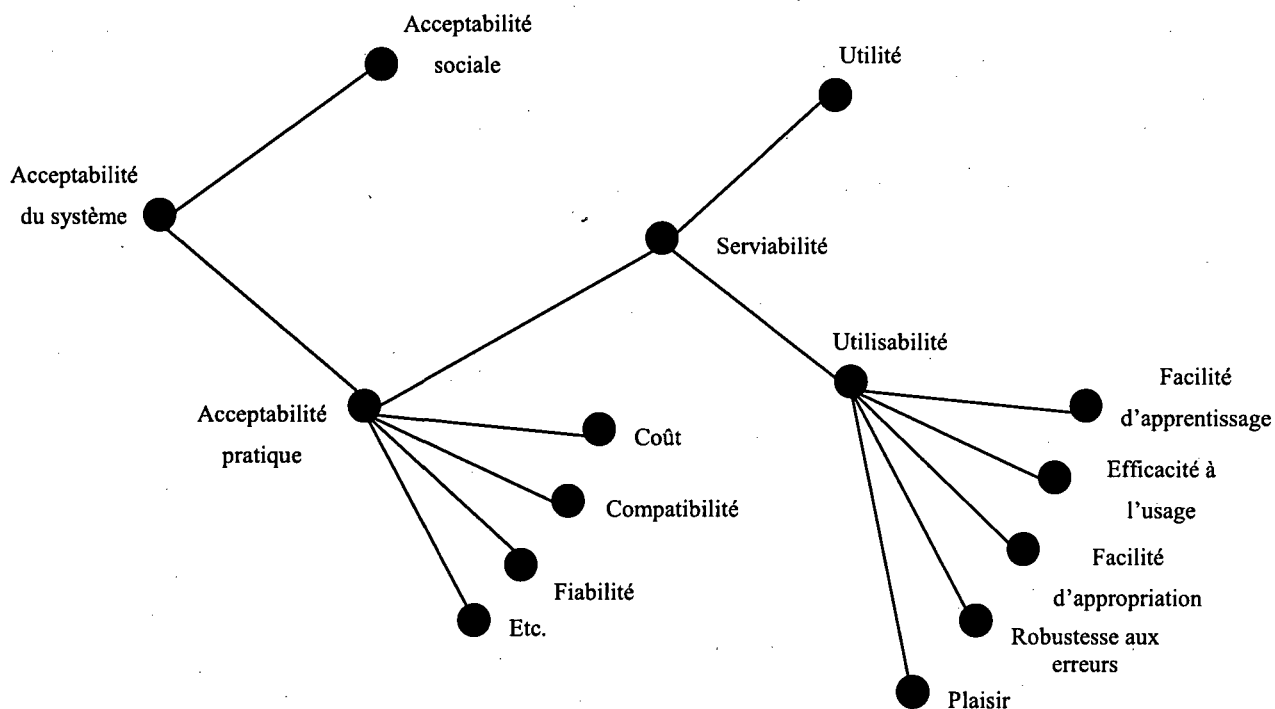


Figure 1 – Modèle d'acceptabilité des systèmes selon (Nielsen, 1993)

Selon le modèle d'acceptabilité des systèmes de Nielsen, l'utilité et l'utilisabilité sont deux attributs qui découlent de l'acceptabilité pratique des systèmes. L'utilité détermine si l'interface permet à l'utilisateur d'atteindre ses objectifs de travail (Senach, 1990). Elle correspond à la capacité fonctionnelle de l'interface, les performances du système et à la qualité d'assistance proposée à l'utilisateur. L'utilisabilité est définie par le degré selon lequel l'interface peut être utilisée, par des utilisateurs identifiés, pour atteindre des objectifs spécifiés. Plusieurs définitions ont été attribuées à l'utilisabilité au fur et à mesure de l'avancement du domaine d'interfaces homme machine. La norme ISO 9241-11 (ISO., 1998) donne les lignes directrices concernant l'utilisabilité. D'après cette norme, l'utilisabilité est définie comme « la mesure dans laquelle un produit peut être utilisé par des utilisateurs spécifiques pour accomplir des buts spécifiques avec efficacité, efficience et satisfaction dans un contexte particulier ». Le standard ISO 9241 définit trois composantes de qualité d'utilisation applicables au design d'interfaces homme machine : l'efficacité, l'efficience et la satisfaction. Ces trois composantes sont mesurées en prenant en compte non seulement le produit, mais aussi l'utilisateur, la tâche et l'environnement ou le contexte d'utilisation. Selon (Nielsen, 1993), l'utilisabilité n'est pas une propriété unique et unidimensionnelle, mais l'utilisabilité comporte plusieurs composantes et est traditionnellement associée avec les cinq attributs suivants :

- L'apprentissage : l'interface doit être facile à apprendre, ceci permet à l'utilisateur de commencer rapidement à travailler avec l'interface.
- L'efficacité : le niveau élevé de productivité que l'utilisateur peut atteindre après avoir appris l'interface.
- La mémorisation : l'utilisateur est capable de réutiliser l'interface après une période de non utilisation sans avoir à apprendre tout de nouveau.
- Les erreurs : le taux d'erreurs dans l'interface doit être minime, ce qui permet à l'utilisateur de commettre moins d'erreurs durant l'utilisation de l'interface, et si l'utilisateur fait des erreurs, il peut facilement les récupérer et les corriger.

- La satisfaction : l'interface doit être plaisante et agréable à utiliser. Ainsi, l'utilisateur apprécie l'interaction avec l'interface.

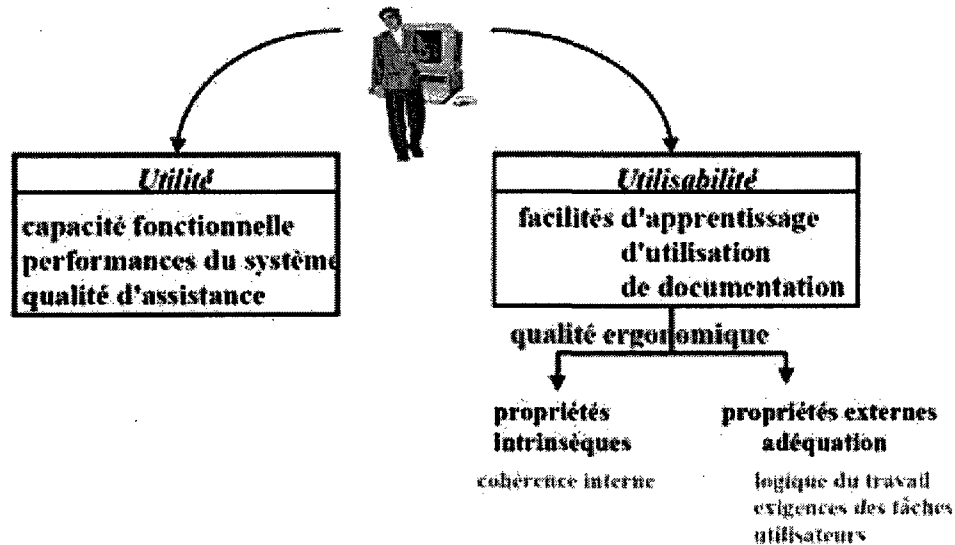


Figure 2 - Dimension de l'évaluation de l'IHM selon Bernard Senach

1.3 Approches d'évaluation d'interfaces homme machine

L'évaluation des IHM est réalisée par l'implication de méthodes et techniques rigoureuses. Ces techniques sont divisées en deux fameuses classes : avec utilisateur et sans utilisateur. La classification la plus fréquente est fondée sur la distinction entre des approches dites prédictives, et des approches dites expérimentales alors que Senach établit une distinction entre approches analytiques et approches empiriques (Nielsen et Molich, 1990; Senach, 1990).

1.3.1 Approches empiriques

Les méthodes classées au sein de cette approche visent l'évaluation d'interfaces réelles, déjà implémentée au moins partiellement. Les approches empiriques ou expérimentales sont basées sur les tests avec les utilisateurs. Ces approches se subdivisent en trois grandes familles :

- Évaluation expérimentale qui consiste à observer l'utilisateur pendant son utilisation de l'interface.
- Évaluation subjective qui consiste à recueillir l'opinion de l'utilisateur en post-utilisation de l'interface, par des interviews libres ou orientés ou des questionnaires.
- Évaluation objective qui vise à analyser les résultats recueillis par l'interaction des utilisateurs et les notes des observateurs.

Les résultats des expérimentations sont recueillis sur un support audiovisuel, capture automatique d'actions (monitorage) lors de l'interaction des utilisateurs avec le système à évaluer, ou encore par des interviews et questionnaires. Ces données permettent d'inférer (selon la précision et la validité des mesures effectuées) les difficultés que rencontrent les utilisateurs et de développer des solutions les réduisant (Senach, 1990). Les données observées font l'objet d'analyse approfondie et d'interprétation. Cette analyse conduit à une synthèse qui indique entre autres le degré de difficulté et les problèmes décelés, et propose éventuellement des recommandations.

Ces approches peuvent être appliquées tout le long du cycle de vie des applications.

1.3.2 Approches analytiques

Les méthodes issues de cette approche visent l'évaluation d'interfaces représentées ou modélisées. L'implémentation, même partielle, de l'interface n'est pas une pré-condition dans le cadre de l'évaluation. Les approches analytiques (prédictives) ne nécessitent pas la présence de l'utilisateur. De fait, elles conviennent dès les premières étapes du cycle de vie d'une application. Ces approches permettent à partir d'une description détaillée des tâches et une analyse approfondie du système, de construire un modèle de l'utilisateur et d'identifier les problèmes potentiels d'utilisabilité. Les méthodes émanant de cette approche sont formelles ou informelles. Plusieurs méthodes et techniques analytiques existent actuellement.

- Méthodes heuristiques : Ces méthodes font appel à des experts qui parcourent les interfaces afin de détecter les points critiques en les confrontant à des critères ergonomiques, aux normes d'interfaces et aux principes de conception.
- Méthode de Cognitive walkthrough : Cette méthode est généralement effectuée par les évaluateurs expérimentés ou des experts. La manipulation de l'interface à évaluer est effectuée au travers des scénarios où les experts interagissent avec le système comme un utilisateur. En faisant un parcours systématique de l'interface, les problèmes éventuels sont alors identifiés. Cette procédure peut être mise en place dès les phases préliminaires de conception de l'interface.
- Modèles prédictifs : Ces modèles servent à prédire les performances d'utilisation à partir des spécifications de conception (parmi ces modèles, on peut citer : GOMS, loi de Fitts, KLM, etc.).
- Modèles de qualité d'interface: Ces modèles cherchent à identifier des propriétés mesurables caractérisant les exigences que doit satisfaire l'interface utilisable (cohérence, lisibilité, etc.) et comme toutes les approches analytiques, ils doivent valider leurs conclusions (la pertinence des propriétés identifiées) par des mesures de performances des utilisateurs (Senach, 1990).

1.4 Choix des méthodes d'évaluation d'interfaces

Le choix de la méthode d'évaluation est une étape importante dans le processus d'évaluation d'interfaces. Plusieurs paramètres doivent être réunis afin de choisir la bonne méthode d'évaluation d'IHM. Ces paramètres comprennent une bonne compréhension de types d'utilisateurs, de tâches, d'applications et d'environnements (Nielsen et Molich, 1990).

Plusieurs composantes humaines sont impliquées lors de l'interaction homme machine. Les composantes les plus fréquemment utilisées sont résumées dans les points suivants :

- 1- Composantes perceptuelles. Ces composantes permettent de percevoir des stimuli (visuels ou auditifs) provenant de l'interface ou même de l'environnement.
- 2- Composante cognitive. Cette composante permet d'identifier et de reconnaître les stimuli perçus par les composantes perceptuelles. Ceci peut se traduire par la récupération de l'information pertinente sur les stimuli perçus auprès de la mémoire humaine.
- 3- Composante motrice. Cette composante permet de répondre aux stimuli en activant selon le contexte et la nature de la tâche la partie motrice correspondante. En effet, si la réponse aux stimuli est sous forme verbale, alors c'est la composante motrice de type parole qui sera déclenchée (activée). Par contre, si la réponse aux stimuli est sous forme manuelle, alors c'est la composante motrice de type moteur qui sera déclenchée (activée).

Étant donné que l'ACD est conçu pour assister les personnes atteintes de troubles cognitifs à réaliser leurs tâches de la vie quotidienne, et vu les différentes composantes impliquées lors de l'interaction de l'utilisateur avec l'interface de l'ACD, le choix des méthodes d'évaluation doit prendre en considération tous ces paramètres.

La composante physique de l'interaction homme machine permet d'avoir une première estimation du temps pris par l'utilisateur pour interagir avec le système. Paul Fitts (1954) établit une relation de l'interaction physique entre la cible à atteindre et la distance à cette cible. Par analogie aux interfaces homme machine, la loi de Fitts peut être facilement appliquée et adaptée en faisant l'analogie entre la cible d'une manière générale et une IHM affichée sur un écran traditionnel ou sur un autre dispositif d'affichage. Dans notre projet, la loi de Fitts décrit d'une manière claire la composante physique ou motrice lors de l'interaction homme machine. Dès lors, la loi de Fitts est la première méthode choisie pour évaluer l'interface de l'ACD dans notre projet de recherche, ce qui nous permet d'estimer le temps requis par la composante motrice de l'utilisateur lors de l'interaction homme machine.

Actuellement, l'interaction homme machine s'oriente de plus en plus vers l'étude non seulement de l'interface ou le système d'une manière générale, mais l'étude de l'utilisateur en soi, en termes de comportement, d'apprentissage, de mémorisation, de raisonnement et même de capacités intellectuelles. En effet, la connaissance du type d'utilisateur et de ses capacités intellectuelles permet de mettre en évidence le design approprié d'interfaces. Par contre, l'étude des caractéristiques de l'utilisateur telles que définies en haut ne peut se faire qu'avec l'implication des spécialistes dans différents domaines tels que, le domaine de la psychologie et le domaine des sciences cognitives.

Selon J. Rivière (2007), les capacités cognitives apparaissent trop précocement pour résulter de l'action motrice exercée sur l'environnement. Ce qui revient à dire que, toute action physique que l'utilisateur accomplit durant l'interaction homme machine suppose qu'un processus cognitif s'est engagé afin d'exercer cette action. Ainsi, pouvoir modéliser et simuler les processus cognitifs, nécessite l'utilisation de méthodes fondées sur des théories cognitives et psychologiques. Pour répondre à cette problématique, les chercheurs dans le domaine de la psychologie et des sciences cognitives en collaboration avec des chercheurs dans le domaine de l'informatique ont développé des méthodes et des techniques dites cognitives. Ces méthodes permettent de prendre en considération, non seulement la composante physique, mais aussi les composantes perceptuelle et cognitive impliquées lors de l'interaction homme machine. Parmi ces méthodes : l'architecture cognitive ACT-R (Anderson et al., 2004) et le modèle GOMS (Kieras, 2003).

1.4.1 Architectures cognitives

De manière générale, une architecture cognitive est un algorithme qui simule une théorie de la cognition humaine. Selon Grant (1996), une architecture cognitive est un ensemble particulier de structures conceptuelles, d'outils, de techniques et méthodes qui peuvent supporter le design et la construction de modèles de la cognition. Anderson (1993) définit les architectures cognitives comme des propositions relativement complètes à propos de la structure de la cognition humaine.

Plusieurs architectures cognitives ont été développées dans les trois dernières décennies. Parmi ces architectures on peut citer les plus utilisées dans la littérature : ACT-R (Anderson, 1993), SOAR (Newell, 1990), EPIC (Kieras et Meyer, 1997), CCT (Bovair et al., 1990; Kieras et Polson, 1999), LICAI (Kitajima et Polson, 1997), CAPS (Just et al., 1996).

Chaque architecture cognitive utilise son propre système de représentation de l'information, déclarative et/ou procédurale, et est appliquée dans un contexte d'utilisation spécifié. Certaines architectures cognitives sont spécifiées avec beaucoup de détails, alors que d'autres non. ACT-R est un exemple d'architecture cognitive très détaillée; elle possède même une implémentation informatique (interface visuelle).

Les architectures cognitives par définition ne sont pas conçues pour simuler ou évaluer des interfaces homme machine. Leur but fondamental est de reproduire avec fidélité le comportement cognitif de l'humain. Ce n'est qu'après l'intégration de la composante « utilisateur » dans le domaine de l'interaction homme machine, que les spécialistes des interactions homme machine ont eu recours aux architectures cognitives. A cet effet, beaucoup d'améliorations ont été apportées aux architectures cognitives pour les adapter au domaine de l'interaction homme machine. Au début, le recours aux architectures cognitives se limitait aux fonctions cognitives telles que la charge cognitive, la fatigue, l'apprentissage, la mémorisation, etc. Mais avec les améliorations récentes comme l'intégration des composantes perceptuelles et motrices dans quelques architectures cognitives (Byrne, 2001), l'utilisation des architectures cognitives s'est élargie jusqu'aux composantes perceptuelles et motrices de l'interaction homme machine.

ACT-R (Adaptative Control of Thought Rational) est une théorie unifiée de la cognition humaine. ACT-R est une architecture cognitive hybride, non dans le sens qu'elle comporte deux sous-systèmes, symbolique et sub-symbolique, pour son fonctionnement général, mais plutôt dans le sens de conception et fondement de cette architecture. En effet, le système symbolique d'ACT-R est inspiré de celui sur lequel se base l'architecture cognitive SOAR comme les chunks et les règles de production, donc tout ce qui est sémantique et procédural (Anderson et al., 2004). De même pour la partie perceptuelle-motrice qui est inspirée de

l'architecture cognitive EPIC (Anderson et al., 2004). Enfin, la composante physique d'ACT-R est basée sur la loi de Fitts (Bothell, 2004). Par conséquent, l'architecture cognitive ACT-R nous semble la plus exhaustive et la plus adaptée au domaine de l'interaction homme machine par l'intégration de toutes les composantes citées en haut.

Durant notre projet de recherche, nous avons utilisé les deux méthodes cognitives ACT-R et GOMS afin de pouvoir d'une part, simuler les composantes cognitives impliquées lors de l'interaction homme machine, dans chacune des méthodes, et d'autre part, comparer la performance de chaque méthode dans le domaine d'évaluation d'interfaces développées dans le contexte d'habitat intelligent. Ces deux méthodes viendront compléter l'analyse réalisée avec la méthode de Fitts.

Chapitre 2

Évaluation des interfaces de l'ACD utilisant ACT-R et la loi de Fitts

Pour répondre aux deux premiers objectifs cités en haut relatifs à la simulation et l'évaluation de l'interface de l'ACD, ce chapitre introduit les deux modèles ACT-R et la loi de Fitts utilisés dans notre projet pour mettre en œuvre ces deux concepts.

2.1 Avant-propos

L'article présenté ci-dessous, est un article accepté dans le journal IJHIS (International Journal of Hybrid Intelligent Systems) et intitulé : "Simulation Model of A Human Machine Interaction Using Analytical Methods", écrit par Belkacem Chikhaoui et Hélène Pigot. Cet article constitue une version étendue de l'article de conférence internationale à comité de lecture, écrit par Belkacem Chikhaoui et Hélène Pigot et publié dans les proceedings du NASTEC 2008 (1st International North American Simulation Technology Conference). Cet article intitulé : "Simulation of a Human Machine Interaction : Locate Objects Using a Contextual Assistant". Dans cet article, l'évaluation des interfaces de l'ACD est effectuée en utilisant les deux modèles analytiques présentés ci-dessus, il s'agit d'ACT-R et la loi de Fitts.

2.2 Résumé de l'article

Le développement standard des interfaces homme machine nécessite le respect des normes ergonomiques et des approches rigoureuses afin de répondre parfaitement aux besoins des utilisateurs finaux. Ceci constitue une préoccupation majeure pour les développeurs,

concepteurs et designers des interfaces homme machine. Le besoin accru de concevoir des interfaces accessibles et facilement utilisables, a poussé les chercheurs dans ce domaine à créer des méthodes et des techniques qui peuvent être appliquées tout au long du processus de développement des interfaces, et qui permettent l'évaluation de ces interfaces en termes d'utilité et d'utilisabilité. Cet article présente une étude sur la simulation des interactions homme machine avec l'ACD développé au laboratoire DOMUS pour assister les personnes présentant des déficits cognitifs lors de la réalisation des activités de la vie quotidienne, et plus particulièrement les activités complexes de cuisine. Pour ce faire, nous avons utilisé l'architecture cognitive ACT-R tout en mettant l'accent sur le temps d'exécution des tâches impliquées lors de la réalisation des deux premières étapes de la recette choisie «préparation des spaghetti ». Afin de valider et supporter les résultats obtenus, nous avons utilisé le modèle de la loi de Fitts.

Les résultats obtenus sont consistants et compatibles avec ceux obtenus par le modèle de la loi de Fitts.

SIMULATION MODEL OF A HUMAN MACHINE INTERACTION USING ANALYTICAL METHODS

Belkacem Chikhaoui

Hélène Pigot

Domus Laboratory, Computer Science Department, Faculty of Science,

University of Sherbrooke, Sherbrooke (QC), J1K 2R1 Canada

email: {belkacem.chikhaoui, helene.pigot}@usherbrooke.ca

KEYWORDS

Simulation, User modeling, Human machine interaction, Cognitive modeling, Evaluation

ABSTRACT

The standard development of human machine interfaces needs the respect of ergonomic norms and rigorous approaches, which constitutes a major concern for computer system designers. The increased need on easily accessible and usable interfaces leads researchers in this domain to create methods and models that make it possible to evaluate these interfaces in terms of utility and usability. This paper presents a study about the simulation of a human machine interaction with an interface of a contextual assistant, using the cognitive architecture ACT-R emphasizing on the time execution of tasks. The results of our model were consistent with those obtained by the Fitts Law model which is a powerful analytical method for evaluating human machine interfaces, developed in this study mainly to support our results.

INTRODUCTION

The evaluation of Human Machine Interfaces (HMI) is becoming increasingly important and constitutes an integral part in the development cycle of computer systems. While the development of interfaces presents some challenges, their evaluation needs rigorous methods to ensure they fulfill the initial specifications and the quality of accessibility, usability and usefulness (Nielsen and Phillips, 1993; Eugenio et al., 2003). Two main approaches for evaluation are currently used, empirical approaches and analytical approaches. Empirical approaches are essentially based on performances or opinions of users gathered in laboratories or other experimental situations. These approaches are user-focused. Unlike the empirical approaches, analytical approaches are not based directly on the user performance, but rather, on the automated examination of interfaces using well-defined structures and rigorous analysis techniques (Yen et al., 2005).

The HMI should be resumed by the actions of pushing buttons displayed on a screen. According to this ap-

proach the Fitts law estimates the time needed to reach the targets displayed on the interface. Nevertheless, the HMI implies three human components, which must be taken in account. The first component is perceptual. In our case the human perceives the signal in a visual manner. The second one is cognitive. Here the human retrieves in his memory the object required and reasons to satisfy specific goals. The third one is motor and necessitates pressing on the selected button.

In this study, we aim to evaluate the interaction with an interface of a contextual assistant developed for cognitively impaired people. The aim of this application is to assist people while preparing meals in their kitchen by using cognitive assistance (Pigot et al., 2005). Due to the related population and the kind of errors they commit we need to take in account the cognitive part involved in the HMI. We then use a powerful analytical method based on cognitive models, emphasizing the cognitive analysis of the tasks and the time execution. We choose to base our analytical method on the cognitive architecture ACT-R (Anderson et al., 2004). Thanks to ACT-R the interaction is decomposed in rules simulating the cognitive behavior of a human using the contextual assistant. We first present an overview of the cognitive architecture ACT-R and of the contextual assistant. Once the task simulated is defined, the model, we developed, is introduced and the results of the simulation are compared to the time estimated by the Fitts law to interact with the contextual assistant.

BACKGROUND

In this section we present an overview of the cognitive architecture ACT-R, and then we introduce the contextual assistant application and the interface to be modeled.

Cognitive architecture ACT-R

The cognitive architecture ACT-R is built to simulate and understand human cognition Anderson et al. (2004, 2005). It consists of a set of modules such as the visual, aural, motor, intentional and declarative module that are integrated through a central production system.

ACT-R is a hybrid architecture that combines two subsystems: symbolic system including semantic and procedural knowledge, and subsymbolic system evaluating knowledge activations.

Each knowledge in the ACT-R's declarative memory called chunk Newell (1990), which is associated with level of activation computed by the subsymbolic system. The activation level reflects the degree of availability of the chunk at any particular time. The subsymbolic system assigns also utility values to rules (procedural knowledge) to determine the predominant knowledge available at a specific time. Therefore, the predominant knowledge defined as the rule with the highest utility. In ACT-R the perceptual and motor modules are used to simulate interfaces between the cognitive modules and the real world (Byrne, 2001; Bothell, 2004).

Visual and Motor Modules of ACT-R

The visual module that is part of the perceptual modules, has two subsystems, the positional system (where) and the identification system (what) that work together in order to send the specified chunk to the visual module. The positional system is used to find objects. When a new object is detected, the chunk representing the location of that object is placed in the visual-location buffer according to some constraints provided by the production rule (Bothell, 2004). The identification system is used to attend to locations which have been found by the positional system. The chunk representing a visual location will cause the identification system to shift visual attention to that location. The result of an attention operation is a chunk, which will be placed in the visual buffer (Byrne, 2001; Bothell, 2004). The motor module contains only one buffer through which it accepts requests (Bothell, 2004). Two actions are available in ACT-R, to click with the mouse or press a key on the keyboard.

Contextual Assistant

The Contextual assistant application is developed to assist persons with cognitive disabilities (Pigot et al., 2007a; Lussier-Desrochers et al., 2007). The aim is to foster autonomy in the daily living tasks and particularly during complex cooking tasks such as preparing pancakes, or spaghetti (Pigot et al., 2007b). The cooking task is decomposed of steps displayed on a touch screen. The two first steps consist of gathering the utensils and ingredients necessary to the recipe (Figure 1). The other steps explicit the recipe using photo and video on the screen as well as information dispatched all around the kitchen. The contextual assistant is specifically designed to help people remembering the places where the objects are stored. To do so, the contextual assistant contains an interface called the locate application displaying the objects to search. When an object

is pushed in the main interface, the contextual assistant looks for the location of that object in the environment using techniques of pervasive computing and indicates the location by highlighting the appropriate locker containing that object as shown in Figure 2. In this study we simulate the first two steps of the spaghetti recipe. They consist of first knowing the list of objects to gather, either utensils or ingredients, and then to use the locate application in order to find each object.

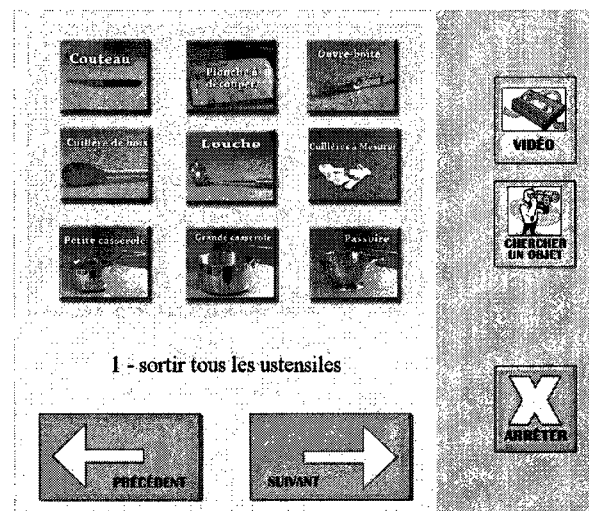


Figure 1: Main interface of the contextual assistant

The contextual assistant interface is displayed on a 1725L 17" LCD Touchscreen, with 13.3" (338 mm) horizontal and 10.6" (270 mm) vertical useful screen area. It is configured to 1024 x 768 optimal native resolution running Macintosh. The screen is fixed under a closet nearby the oven in order to be easily accessible and also protected against the cooking splashes.

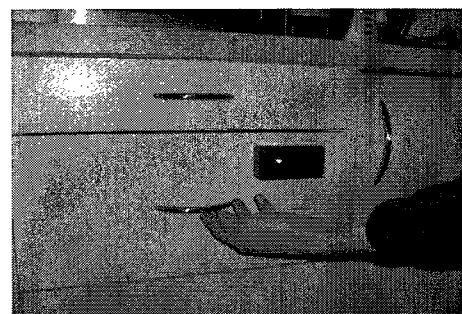


Figure 2: Locker state when an object is pushed

MODELING THE INTERACTION WITH THE CONTEXTUAL ASSISTANT USING ACT-R

In this section, we present the modeling process of the tasks involved in our study, which are gathering utensils and gathering ingredients, emphasizing on the perceptual and cognitive parts, using the perceptual motor modules of ACT-R.

Task analysis: gathering utensils and ingredients

We model the first two steps of the recipe, gathering utensils and gathering ingredients. The interactions with the touch screen are simulated without taking in account the time taken by the subject to pick up the objects in the environment. The two first steps require three subtasks (Figure 3). The first subtask consists of activating the locate application in order to locate each object required by the recipe. This is done by pushing the button “LOOK-FOR-OBJECT” (in French, “CHERCHER-UN-OBJET”), which is displayed on the main interface of the contextual assistant (Figure 1). The second subtask is to locate each object, either utensils or ingredients, needed in the current step by pushing the button corresponding to the object in the locate application. The third task consists of coming back to the main application in order to know the next step of the recipe. The tree decomposition is presented in figure 3, where the translation in English is available to compare the tasks tree from the interface of Figure 1. The nodes in capital indicate the action to click on the named button, while the other nodes represent tasks to be decomposed.

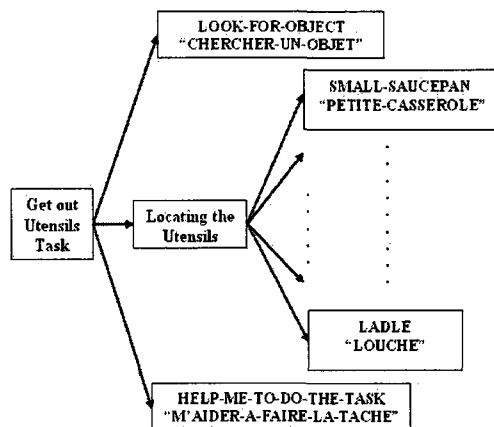


Figure 3: Tree decomposition of the gathering utensils task

Gathering ingredients and utensils model

The model developed aims to simulate the HMI during the two first steps of the recipe. In that task, three different interfaces are involved, the interface of the locate application and the two of the contextual application displaying the utensils and ingredients needed in the recipe. The model uses ACT-R to emphasize the cognitive processes involved when looking for an object and choosing the button to push. It is decomposed of three phases, the visual phase, the recognition phase and the motor phase. The visual phase consists of localizing the object to perceive and then identifying it. We consider that all buttons displayed on the screen are objects, either the button used to locate a utensil or ingredient, or the buttons to navigate in the interface. The first one is the button “LOOK-FOR-OBJECT” as described in Figure 3. Then, all the utensils needed in the recipe are presented in the visual interface of ACT-R. Finally, to complete the first step of the recipe, the button “HELP-ME-TO-DO-THE-TASK” is presented in order to come back to the main interface of the contextual assistant and pursue the second step of the recipe. Each object of the interface is displayed at defined coordinates (x, y) on the screen. These coordinates specify the request made to the visual-location buffer of ACT-R, which creates a chunk representing the location of the specified object. After that, the identification system identifies the name of the object and creates a chunk placed in the visual buffer. The location and identification phases last 185 ms (Bothell, 2004; Byrne, 2001). The objects are presented to the visual module of ACT-R by the mean of a list of all the objects (buttons of the interface) to be pushed on. Figure 4 shows some ACT-R productions responsible of the visual encoding phase.

The recognition phase begins when the chunk of the object is placed in the visual module. This phase implies to recover that specific chunk from the declarative memory. The result of this phase is a chunk that represents the object with some characteristics as color, localization on the screen, name, and kind of object. The motor phase consists of activating the motor actions via a request to the motor buffer in order to click on the object. The three phases process is applied for each object displayed in the interface for the two steps of the recipe. The gathering utensils and ingredients model finishes when the last object of the ingredient list is reached.

The ACT-R model is developed using the ACT-R 6 environment. No noise is introduced in the perceptual motor modules. no retrieval error is modeled in the recognition phase. These restrictions lead to a deterministic model. Figure 5 shows an example of execution traces of the ACT-R model for the visual encoding and the shift attention actions respectively. The visual-location request takes place at time 0.050 seconds and the request to move-attention is made at time 0.100 seconds. The encoding needs still 0.085 seconds to be completed

```

(P start-application
 =goal>
  ISA      begin
    ; Initializing the model
=>>
  -imaginal>
  +visual-location>
  ; Making request of the visual-location buffer
  ISA      visual-location
    :attended nil
  +goal>
  ISA      get-object
  state    find-location
)
(P attend-utensil
 =goal>
  ISA      get-object
  state    find-location
  ; Move attention to the location
  ; screen-x 122 and screen-y 250
 =visual-location>
  ISA      visual-location
  screen-x 122
  screen-y 250
 ?visual>
  state    free
=>>
  +visual>
  ISA      move-attention
  screen-pos =visual-location
 =goal>
  state    attend
)

```

Figure 4: Example of some ACT-R productions responsible for the visual encoding phase

and store the chunk into the visual buffer.

Results of the ACT-R model

Figures 6 and 7 show the progress of performing the two tasks: get out utensils and get out ingredients respectively. The first task (get out utensils) lasted 6510 ms and the second task (get out ingredients) lasted 8101 ms. The overall time to complete the whole task equals to the sum of the two previous times: $7107 + 8101 = 15208$ ms. The time taken to gather the utensils and ingredients follows a linear model depending on the number of objects to search. No differences are observed between the various location of the objects on the screen.

MODELING THE INTERACTION WITH THE CONTEXTUAL ASSISTANT USING FITTS LAW

In order to support and validate our results, we used the Fitts Law model, widely used in the evaluation of human machine interfaces. In the Fitts Law, the movement time is proportional to the target amplitude and inversely proportional to the target width.

Fitts Law model

In human machine interfaces, the formulation of Fitts Law (Fitts, 1954) states that the movement time (MT) is

```

0.000 PROCEDURAL PRODUCTION-SELECTED
START-APPLICATION
0.000 PROCEDURAL BUFFER-READ-ACTION GOAL
0.050 PROCEDURAL PRODUCTION-FIRED
START-APPLICATION
THE SUBJECT STARTS TO LOOK FOR NEW OBJECT
0.050 PROCEDURAL MODULE-REQUEST
VISUAL-LOCATION
0.050 PROCEDURAL MODULE-REQUEST GOAL
0.050 PROCEDURAL CLEAR-BUFFER IMAGINAL
0.050 PROCEDURAL CLEAR-BUFFER
VISUAL-LOCATION
0.050 PROCEDURAL CLEAR-BUFFER GOAL
0.050 VISION Find-location
0.050 VISION SET-BUFFER-CHUNK
VISUAL-LOCATION LOC1
0.050 GOAL CREATE-NEW-BUFFER-CHUNK
GOAL ISA GET-OBJECT
0.050 GOAL SET-BUFFER-CHUNK GOAL
GET-OBJECT0
0.050 PROCEDURAL CONFLICT-RESOLUTION
0.050 PROCEDURAL PRODUCTION-SELECTED
ATTEND-UTENSIL
0.050 PROCEDURAL BUFFER-READ-ACTION GOAL
0.050 PROCEDURAL BUFFER-READ-ACTION
VISUAL-LOCATION
0.050 PROCEDURAL QUERY-BUFFER-ACTION VISUAL
0.100 PROCEDURAL PRODUCTION-FIRED ATTEND-UTENSIL
SHIFT ATTENTION TO A SPECIFIED LOCATION
ON THE SCREEN
0.100 PROCEDURAL MOD-BUFFER-CHUNK GOAL
0.100 PROCEDURAL MODULE-REQUEST VISUAL
0.100 PROCEDURAL CLEAR-BUFFER VISUAL-LOCATION
0.100 PROCEDURAL CLEAR-BUFFER VISUAL
0.100 VISION Move-attention LOC1-0 NIL
0.100 PROCEDURAL CONFLICT-RESOLUTION
0.185 VISION Encoding-complete LOC1-0 NIL
0.185 VISION SET-BUFFER-CHUNK VISUAL TEXT1

```

Figure 5: Example of execution trace of the ACT-R model for the visual encoding action

function of target amplitude (A) and target width (W). Our model is based on the Mackenzie's [1995] version of Fitts Law in which the movement time (MT) follows the equation:

$$MT = a + b * \log_2\left(\frac{A}{W} + 1\right) \quad (1)$$

The second term of the equation (1): $\log_2\left(\frac{A}{W} + 1\right)$ is known as the index of difficulty ID, where a and b are constants derived empirically. They can be interpreted by the y-intercept and the slope of a predictive linear regression equation (MacKenzie, 1995) (MacKenzie et al., 1991). In our study, the user-interface interaction is based on the use of a touchscreen, assuming that users remain standing at a distance of 30 cm from the touchscreen, and point directly on the displayed objects by touching them using their index finger. The index finger is held down before starting the interaction, which constitutes the start position. After each pointing action, users returned their index finger to the start position, and the procedure continued like that.

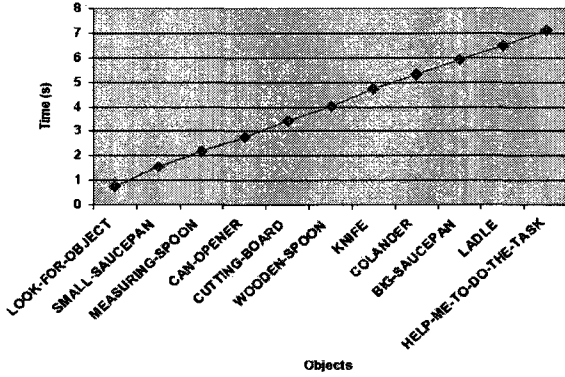


Figure 6: Progress of time depending on progress in the tasks of get out utensils and get out ingredients

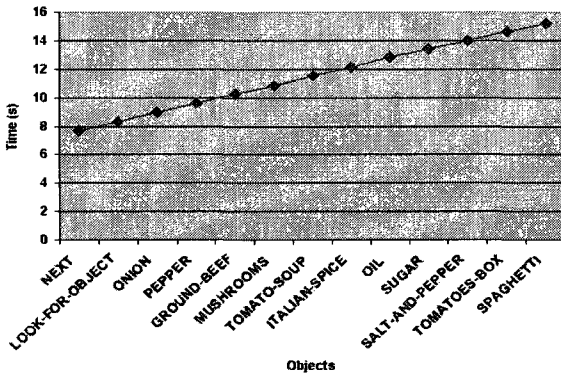


Figure 7: Progress of time depending on progress in the tasks of get out utensils and get out ingredients

Results of the Fitts Law model

Table 1 shows the index of difficulty values obtained when applying the formula $\log_2\left(\frac{A}{W} + 1\right)$ on some objects displayed in the interface, and the corresponding predicted movement time (MT) obtained by applying the equation 1. The target amplitude (A) remains constant while the button width (W) varies as seen in figure 1.

The total time of the whole task applying the Fitts Law is estimated using the following equation:

$$MT_{Total} = \sum_{i=1}^n MT_i \quad (2)$$

Where n represents the number of objects used by the user in the interface, and MT_i the corresponding movement time of each object. The total movement time of the whole task applying the equation 2 is : 14977 ms.

Object-Name	A (cm)	W (cm)	ID (bits)	MT (ms)
BIG-SAUCEPAN	30	5.8	2.625	614.125
NEXT-BUTTON	30	7.6	2.306	553.834
LOOK-FOR-OBJECT	30	3.8	3.152	713.728
MUSHROOMS	30	5.8	2.625	614.125

Table 1: Index of Difficulty values for some Objects in the Interface and the corresponding movement time

COMPARISON OF RESULTS

The results of the predicted time of the task gathering utensils, gathering ingredients and the predicted time of the whole task in both models ACT-R model and Fitts Law model are shown in Table 2.

Tasks	ACT-R	Fitts Law
Predicted time of getting out Utensils Task (ms)	7107	6954
Predicted time of getting out Ingredients Task (ms)	8101	8023
Predicted time of the whole Task (ms)	15208	14977

Table 2: Time estimation of gathering utensils task, gathering ingredients task and the whole task in both models ACT-R and Fitts Law

The ACT-R results are consistent with those obtained by the Fitts Law model as shown in Figures 8 and 9. The times to press each object predicted in both models ACT-R and Fitts Law are very close.

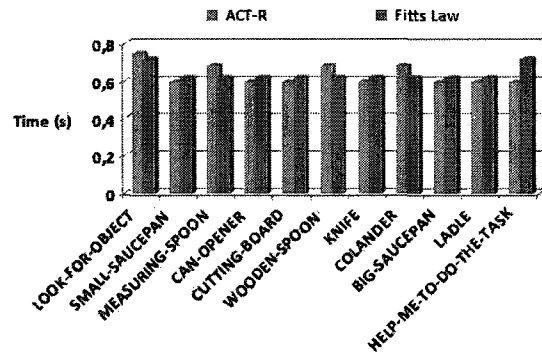


Figure 8: ACT-R and Fitts Law model predictions for the gathering utensils task

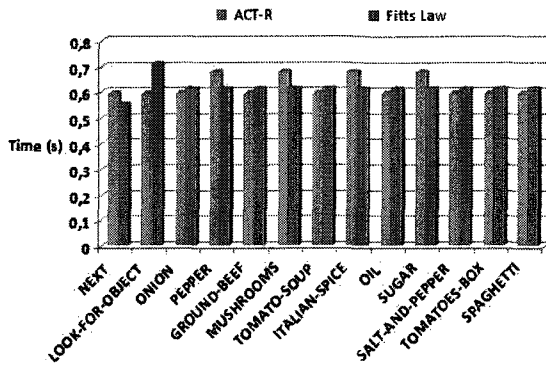


Figure 9: ACT-R and Fitts Law model predictions for the gathering ingredients task

GENERAL DISCUSSION

The ACT-R model we developed is proved robust and efficient in our analysis. In fact, the results obtained by the ACT-R model were consistent with those obtained by the Fitts Law model in terms of the predicted time execution of tasks as mentioned previously; this demonstrates that cognitive models and particularly ACT-R can give good predictions in the evaluation of HMI.

The results of the ACT-R model show that, the size of objects in the interface is not taken into consideration, and our model does not make difference in the predicted time of the pushing “HELP-ME-TO-DO-THE-TASK” button for example, and the pushing “WOODEN-SPOON” object; these two actions have the same predicted time which equals to 597 ms. Unlike the ACT-R model, the Fitts Law model takes in account the object’s size in the interface. The predicted time to push the “HELP-ME-TO-DO-THE-TASK” button using the Fitts Law is 713 ms and the predicted time to push the “WOODEN-SPOON” object is 614 ms. However, some differences are observed as presented in Figures 8 and 9. The simulation of the HMI with the objects “HELP-ME-TO-DO-THE-TASK” (Figure 8) and “LOOK-FOR-OBJECT” (Figure 9) takes more time with the Fitts Law. This is due to the smaller size of these buttons (width = 3.8 cm) compared with the size of the other objects. However, the object “NEXT” (Figure 9) necessitates less time to be pushed, which has the largest size (width = 5.8 cm) in the interface. The simulation of the HMI with the object “LOOK-FOR-OBJECT” as shown in Figure 8 takes more time with the ACT-R model. This is due to the initialization of the model such as the goal buffer, the retrieval buffer and the visual buffers. In the ACT-R model, the focus is essentially on the visual encoding and the recognition of objects and how to interact with the interface using motor actions. This is supported by some scientific literature such as the

use of cognitive models in the evaluation of expert cell phone menu interaction and the evaluation of a contextual assistant’s interface designed in the context of smart homes respectively (Amant et al., 2007; Chikhaoui and Pigot, 2008a).

The results of the ACT-R model are considered suitable and correct comparing them to those obtained by the Fitts Law model. In fact, as shown in Table 2 the estimated time of the whole task in the ACT-R model (15208 ms) is very close to the Fitts Law model time estimated to 14977 ms.

We believe nevertheless, that our study lays out new perspectives of research in this domain particularly how to use perceptual motor modules of the ACT-R architecture to simulate the HMI.

CONCLUSION

The main goal of our study is to evaluate the HMI of a contextual assistant by simulating the interaction with these interfaces, focusing on the time execution of tasks. We used the cognitive architecture ACT-R as a powerful tool to develop our model.

Our ACT-R model consists of two parts, the model of the interface of the contextual assistant which represents the environment to interact with, and the model of the cognitive processes required to interact with the interface. The perceptual part of the cognitive processes constitutes the difficult part in our ACT-R model, due to the scarcity in the documentation about the perceptual module in the literature.

The results of the ACT-R model were compared with those obtained by the Fitts Law model, developed in this study in order to argue and support our results. The results of our model were consistent with the results of the Fitts Law model. Our model gives a good prediction of user performance, which makes it powerful and realistic.

FUTURE IMPROVEMENTS

The model we developed constitutes the first step of the evaluation of HMI using a contextual assistant. Three future improvements will add scientific validity to our model. First, the results of our model were compared with those obtained by the Fitts Law model. The results of the Fitts Law model are not always good and exact, but have a certain percentage of errors. It would be interesting to do some experiments with real persons to collect real data and compare them with our results. Second, our model is deterministic and does not make errors. It should be extended to allow errors in the pointing actions. These errors are essentially related to memory problems that may occur in the task modeling (Serna et al., 2005; Dion and Pigot, 2007) and during the interaction with the contextual assistant’s interface (Chikhaoui and Pigot, 2008b). Finally, the ac-

tion of searching an object is resumed to the HMI with the touch screen. The contextual assistant offers an interaction with the environment to help people recovering utensils and ingredients dispatched in the kitchen. It would be interesting in the future to model this part and simulate the movement of users picking up the objects in the kitchen. Therefore, the extended model should simulate people making a task with contextual assistant and the errors committed by people with cognitive impairments.

REFERENCES

- Amant R.S.; Horton T.E.; and Ritter F.E., 2007. *Model-based Evaluation of Expert Cell Phone Menu Interaction*. *ACM Transactions on Computer-Human Interaction*, 14(1), 1–24.
- Anderson J.R.; Bothell D.; Byrne M.D.; Douglass S.; Lebiere C.; and Qin Y., 2004. *An Integrated Theory of the Mind*. *Psychological Review*, 111, 136–1060.
- Anderson J.R.; Taatgen N.A.; and Byrne M.D., 2005. *Learning to Achieve Perfect Time Sharing: Architectural Implications of Hazeltine, Teague, Ivry (2002)*. *Journal of Experimental Psychology: Human Perception and Performance*, 31, No. 4, 749–761.
- Bothell D., 2004. *ACT-R 6.0 Reference Manual*. Working draft.
- Byrne M.D., 2001. *ACT-R/PM and menu selection: Applying a cognitive architecture to HCI*. *International Journal of Human-Computer Studies*, 55, 41–84.
- Chikhaoui B. and Pigot H., 2008a. *Evaluation of a Contextual Assistant Interface Using Cognitive Models*. In waset (Ed.), *proceedings of the 5th International Conference on Human-Computer Interaction*. vol. 34, 36–43.
- Chikhaoui B. and Pigot H., 2008b. *Simulation of a Human Machine Interaction: Locate Objects Using a Contextual Assistant*. In M. Beldjehem (Ed.), *proceedings of the 1st International North American Simulation Technology Conference*. 75–80.
- Dion A. and Pigot H., 2007. *Modeling cognitive errors in the realization of an activity of the everyday life*. In *Cognitio 2007*.
- Eugenio B.D.; Haller S.; and Glass M., 2003. *Development and Evaluation of NL interfaces in a Small Shop*. *AAAI Spring Symposium on Natural Language Generation in Spoken and Written Dialogue*, 1–8.
- Fitts P.M., 1954. *The information capacity of the human motor system in controlling the amplitude of movement*. *Journal of Experimental Psychology*, 47, NO. 6, 381–391.
- Lussier-Desrochers D.; Lachapelle Y.; Pigot H.; and Bauchet J., 2007. *Apartments for People with Intellectual Disability: Promoting Innovative Community Living Services*. In *2nd International Conference on Intellectual Disabilities/Mental Retardation*.
- MacKenzie I.S., 1995. *Movement Time Prediction in Human Computer Interfaces*. In *Readings in human-computer interaction*. 2nd, 483–493.
- MacKenzie I.S.; Sellen A.; and Buxton W., 1991. *A comparison of input devices in elemental pointing and dragging tasks*. In *Proceedings of the CHI '91 Conference on Human Factors in Computing Systems*. New York: ACM, 161–166.
- Newell A., 1990. *Unified Theories of Cognition*. Harvard University Press, Cambridge (Mass). ISBN 0-674-92099-6.
- Nielsen J. and Phillips V.L., 1993. *Estimating the relative usability of two interfaces: heuristic, formal, and empirical methods compared*. *Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, 214–221.
- Pigot H.; Bauchet J.; and Giroux S., 2007a. *Assistive devices for people with cognitive impairments*, John Wiley and Sons, chap. 12. *The Engineering Handbook on Smart Technology for Aging, Disability and Independence*.
- Pigot H.; Lussier-Desrochers D.; Bauchet J.; Lachapelle Y.; and Giroux S., 2007b. *A smart home to assist recipes completion*. In *Festival of International Conferences on Caregiving, Disability, Aging and Technology (FICCDAT), 2nd International Conference on Technology and Aging (ICTA)*.
- Pigot H.; Savary J.P.; Metzger J.L.; Rochon A.; and Beaulieu M., 2005. *Advanced technology guidelines to fulfill the needs of the cognitively impaired population*. In *3rd International Conference On Smart Homes and health Telematic (ICOST)*. IOS Press, Assistive Technology Research Series, 25–32.
- Serna A.; Pigot H.; and Rialle V., 2005. *Modeling the performances of persons suffering Alzheimer's disease on an activity of the daily living*. In *18th Congress of the International Association of Gerontology*.
- Yen B.; Hu P.; and Wang M., 2005. *Towards Effective Web Site Designs: A Framework for Modeling, Design Evaluation and Enhancement*. *Proceedings of IEEE International Conference on e-Technology, e-Commerce, and e-Service (EEE05)*, 1–6.

AUTHOR BIOGRAPHY

BELKACEM CHIKHAOUI received the Eng. degree in computer science from the University of Boumerdes, Algeria, in 2004. He is currently pursuing the Master degree in computer science at the University of Sherbrooke, Sherbrooke, QC, Canada. His research interests include smart homes, cognitive modeling and evaluation of Human-Machine Interfaces.

HÉLÈNE PIGOT is professor in computer science at Sherbrooke University, Canada. She co-founded the DOMUS laboratory. She is director of the Center on smart environment of the Sherbrooke University. Hélène Pigot received his PhD on speech recognition in 1985, at P.M. Curie University, France. She was then graduated in occupational therapy at Montreal University, Canada. Professor at Montreal University during five years, she worked on spatial orientation deficits in Alzheimer disease. Since 2000, she dedicates her research on smart home, cognitive modeling and cognitive assistance for people with cognitive deficits.

Chapitre 3

Évaluation des interfaces de l'ACD utilisant ACT-R et GOMS

Pour répondre à l'objectif de validation des modèles analytiques développés, ce chapitre introduit l'étude expérimentale menée au laboratoire DOMUS avec des humains, et présente par la suite les deux modèles cognitifs utilisés dans l'évaluation des interfaces de l'ACD. Le modèle d'ACT-R constitue un ajustement du premier modèle décrit dans le chapitre 2. Le deuxième modèle est basé sur le modèle GOMS, qui est fondé sur une représentation hiérarchique des tâches. Enfin, on présente le résumé de l'article publié dans le journal IJIT (International Journal of Information Technology).

3.1 Expérimentation

L'expérimentation consiste à sélectionner des objets dans l'interface de l'ACD. Ces objets correspondent aux ustensiles et ingrédients nécessaires pour la préparation de la recette de cuisine. L'expérimentation est réalisée de telle sorte que les trois principaux points suivants soient respectés :

- 1) Le temps d'exécution des tâches doit être mesuré avec précision.
- 2) L'ordre dont lequel les objets sont affichés dans l'interface de l'ACD n'affecte pas la vitesse de sélection.
- 3) L'expérimentation est uniforme pour tous les participants.

Pour répondre au premier point, on a décidé de séparer le temps de reconnaissance (identification) des objets du temps de sélection.

La reconnaissance des objets peut être présentée expérimentalement en utilisant un dispositif mobile tel que le PDA (Personal Digital Assistant). Pour ce faire, une application est développée et installée sur le PDA, cette application affiche à l'utilisateur le nom de l'objet à sélectionner dans l'interface de l'ACD. Il est à mentionner, que l'ordre dans lequel les objets sont affichés sur le PDA est différent de celui dans lequel les objets sont affichés sur l'interface de l'ACD. Ceci évite l'automatisme et la sélection séquentielle des objets.

Le temps d'exécution des tâches est récupéré dans un fichier journal à la fin de chaque expérimentation.

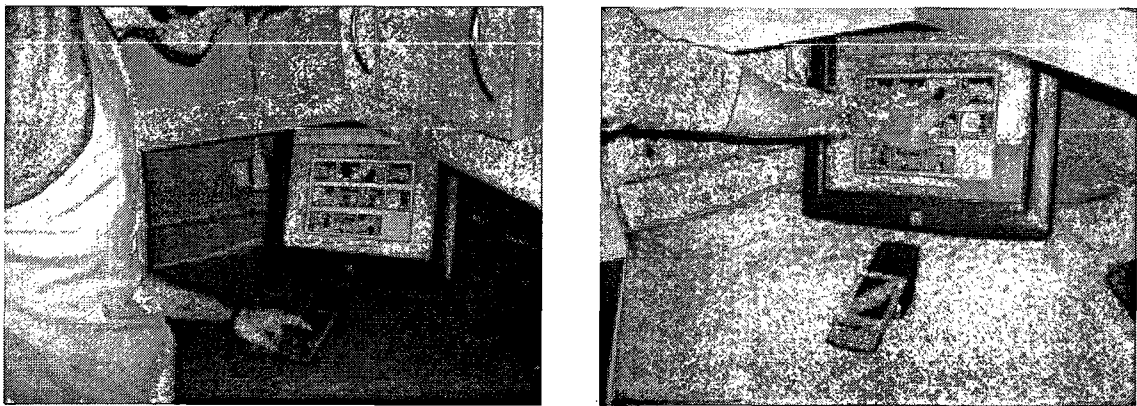


Figure 3 - Exemple de déroulement des expérimentations

3.2 Ajustement du modèle ACT-R

Afin que le modèle d'ACT-R soit conforme avec l'étude expérimentale, un ajustement de ce modèle est nécessaire. Le modèle d'ACT-R présenté dans le chapitre 2 est constitué de trois phases essentielles : visuelle, cognitive et motrice. Dans le nouveau modèle, l'utilisateur accomplit deux phases visuelles (celle sur le PDA pour percevoir l'objet à faire sortir, et celle sur l'interface de l'ACD pour percevoir l'objet à sélectionner), deux phases cognitives

(relatives à la reconnaissance de l'objet sur le PDA et l'interface de l'ACD respectivement) et deux phases motrices (qui correspondent à la sélection de l'objet sur l'interface de l'ACD et le click sur le PDA pour afficher le nom du prochain objet à faire sortir). Dès lors, les trois phases sont comptabilisées deux fois dans le nouveau modèle d'ACT-R.

3.3 Le modèle GOMS

Le modèle GOMS (Goals, Operators, Methods and Selection rules) est l'un des modèles cognitifs les plus puissants dans la prédiction des temps d'exécution des tâches et les performances des utilisateurs. Quatre familles du modèle GOMS existent actuellement.

- 1) CMN-GOMS : Card, Moran & Newell GOMS, c'est la formulation originale proposée par Card, Moran et Newell. Elle définit la façon d'exprimer les buts et les sous-buts en représentation hiérarchique, les découper en opérateurs élémentaires, spécifier les méthodes en séquences et formuler la sélection des méthodes. Pour une tâche particulière, le CMN-GOMS peut prédire la séquence des opérateurs et le temps d'exécution. Toutefois, cette description reste vague.
- 2) KLM : Keystroke Level Model (Kieras, 2001), c'est une version simplifiée de CMN. Ce modèle utilise uniquement les opérateurs de niveau de frappe. Il n'y a pas de notion de buts, de méthodes ou de sélection de règles. Dans ce modèle l'analyste liste les frappes et mouvements de souris que l'utilisateur doit effectuer pour accomplir une tâche, puis il utilise des heuristiques pour placer des opérations mentales qui précèdent l'exécution des tâches.
- 3) CPM-GOMS : Critical Path Method, est un modèle d'analyse de tâche basé directement sur le modèle du processeur humain. Il représente une version parallèle de l'activité, i.e. les opérateurs du processeur humain peuvent être exécutés en parallèle.

- 4) NGOMSL : Natural GOMS Language, est une version plus rigoureuse de CMN-GOMS. NGOMSL présente une procédure pour identifier tous les composants de GOMS sous une forme similaire à celle d'un langage de programmation ordinaire.

Une étude comparative entre les différentes versions de GOMS est présentée dans le tableau 1.

Tableau 1 - Étude comparative entre les différentes versions de GOMS

TYPE DE TÂCHE DESIGNATION	Séquentielle	Parallèle
Consistance	NGOMSL	X
Séquence d'opérateurs	CMN-GOMS NGOMSL	CPM-GOMS
Temps d'exécution	KLM, CMN-GOMS NGOMSL	CPM-GOMS
Temps d'apprentissage	CMN-GOMS NGOMSL	X
Couverture d'erreurs	NGOMSL	X

Dans notre projet, nous avons choisi la version la plus rigoureuse et la plus utilisée dans la littérature, celle de NGOMSL. Avec cette version on peut modéliser les trois phases essentielles dans l'accomplissement des deux tâches définies précédemment. Le modèle GOMS est développé suivant les recommandations de l'étude expérimentale présentées dans la section 3.1.

3.4 Avant-propos de l'article

L'article présenté ci-dessous, est un article publié dans le journal IIJT (International Journal of Information Technology) et intitulé : "Analytical Model Based Evaluation of Human

Machine Interfaces Using Cognitive Modeling'', écrit par Belkacem Chikhaoui et H el ene Pigot. Cet article constitue une version  tendue de l'article de conf erence internationale   comit e de lecture,  crit par Belkacem Chikhaoui et H el ene Pigot et publi e dans les proceedings de ICHCI 2008 (5th International Conference on Human Computer Interaction). Dans cet article l' valuation des interfaces de l'ACD est effectu e en utilisant les deux mod es cognitifs ACT-R et GOMS, et les r sultats cette fois ci ont  t  compar s avec ceux obtenus dans l' tude exp rimentale.

3.5 R sum  de l'article

Les mod es cognitifs font partie des m thodes analytiques d' valuation d'IHM, qui permettent de pr dire quelques aspects d'utilit  et d'utilisabilit  des interfaces, et de simuler les interactions avec ces interfaces. L'action de pr diction est bas e principalement sur l'analyse de t ches que l'utilisateur est cens  faire en termes d'actions et de processus cognitifs pour atteindre ses objectifs. L'analyse de t ches est l' tape la plus primordiale dans le processus d' valuation d'IHM, elle facilite consid rablement la compr hension et les fonctionnalit s du syst me ou l'interface    valuer. Les mod es cognitifs ne font pas n cessairement appel   l'utilisateur pendant le processus d' valuation d'interface. Toutefois, l' valuation des interfaces avec ces mod es doit  tre op r e par des experts ou des personnes dot es d'une expertise tr s  lev e.

Cet article pr sente une  tude approfondie sur l' valuation des interactions homme machine avec l'interface de l'ACD, en utilisant deux m thodes analytiques tr s performantes et largement utilis es dans la litt rature, il s'agit de l'architecture cognitive ACT-R et le mod le GOMS. Le pr sent travail met en  vidence le degr  de pertinence de ces mod es et comment ces mod es peuvent  tre utilis s dans l' valuation, le design et la recherche dans le domaine des interactions homme machine, en se focalisant sur l'analyse et la description des t ches d'un cot , et sur le temps d'ex cution de ces t ches de l'autre cot . Afin que notre  valuation aurait des feedbacks positifs dans la litt rature, et que les r sultats obtenus refl tent le r alisme de l'interaction homme machine, nous avons proc d    une  tude exp rimentale qui

nous permettait d'avoir des données réelles sur l'interaction de l'utilisateur avec l'interface de l'ACD, et de comparer les données simulées avec les données réelles. Les résultats obtenus montrent que les deux modèles donnent des meilleures prédictions sur les performances d'utilisateurs, et plus spécifiquement le modèle d'ACT-R, au niveau des objets ainsi qu'au niveau des tâches. Les résultats obtenus sont très proches de ceux obtenus dans l'étude expérimentale.

Analytical Model Based Evaluation of Human Machine Interfaces Using Cognitive Modeling

Belkacem Chikhaoui, H el ene Pigot

Abstract—Cognitive models allow predicting some aspects of utility and usability of human machine interfaces (HMI), and simulating the interaction with these interfaces. The action of predicting is based on a task analysis, which investigates what a user is required to do in terms of actions and cognitive processes to achieve a task. Task analysis facilitates the understanding of the system’s functionalities. Cognitive models are part of the analytical approaches, that do not associate the users during the development process of the interface. This article presents a study about the evaluation of a human machine interaction with a contextual assistant’s interface using ACT-R and GOMS cognitive models. The present work shows how these techniques may be applied in the evaluation of HMI, design and research by emphasizing firstly the task analysis and secondly the time execution of the task. In order to validate and support our results, an experimental study of user performance is conducted at the DOMUS laboratory, during the interaction with the contextual assistant’s interface. The results of our models show that the GOMS and ACT-R models give good and excellent predictions respectively of users performance at the task level, as well as the object level. Therefore, the simulated results are very close to the results obtained in the experimental study.

Keywords—HMI, interface evaluation, Analytical evaluation, cognitive modeling, user modeling, user performance.

I. INTRODUCTION

THE evaluation of human machine interfaces is becoming increasingly important. While their development presents some challenges, the evaluation of interfaces needs rigorous methods to ensure that they fulfill the initial specifications as well as the quality of accessibility and the usability of these interfaces [1], [2].

Two main approaches are currently used for the evaluation of HMI. The first one is empirical approaches, which are essentially based on performances or opinions of users gathered in laboratories or other experimental situations. The second one is analytical approaches, which are not based directly on the user performance, but on the interfaces’ examination using well defined structures and rigorous analytical techniques [3]. Analytical approaches allow to predict mainly user performance, time execution of tasks, performance design and the explanation of an existing interface’s performance [4]. Since these approaches can predict time execution of tasks, this latter should be accurately measured and evaluated. This can be done by adding each time the user interacts physically with the interface, either by the stroking on a keyboard, pointing with

the mouse on the screen, or even by pointing with a human finger on a touch screen.

Paul M Fitts has defined the law of physical components of the interaction with the interface by making the analogy between the interface and the target to reach [5]. However, reaching physically a component of the interface supposes that, a cognitive process was engaged before choosing the component to interact with. Therefore, the interaction process with the interface implies three human components. The first component is perceptual, which concerns more specifically the visual and aural perceptions in HMI. The second component is cognitive, implying the human to reason and retrieve in his memory the application of rules and the remembrance of objects in order to satisfy specific goals [6]. The third and the last component is motor, where the user reaches and interacts with the specific interface component.

The most important challenge of analytical methods is their capability to define and simulate the three components elicited in the HMI, in order to predict the users behavior.

The action of prediction can be performed using predictive models, which are an integral part of analytical approaches.

The GOMS is a predictive model, which estimates the time a user interacts with the interface, taking in account the time requested for the cognitive process to select the appropriate interaction [7]. ACT-R, a cognitive architecture, predicts the time needed to perceive stimulus, either aural or visual, to retrieve knowledge in memory and to execute the motor actions [6]. The two methods give opportunity to explain the way users accomplish goals.

In this study, we aim to evaluate the interaction with the interface of a contextual assistant application, developed to help persons with cognitive disabilities perform autonomously their daily living tasks. This application assists people while preparing meals in the kitchen by using cognitive assistance [8]. Due to the related population and the kind of errors they commit, we need to take in account the cognitive part involved in the interaction with the HMI. Then we use a powerful analytical methods based specifically on cognitive models to evaluate the contextual assistant’s interface, emphasizing the cognitive analysis of the tasks in one side, and the time execution of these tasks on the other side.

Our analytical evaluation is based on two methods. The first method simulates the task thanks to the cognitive architecture ACT-R [6] in which the interaction is decomposed in rules simulating the behavior of a human interacting with contextual assistant’s interface. The second method is the GOMS model (Goals, Operators, Methods and Selection rules), which is a formalized representation that can be used to predict task

Belkacem Chikhaoui is with the Domus Laboratory, Computer Science Department, Faculty of Science, University of Sherbrooke, Sherbrooke QC J1K 2R1 Canada, email: belkacem.chikhaoui@usherbrooke.ca

H el ene Pigot is with the Domus Laboratory, Computer Science Department, Faculty of Science, University of Sherbrooke, Sherbrooke QC J1K 2R1 Canada, email: helene.pigot@usherbrooke.ca

performance [9]. The GOMS model is a way in which users achieve goals by solving subgoals in a divide-and-conquer fashion [10].

In order to create an effective evaluation, an empirical study is conducted at the DOMUS laboratory over ten healthy persons. The results of our models are compared with those obtained in the experimental study.

After introducing a theoretical background about the concept of evaluation (section II), we present an overview of the analytical methods chosen to evaluate the contextual assistant's interface: the cognitive architecture ACT-R and the GOMS model (section III). The interface to be evaluated is presented in (section IV) and the experimental study is then introduced in (section V). The models developed are then presented in (section VI) and the results of the simulation are compared to the results obtained in the experimental study (section VII).

II. THEORETICAL BACKGROUND

The evaluation of systems focuses on two main aspects : the utility and the usability. The utility is defined as the question of whether the functionality of the system can do what is needed [11], and the usability is defined as the easiness of learning and using the system [12]. The evaluation of HMI ensures that the applications fulfill the users needs and requirements, and ensures that the interaction is motivate and enjoyable. Due to the usability problems detected during the evaluation process, more efficiency, adaptability and accessibility are expected in the interface [13], [14].

According to J. Preece and al, the evaluation can be defined as “ the process of systematically collecting data that informs us about what it is like for a particular user or group of users to use a product for a particular task in a certain type of environment” [12].

The evaluation is either empirical or analytical depending on the used methods. While the empirical methods evaluate the performance of an interface when users interact with it, the analytical methods simulate a user behavior based on theoretical knowledge.

The empirical methods are widely used in the literature to evaluate interfaces in various situations, either for traditional interaction with computers [15], [16], [17], mobile devices [18], [19] or pervasive interfaces [20]. Caution is needed for these methods to ensure that the subjects selected for the experimentation are representatives of the final users. During the test, the tasks to perform need also to be carefully designed to evaluate the way the final users will interact with the application. The evaluation of the interactions in real settings with a numerous set of people constitutes the trends of empirical evaluation. For example, the evaluation of cell phone menu's interaction requires fourteen experienced cell phone users performing tasks [18].

Pervasive computing systems involve different systems that make the process of evaluation difficult. The necessity to evaluate interfaces in real settings leads to long experiments when pervasive computing is evaluated at home, for instance three month period test by a young couple living in a smart apartment were needed to evaluate pervasive computing

system at home [19], [20]. Thus, the empirical evaluation necessitates high costs and time consuming.

To mitigate these drawbacks, analytical evaluation allows to simulate as much users as needed to perform various tasks on different versions of the interface. The analytical evaluation of HMI is based on theories and methods and the results bring a clear understanding of the way the users interact with the interface [21]. Analytical methods predict and identify practical errors and usability problems [14]. The analytical evaluation process should be conducted to evaluate the applications as well as the HMI. P Antunes and al propose to evaluate the groupware design (collaborative tool), which is a multi-user context using an analytical method derived from the GOMS model [22]. The tasks, users and the environment are then modeled. Therefore, different situations are simulated varying upon the different versions of the interface, the tasks to perform and the abilities of the user. According to St. Amant and al, the analytical evaluation of the smart phone menu's interaction increases the optimum version of menu on small screens [18]. The cognition evaluation should also demonstrate how the time is shared between the three components involved during the interaction [23]. Through ACT-R model, D. Salvucci demonstrates the impact of phone call during driving [24].

III. ANALYTICAL METHODS TO EVALUATE THE CONTEXTUAL ASSISTANT'S INTERFACE

The contextual assistant application aims to help people to complete daily living's activities. It is dedicated for people with cognitive deficits to foster autonomy, such as people with mental retardation. These users fail due to difficulties in planing, memory and attention. The interface must be specially designed for this population. However, the involvement of this kind of population during the evaluation process must remain limited. To do so, the researchers develop analytical methods to avoid numerous empirical evaluation problems as mentioned previously.

Due to the related population, the analytical approaches should emphasize the cognitive and perceptual processes required while using the contextual assistant. Therefore, we choose analytical methods based on cognitive theories, which are GOMS and ACT-R. Those methods are derived respectively from the human model processor theory and from the unified theories inspired by the work of Allen Newell [25], [26]. The objective of the study is to validate the cognitive simulation using these two methods and to analyze how they provide theoretical explanations upon the errors committed by the users. The two next sections present the analytical methods used in the simulation.

A. Cognitive Architecture ACT-R

The cognitive architecture ACT-R is built to simulate and understand human cognition [6], [27]. It consists of a set of modules such as the visual, aural, motor, intentional and declarative module that are integrated through a central production system. ACT-R is an hybrid architecture that combines two subsystems: symbolic system including semantic and

procedural knowledge, and subsymbolic system evaluating knowledge activations.

Each knowledge in the ACT-R's declarative memory is called chunk, and is associated with a level of activation computed by the subsymbolic system [26]. The activation level reflects the degree of availability of the chunk at any particular time. The subsymbolic system assigns also utility values to rules (procedural knowledge) to determine the predominant knowledge available at a specific time. The predominant knowledge is defined as the rule with the highest utility.

In ACT-R, the perceptual and motor modules are used to simulate interfaces between the cognitive modules and the real world. The perceptual modules allow the model to attend to visual and aural stimuli, while the motor modules are responsible for preparing and executing basic motor actions such as key presses and mouse movements [28], [29].

The visual module is decomposed in two subsystems, the positional system (where) and the identification system (what), that work together in order to send the visual stimulus to the visual buffer. The positional system is used to find objects. When a new object is detected, the chunk that represents the location of that object is placed in the visual-location buffer according to some constraints provided by the production rule. The identification system is used to attend to locations, which have been found by the positional system. The chunk represents a visual location that will request the identification system to shift visual attention to that location. The result of an attention operation is a chunk, which will be placed in the visual buffer [28], [29].

The motor module contains only one buffer through which it accepts requests. Two actions are available in ACT-R, to click with the mouse or press a key on a virtual keyboard.

B. GOMS model

GOMS is an acronym for Goals, Operators, Methods and Selection rules. It is a formalized method used to predict task performance [7], [9], [10]. A GOMS description consists of these 4 elements:

- 1) *Goals*: The user's goals describe what the user wants to achieve.
- 2) *Operators*: The basic actions that the user must perform in a lowest level of analysis in order to use the system.
- 3) *Methods*: Methods are sequences of steps consisting of operators and subgoal invocations that the user performs in order to accomplish a goal.
- 4) *Selection rules*: Selection rules choose the appropriate method depending on the context when choice of methods arises.

Each task is decomposed hierarchically in goals and subgoals according to the divide and conquer technique. The subgoals are also decomposed down until reaching the basic operations description. The total execution time is then estimated by summing the times of basic operations.

IV. CONTEXTUAL ASSISTANT

After having presented the two analytical methods selected to conduct our evaluation, we present now the application to

be evaluated.

The Contextual assistant is an application developed to assist persons with cognitive disabilities [30], [31]. The aim is to foster autonomy in daily living tasks, and particularly during complex cooking tasks such as preparing spaghetti [8]. The cooking task is decomposed of steps that are displayed on a touch screen. The two first steps consist of gathering the utensils and ingredients necessary to the recipe (Fig. 1). The other steps explicit the recipe using photo and video on the screen and also explicit the information dispatched all around the kitchen. The contextual assistant is specifically designed to help people remembering the places where the objects are stored. To do so, the contextual assistant contains an interface called the object locator that displays the objects to search. When an object is selected on the main interface, the contextual assistant looks for the location of that object in the environment using techniques of pervasive computing, and indicates the object location by highlighting the appropriate locker containing that object. In this study we simulate the first two steps of the spaghetti recipe. They consist of first, knowing the list of objects to gather, either utensils or ingredients, and second to use the object locator in order to locate each object in the environment.

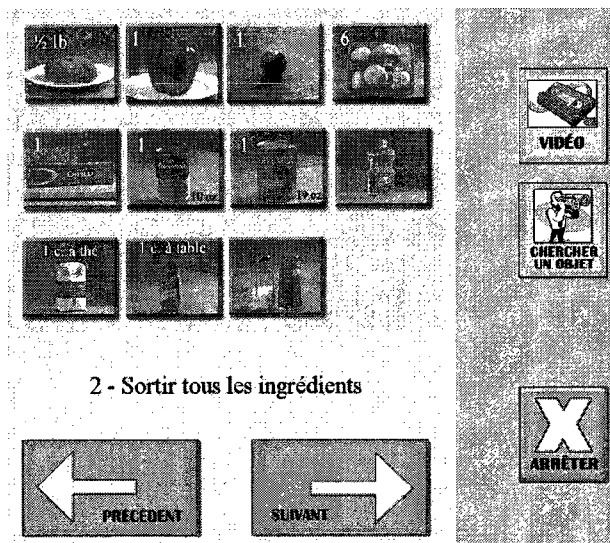


Fig. 1. The contextual assistant's interface representing the gathering ingredients task

The contextual assistant's interface is displayed on a 1725L 17" LCD Touchscreen, with 13.3" (338 mm) horizontal and 10.6" (270 mm) vertical useful screen area. It is configured to 1024 x 768 optimal native resolution running Macintosh. The screen is fixed under a closet nearby the oven in order to be easily accessible and also protected against the cooking splashes.

V. EXPERIMENTAL STUDY

In this section, we describe the conducted experiment at the DOMUS laboratory in terms of users, apparatus and applications used to perform this study.

A. Apparatus and Application

The experiment consists on selecting items in the contextual assistant's interface. The items correspond to the list of utensils and ingredients needed to realize the cooking task. Each item is displayed with a large button in the contextual assistant's interface (Fig. 1). The experiment is conducted according to three main criteria:

- 1) The time to select items is measured accurately.
- 2) The order in which the objects are displayed does not affect the speed of selection.
- 3) The experiment is uniform for all subjects.

To ensure the first criterion, we decide to isolate the time of object recognition and the time of selection in the contextual assistant's interface. Therefore, we measure the time needed to decide which object to get out and we measure the time to push on that object in the interface.

The first action corresponding on deciding which object users want to get out is presented experimentally using a PDA (Personal Digital Assistant). The name of the object to get out is displayed on the PDA in order to highlight the phase of objects recognition involved in the cognitive processes. To avoid automatic selection on the contextual assistant's interface, the names of objects displayed on the PDA are chosen randomly.

Knowing the object name, the subject executes the second phase which consists of pushing the correspondent button on the contextual assistant's interface. For each object needed in the experiment, the two phases' times are recorded in a log file and recovered at the end of experiments.

B. Subjects

Ten students of Sherbrooke's University participate to the study. All subjects are male and their ages range from 27 to 32 years. The subjects have good vision with no physical impairments being reported. All subjects have a good knowledge in computer science, but they have no prior knowledge in the application and the cognitive assistance field.

C. Method

The PDA is placed at a distance of 15 cm from the touch screen, subjects remain standing at a distance of approximately 30 cm from the touchscreen during the entire test as shown in Fig. 2.

The subjects familiarize themselves with the interface during a practical stage. When the test begins, the subjects look first on the PDA to know the name of the object to get out and second, push the corresponding button in the contextual assistant's interface using the index finger. To know the next object to reach, the subjects click on the PDA to display its name.

The experiment continues until the last object of the gathering objects' task is reached. The objects displayed on the PDA are presented to subjects under a random order. This emphasizes the recognition of object's phase in the cognitive process. Each subject accomplishes 5 trials, where a trial is composed of two tasks, which are gathering utensils and gathering

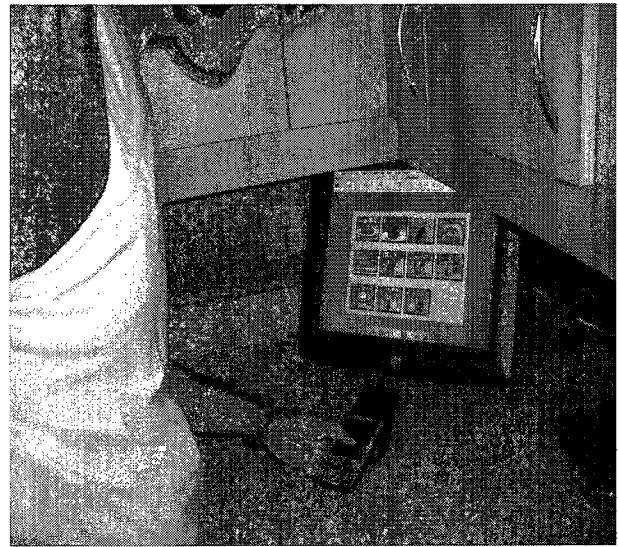


Fig. 2. The human machine interaction during the experimental study

ingredients. Each trial needs 25 actions "pressing on the PDA" and 25 actions "pushing button in the contextual assistant's interface". Altogether 2500 (10 subjects x 5 trials x (25 actions x 2 interfaces) = 2500) actions are observed during the experiment.

In our study, the action of getting out the objects from their locations in the environment is not modeled.

Table I shows the mean duration with the standard deviation in selecting each object in the two tasks, over all subjects in our study.

TABLE I
USER PERFORMANCE DATA ACROSS OBJECTS WITH MEAN AND STANDARD DEVIATION

Objects	Duration (s)	Standard Deviation (S)
LOOK-FOR-OBJECT (1)	5.299	1,052
CAN-OPENER	2.291	0,717
COLANDER	2.966	0,786
MEASURING-SPOON	2.167	0,605
LADLE	2.847	0,829
SMALL-SAUCEPAN	1.980	0,371
WOODEN-SPOON	2.590	0,536
KNIFE	2.328	0,430
BIG-SAUCEPAN	1.779	0,308
CUTTING-BOARD	2.000	0,309
HELP-ME-TO-DO-THE-TASK (1)	2.039	0,386
NEXT	2.142	0,540
LOOK-FOR-OBJECT (2)	1.955	0,265
PEPPER	2.448	0,825
SPAGHETTI	1.939	0,552
TOMATOES-BOX	1.794	0,377
GROUND-BEEF	2.491	0,591
ONION	2.021	0,484
TOMATO-SOUP	1.970	0,422
SALT-AND-PEPPER	2.490	0,481
OIL	1.965	0,348
MUSHROOMS	1.809	0,369
SUGAR	1.774	0,341
ITALIAN-SPICE	1.736	0,436
HELP-ME-TO-DO-THE-TASK (2)	2.432	0,614

VI. MODELING THE INTERACTION WITH THE CONTEXTUAL ASSISTANT

In this section, we present the modeling process of the tasks evaluated in our study, which are gathering utensils and gathering ingredients. After analyzing the tasks to modelize, we present first the model with ACT-R and second the model with GOMS.

A. Task analysis: gathering utensils and ingredients

The two first steps of the recipe gathering utensils and ingredients require three subtasks (Fig. 3). The first subtask consists of activating the object locator in order to locate each needed object for the recipe. This is done by pushing the button “LOOK-FOR-OBJECT (2)”, which is displayed in the main contextual assistant’s interface (Fig. 1). The second subtask consists of locating each object, either utensils or ingredients, needed in the current step by pushing the button corresponding to the object in the object locator. The third task consists of coming back to the main contextual assistant’s interface in order to know the next step of the recipe. The tree decomposition of the gathering ingredients’ task is presented in Fig. 3. The nodes in capital indicate the action to click on the named button, while the other nodes represent the tasks to be decomposed.

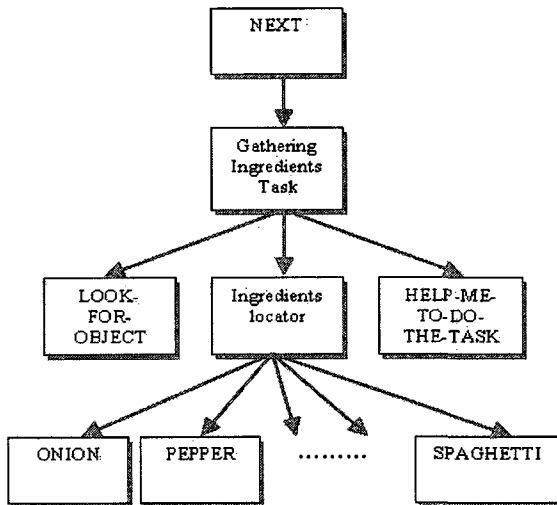


Fig. 3. Tree representation of the gathering ingredients task

B. Modeling The Interaction With The Contextual Assistant using ACT-R

The model uses ACT-R to emphasize the cognitive processes involved, when looking on an object and when choosing the button to push. The model is subdivided in three phases, the visual phase, the recognition phase and the motor phase. The visual phase consists of two steps: localizing the object to perceive and identifying it. We consider that all buttons displayed on the screen are objects, either the buttons used to locate a utensil or ingredient, or the buttons to navigate

in the interface. The first object is the button “LOOK-FOR-OBJECT (2)” as described in Fig. 3. Then, all the utensils (or ingredients) needed for the recipe are presented in the visual interface of ACT-R. Finally, to complete the current step of the recipe, the button “HELP-ME-TO-DO-THE-TASK (2)” is presented in order to come back to the main contextual assistant’s interface and pursue the next step of the recipe. Each object of the interface is displayed at defined coordinates (x, y) on the screen. These coordinates specify the made request to the visual-location buffer of ACT-R, which creates a chunk representing the location of the specified object. When the location’ step is over, the identification system identifies the name of the object and creates a chunk. This chunk is placed in the visual buffer. The steps of location and identification last 185 (ms) [28], [29]. The objects are presented to the visual module of ACT-R by the mean of a list of all the objects (buttons of the interface) to be pushed on.

The recognition’s phase begins when the chunk of the object is stored in the visual buffer. This phase implies to recover that specific chunk from the declarative memory. The result of this phase is a chunk that represents the object with some characteristics as color, localization on the screen, name, and kind of object. The motor’s phase consists of activating the motor actions through a request to the motor buffer in order to click on the object.

The three phases are applied for each object displayed in the interface for the two steps of the recipe. The gathering utensils and ingredients model finishes when the last object of the gathering ingredients’ task is reached.

In our ACT-R model, the contextual assistant’s interface is simulated using a virtual display based on a vertical list in the *Lisp* environment. The virtual display maintains a representation of each object used in the interface at a given time by displaying its name surrounded by a red circle, which reflects the shift attention to that object as shown in Fig. 4.

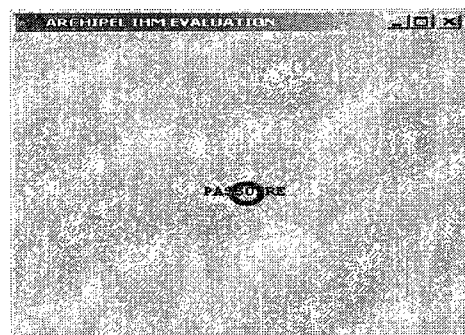


Fig. 4. Shift attention representation in the ACT-R model

The ACT-R model is developed using the ACT-R 6 environment. All memory chunks get the same value of activation and all requests to the retrieval buffer are correctly satisfied without errors. These characteristics lead to a deterministic model where no error could happen.

Fig. 5 shows in a very low detailed form an example of the execution trace of the ACT-R model. The visual phase is

principally based on the shift attention and the visual encoding actions, which are presented in the section I of Fig. 5 followed by the recognition phase in section II of Fig. 5. Finally, the motor phase is presented in section III of Fig. 5. The visual-location request takes place at time 0.050 (seconds) and the request to move-attention is made at time 0.100 (seconds). The encoding still needs 0.085 (seconds) to be completed and stores the chunk into the visual buffer. During the recognition phase, a retrieval request is made on the retrieval buffer in order to recover the specified chunk from the declarative memory. This phase will finish at time 0.397 (seconds). Finally a request on the motor buffer starts at time 0.447 (seconds). In the experimental study, users interact with the PDA and the screen. Each device involves three cognitive processes including the visual, cognitive and motor phase. Therefore, the ACT-R model simulates the time twice during the interaction with each object, on the PDA firstly and on the touch screen secondly. The simulation time is computed as the summation of the time estimated for each object to click on the PDA and on the touch screen.

Section 1. Visual Phase

```

0.000 GOAL SET-BUFFER-CHUNK GOAL FIRST-GOAL REQUESTED NIL
0.000 PROCEDURAL CONFLICT-RESOLUTION
0.050 PROCEDURAL PRODUCTION-FIRED START-APPLICATION
THE SUBJECT STARTS TO LOOK FOR NEW OBJECT
0.050 PROCEDURAL CLEAR-BUFFER IMAGINAL
0.050 PROCEDURAL CLEAR-BUFFER VISUAL-LOCATION
0.050 PROCEDURAL CLEAR-BUFFER GOAL
0.050 VISION Find-location
0.050 VISION SET-BUFFER-CHUNK VISUAL-LOCATION LOC1
0.050 GOAL SET-BUFFER-CHUNK GOAL GET-OBJECT0
0.050 PROCEDURAL CONFLICT-RESOLUTION
0.100 PROCEDURAL PRODUCTION-FIRED ATTEND-UTENSIL
SHIFT ATTENTION TO A SPECIFIED LOCATION ON THE SCREEN
0.100 PROCEDURAL CLEAR-BUFFER VISUAL-LOCATION
0.100 PROCEDURAL CLEAR-BUFFER VISUAL
0.100 PROCEDURAL CONFLICT-RESOLUTION
0.185 VISION Encoding-complete LOC1-0 NIL
0.185 VISION SET-BUFFER-CHUNK VISUAL TEXT1
0.185 PROCEDURAL CONFLICT-RESOLUTION

```

Section 2. Recognition Phase

```

0.235 PROCEDURAL PRODUCTION-FIRED ENCODE-UTENSIL
ENCODING THE OBJECT AFTER VISUAL ATTENTION
0.235 PROCEDURAL CLEAR-BUFFER VISUAL
0.235 PROCEDURAL CLEAR-BUFFER IMAGINAL
0.235 IMAGINAL SET-BUFFER-CHUNK IMAGINAL OBJECT1
0.235 PROCEDURAL CONFLICT-RESOLUTION
0.285 PROCEDURAL PRODUCTION-FIRED FOUND-OBJECT
RETRIEVE THE CORRESPONDING CHUNK FROM THE DECLARATIVE MEMORY
0.285 PROCEDURAL CLEAR-BUFFER IMAGINAL
0.285 PROCEDURAL CLEAR-BUFFER RETRIEVAL
0.285 DECLARATIVE START-RETRIEVAL
0.285 PROCEDURAL CONFLICT-RESOLUTION
0.397 DECLARATIVE RETRIEVED-CHUNK OBJECT1-0
0.397 DECLARATIVE SET-BUFFER-CHUNK RETRIEVAL OBJECT1-0
0.397 PROCEDURAL CONFLICT-RESOLUTION

```

Section 3. Motor Phase

```

0.447 PROCEDURAL PRODUCTION-FIRED MOTOR-ACTION
THE SUBJECT PUSHES THE CORRESPONDING ICON OF THE OBJECT
0.447 PROCEDURAL CLEAR-BUFFER VISUAL
0.447 PROCEDURAL CLEAR-BUFFER RETRIEVAL
0.447 PROCEDURAL CLEAR-BUFFER MANUAL

```

Fig. 5. Example of execution trace of the ACT-R model

C. Modeling The Interaction With The Contextual Assistant Using GOMS

The first two steps of the recipe, gathering utensils and gathering ingredients have been described previously, which can be interpreted in the GOMS language by a method that is divided in three steps as shown in Fig. 6. Each step defines a new goal to be reach.

```

Method_for_goal: Archipel Evaluation
    Step 1. Accomplish_goal: Select Utensils.
    Step 2. Accomplish_goal: Select Ingredients.
    Step 3. Return_with_goal_accomplished.

```

Fig. 6. Main method of the GOMS model

For each step in our study, a method is defined according to the concepts of GOMS methods in the definition of goals and subgoals. The GOMS model is based on a hierarchical representation of goals. In fact, by solving subgoals the user achieves goals until reaching the basic operations called “operators”, which can not be subdivided [10]. The methods have a hierarchical structure. Therefore, a method may call for subgoals to be accomplished [32]. Fig. 7 shows explicitly the subgoal’s invocations in the hierarchy of the “Select Ingredients” subgoal.

```

Method_for_goal: Select Ingredients
    Step 1. Accomplish_goal: Select NEXT.
    Step 2. Accomplish_goal: Select LOOK-FOR-OBJECT(2).
    Step 3. Accomplish_goal: Select ONION.
    Step 4. Accomplish_goal: Select PEPPER.
    Step 5. Accomplish_goal: Select GROUND-BEEF.
    Step 6. Accomplish_goal: Select MUSHROOMS.
    Step 7. Accomplish_goal: Select TOMATO-SOUP.
    Step 8. Accomplish_goal: Select ITALIAN-SPICE.
    Step 9. Accomplish_goal: Select OIL.
    Step 10. Accomplish_goal: Select SUGAR.
    Step 11. Accomplish_goal: Select SALT-AND-PEPPER.
    Step 12. Accomplish_goal: Select TOMATOES-BOX.
    Step 13. Accomplish_goal: Select SPAGHETTI.
    Step 14. Accomplish_goal: Select HELP-ME-TO-DO-THE-TASK(2).
    Step 15. Return_with_goal_accomplished.

```

Fig. 7. GOMS Method for Select Ingredients task

The main method presented in Fig. 6 constitutes the root of the tree hierarchy and all the other methods are generated automatically using the divide-and-conquer technique [10]. In our GOMS model, each object is defined as visual object. The selected methods have the same form for all objects. Fig. 8 explains the tree decomposition corresponding to the main method of the GOMS model.

The duration of a step in the GOMS model can be defined as the sum of the production cycle’s duration and the duration of all actions included inside the step. Therefore, the production cycle’s duration equals to 50 (ms) and for instance, the performance of key presses is estimated to 280 (ms) [33], [18].

Our GOMS model is executed using the GLEAN3 modeling tool [33].

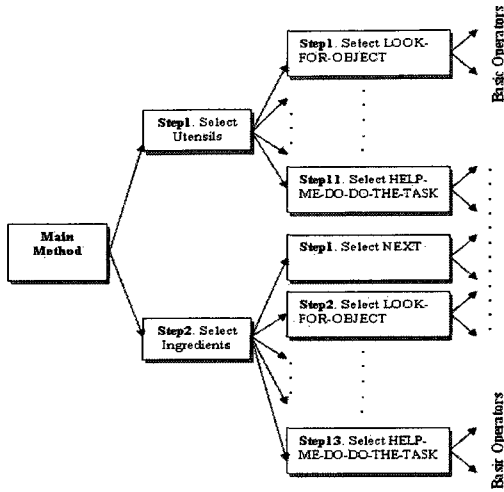


Fig. 8. Tree decomposition of the main method in the GOMS model

VII. COMPARISON OF RESULTS

We describe the performance of our models at two levels: the accuracy with which our models predict the overall duration of the tasks, and the accuracy to predict the duration to push each object displayed in the interface.

A. Object level performance

Table II shows the comparison of the time needed by the user, the ACT-R and GOMS model predictions. Values in parentheses represent the smallest and greatest time needed by the user to press each object. The objects “LOOK-FOR-OBJECT (1)” and “LOOK-FOR-OBJECT (2)” are displayed respectively on the gathering utensils and gathering ingredients interfaces and the same thing is applied for the objects “HELP-ME-TO-DO-THE-TASK (1)” and “HELP-ME-TO-DO-THE-TASK (2)”. Fig. 9 and Fig. 10 show respectively the predicted time of each object during the gathering utensils and gathering ingredients tasks in a detailed graphical form. According to Fig. 9 and Fig. 10, the results of both ACT-R and GOMS models are very close and have approximately the same predicted time values for several objects.

B. Task level performance

Table III shows the user performance data, the ACT-R and GOMS model predictions in both tasks: gathering utensils and gathering ingredients. Fig. 11 shows the same data in a detailed graphical form.

Fig. 12 shows the progression in performing tasks over the time in the experimental study, ACT-R and GOMS models. Since the prediction’s procedure is applied for each object in the interface, the two models follow a linear model. This is supported by some scientific literature [18]. The predicted time in both ACT-R and GOMS models is very close depending on the time progression of tasks of user performance as shown in Fig. 12.

TABLE II
COMPARISON OF USER DATA, ACT-R AND GOMS MODEL PREDICTIONS
BY OBJECT

Objects	User	ACT-R	GOMS
	Performance (s)	(S)	(S)
LOOK-FOR-OBJECT (1)	5.299 (3.838 7.052)	2.230	4.250
CAN-OPENER	2.291 (1.308 3.676)	2.165	2.550
COLANDER	2.966 (1.703 4.130)	2.250	2.550
MEASURING-SPOON	2.167 (1.206 3.262)	2.250	2.550
LADLE	2.847 (1.434 4.143)	2.165	2.550
SMALL-SAUCEPAN	1.980 (1.351 2.652)	2.080	2.550
WOODEN-SPOON	2.590 (1.561 3.622)	2.250	2.550
KNIFE	2.328 (1.676 2.947)	2.165	2.550
BIG-SAUCEPAN	1.779 (1.284 2.330)	2.165	2.550
CUTTING-BOARD	2.000 (1.600 2.451)	2.165	2.550
HELP-ME-TO-DO-THE-TASK (1)	2.039 (1.349 2.680)	2.165	2.550
NEXT	2.142 (1.289 3.082)	2.165	2.550
LOOK-FOR-OBJECT (2)	1.955 (1.384 2.362)	2.165	2.650
PEPPER	2.448 (1.159 3.847)	2.250	2.550
SPAGHETTI	1.939 (1.311 3.180)	2.165	2.550
TOMATOES-BOX	1.794 (1.265 2.382)	2.165	2.550
GROUND-BEEF	2.491 (1.634 3.643)	2.165	2.550
ONION	2.021 (1.391 2.906)	2.165	2.550
TOMATO-SOUP	1.970 (1.373 2.714)	2.165	2.550
SALT-AND-PEPPER	2.490 (1.546 3.227)	2.165	2.550
OIL	1.965 (1.389 2.544)	2.165	2.550
MUSHROOMS	1.809 (1.232 2.477)	2.250	2.550
SUGAR	1.774 (1.213 2.300)	2.250	2.550
ITALIAN-SPICE	1.736 (1.217 2.772)	2.250	2.550
HELP-ME-TO-DO-THE-TASK (2)	2.432 (1.265 3.654)	2.165	2.550

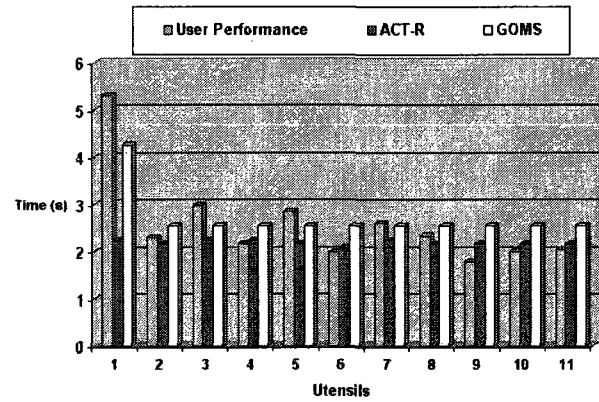


Fig. 9. User data, ACT-R and GOMS model predictions for the gathering utensils task

VIII. GENERAL DISCUSSION

The ACT-R and GOMS models, which we developed have proved robust and efficient. In fact, the results of both models are very close to the user performance data obtained in the experimental study. The GOMS and ACT-R models give good to excellent predictions of time execution of tasks as well as objects as shown respectively in Table II and Table III.

As shown in Table II, the object “LOOK-FOR-OBJECT (1)” needs more time to be pushed using the GOMS model

TABLE III
COMPARISON OF USER DATA, ACT-R AND GOMS MODEL PREDICTIONS
BY TASK

Task	User	ACT-R	GOMS
	Performance (s)	(s)	(s)
Gathering utensils (11 objects)	28.290	24.050	29.750
Gathering ingredients (14 objects)	28.973	30.650	35.800

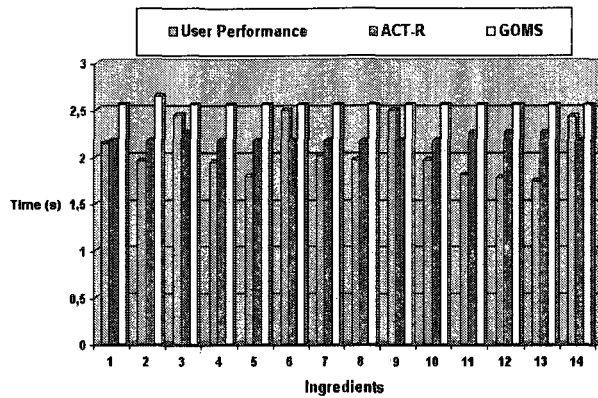


Fig. 10. User data, ACT-R and GOMS model predictions for the gathering ingredients task

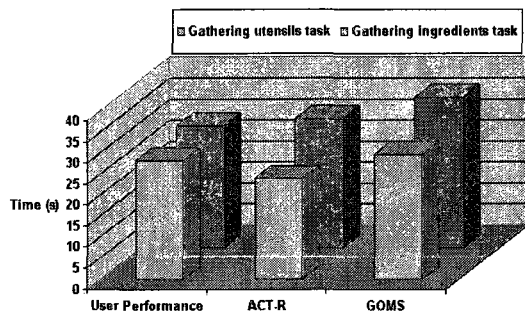


Fig. 11. User data, ACT-R and GOMS model predictions by task

4.250 (seconds) than the ACT-R model 2.230 (seconds). This significant difference can be interpreted by the fact that the GOMS model includes mental operator at the beginning of the gathering utensils task. This mental operator takes more than one second to be accomplished. The same difference is observed in the predicted time of the object "LOOK-FOR-OBJECT (2)" with the GOMS model. This object necessitates more time to be pushed 2.650 (seconds), which can be interpreted by the addition of a mental operator at the beginning of the task gathering ingredients.

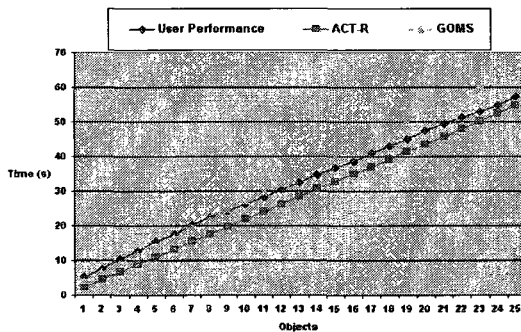


Fig. 12. Progression in accomplishing tasks over the time

Some differences in the predicted time of some objects using the ACT-R model are observed in Table II. This is due to several rules such as visual processing when a new object is detected in the visual field, information retrieval and motor actions. The visual part in the ACT-R model is explicitly defined using requests to the visual buffers unlike the GOMS model in which the visual part is implicitly defined. Both ACT-R and GOMS models do not take into account the fact that the objects are not displayed in the same location on the screen, but subjects in the experimental study performed differently depending on the exact position of the objects on the screen. For instance, the object "PEPPER", which is displayed at the top of the screen as shown in Fig. 1, needs more time to be selected 2.448 (seconds) than the object "TOMATOES-BOX", which is displayed in the center of the screen and needs 1.794 (seconds) to be selected.

In Table III, an important remark must be mentioned. The time taken to accomplish the first task in the experimental study, which equals to 28.290 (seconds) is very close to the time taken to accomplish the second task, which equals to 28.973 (seconds). Although the number of objects needed for the first task (11 objects) is lower than the one needed for the second task (14 objects). Two interpretations are possible:

- 1) The first interpretation is that the user learns gradually the position of objects in the interface and the navigation in the interface. Due to the learning, the second task will be performed faster than the first one and the last experiments will be performed generally faster than the first experiments.
- 2) The second interpretation is related to the nature of objects displayed in the contextual assistant's interface. At each name given on the PDA corresponds an image on the contextual assistant's interface. Some objects of the first task seem to be more difficult to identify such as the one shown in Fig. 13. This should provoke delay in the execution of the tasks.



Fig. 13. Example of an object not easily identifiable

Our results show that the evaluation of HMI designed for persons with cognitive disabilities at a detailed low level is possible using cognitive modeling techniques, particularly ACT-R and GOMS models.

IX. CONCLUSION

This study empirically demonstrated that cognitive models are a powerful tool for evaluating interfaces and predicting user's performance. The main goal of our study is to build and validate models for the evaluation of the contextual assistant's interface by simulating the HMI focusing on the time execution of tasks. We used two efficient and powerful cognitive models to evaluate the specified interface. The first

model is based on the cognitive architecture ACT-R and the second one is based on the GOMS model. Table III shows that both models ACT-R and GOMS give good approximations of user performance at the task level. The results of our models are considered suitable and correct comparing them to the user performance data obtained in the experimental study. The results show that the GOMS model can predict user's performance at good level and the ACT-R model can predict user's performance at more detailed level and performs almost as well. Our models are powerful and realistic as demonstrated with the comparison of the time taken by subjects performing the same tasks.

According to these results, the two models could be used to improve the design of the contextual assistant's interface and to optimize it.

During the conception of the GOMS and ACT-R models, we observed that the GOMS model gives more flexibility in modeling than the ACT-R model, which constitutes the complicated part in our study. However, the ACT-R model proposes a more accurate explanation about the cognitive processes involved during the interaction with the contextual assistant's interface, and hence, the possibility of introducing cognitive errors.

Our study makes two main contributions, the first contribution is to design an analytical evaluation of HMI designed for cognitively impaired people. It constitutes a new study in that field. The second contribution is the use of cognitive models to evaluate these interfaces emphasizing on the cognitive processes involved during the human machine interaction.

X. FUTURE IMPROVEMENTS

Some improvements should be brought to our models. First, our models are deterministic and do not make errors. They should be extended to allow errors in the pointing actions such as pushing an object several times before or after looking for the location of that object in the environment or pushing an object instead of another one. These errors are essentially related to memory problems that may occur in the task modeling and during the interaction with the contextual assistant's interface [20], [34], [35].

Second, since the contextual assistant is designed to assist cognitively impaired people in smart homes, it would be interesting to do some experiments with this population. The comparison between the results of a non deterministic model and the experiments' results allows us to study the behavior of our models in real situations and to evaluate their performance and effectiveness.

Finally, the action of searching an object is summarized into the human machine interaction with the touch screen. The contextual assistant offers an interaction with the environment to help people recovering utensils and ingredients dispatched in the kitchen. It would be interesting in the future to model this part and simulate the movement of users picking up the objects in the kitchen.

REFERENCES

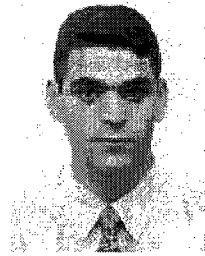
- [1] J. Nielsen and V. L. Phillips, "Estimating the relative usability of two interfaces: heuristic, formal, and empirical methods compared,"

Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems, pp. 214–221, 1993.

- [2] B. D. Eugenio, S. Haller, and M. Glass, "Development and evaluation of nl interfaces in a small shop," *AAAI Spring Symposium on Natural Language Generation in Spoken and Written Dialogue*, pp. 1–8, 2003.
- [3] B. Yen, P. Hu, and M. Wang, "Towards effective web site designs: A framework for modeling, design evaluation and enhancement," *Proceedings of IEEE International Conference on e-Technology, e-Commerce, and e-Service (EEE05)*, pp. 1–6, 2005.
- [4] L. M. Leventhal and J. A. Barnes, *Usability Assessment*, pearson prentice hall ed., ser. Usability Engineering: Process, Products, and Examples, 2007, ch. 11, pp. 206–220.
- [5] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement," *Journal of Experimental Psychology*, vol. 47, NO. 6, pp. 381–391, 1954.
- [6] J. R. Anderson, D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, and Y. Qin, "An integrated theory of the mind," *Psychological Review*, vol. 111, 136–1060, 2004.
- [7] B. E. John and D. E. Kieras, "Using goms for user interface design and evaluation: Which technique?" *ACM Transactions on Computer-Human Interaction*, vol. 3, pp. 287–319, 1996.
- [8] H. Pigot, D. Lussier-Desrochers, J. Bauchet, Y. Lachapelle, and S. Giroux, "A smart home to assist recipes' completion." in *Festival of International Conferences on Caregiving, Disability, Aging and Technology (FICCDAT), 2nd International Conference on Technology and Aging (ICTA)*, 2007.
- [9] D. E. Kieras, "Goms models for task analysis," *Handbook of task analysis for human-computer interaction*, Lawrence Erlbaum Associates, no. Diaper D, Stanton N, pp. 83–116, 2003.
- [10] A. Dix, J. Finlay, G. D. Abowd, and R. Beale, *Human-Computer Interaction Third edition*. England: Pearson, Prentice-Hall, Inc, 2004.
- [11] J. Nielsen, *Usability Engineering*. Boston, MA: Academic Press, 1993.
- [12] J. Preece, Y. Rogers, and H. Sharp, *Introducing evaluation*, ser. Interaction Design: Beyond Human-Computer Interaction. John Wiley and Sons, 2002, ch. 10, pp. 317–337.
- [13] J. Nielsen and R. Molich, "Heuristic evaluation of user interfaces," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM New York, NY, USA, 1990, pp. 249–256.
- [14] L.-O. Bligard and A.-L. Osvalder, *An Analytical Approach for Predicting and Identifying Use Error and Usability Problem*, a. holzinger. ed., ser. HCI and Usability for Medicine and Health Care, 2007, vol. 4799/2007, ch. 38, pp. 427–440.
- [15] C. Stephanidis, D. Akoumianakis, M. Sfyarakis, and A. Paramythi, "Universal accessibility in hci: Process-oriented design guidelines and tool requirements," in *Proceedings of the 4th ERCIM Workshop on User Interfaces for All*, 1998, pp. 1–15.
- [16] A. C. Siochi and D. Hix, "A study of computer-supported user interface evaluation using maximal repeating pattern analysis," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1991, pp. 301–305.
- [17] J. H. Goldberg and X. P. Kotval, "Computer interface evaluation using eye movements: methods and constructs," *International Journal of Industrial Ergonomics*, vol. 24, no. 6, pp. 631–645, 1999.
- [18] R. S. Amant, T. E. Horton, and F. E. Ritter, "Model-based evaluation of expert cell phone menu interaction," *ACM Transactions on Computer-Human Interaction*, vol. 14(1), pp. 1–24, 2007.
- [19] T. Koskela, K. V.-V. Mattila, and L. Lehti, "Home is where your phone is: Usability evaluation of mobile phone ui for a smart home," in *Proceedings of MobileHCI 2004*, vol. 3160, 2004, pp. 74–85.
- [20] B. E. John and D. D. Salvucci, "Multi-purpose prototypes for assessing user interfaces in pervasive computing systems," *IEEE pervasive computing*, vol. 4, no4, pp. 27–34, 2005.
- [21] B. Chikhaoui and H. Pigot, "Simulation of a human machine interaction: Locate objects using a contextual assistant," in *proceedings of the 1st International North American Simulation Technology Conference*, M. Beldjehem, Ed., 2008, pp. 75–80.
- [22] P. Antunes, M. R. S. Borges, J. A. Pino, and L. Carrio, *Analytic Evaluation of Groupware Design*, ser. Computer Supported Cooperative Work in Design II, 2006, vol. 3865/2006, ch. 4, pp. 31–40.
- [23] B. Chikhaoui and H. Pigot, "Evaluation of a contextual assistant interface using cognitive models," in *proceedings of the 5th International Conference on Human-Computer Interaction*, waset, Ed., vol. 34, 2008, pp. 36–43.
- [24] D. D. Salvucci and K. L. Macuga, "Predicting the effects of cellular-phone dialing on driver performance," *Cognitive Systems Research*, vol. 3, no. 1, pp. 95–102, 2002.

- [25] B. E. John and D. E. Kieras, "The goms family of analysis techniques: Tools for design and evaluation," Carnegie Mellon University School of Computer Science, Technical Report CMU-CS-94-181, 1994.
- [26] A. Newell, *Unified Theories of Cognition*. Cambridge (Mass): Harvard University Press, 1990.
- [27] J. R. Anderson, N. A. Taatgen, and M. D. Byrne, "Learning to achieve perfect time sharing: Architectural implications of hazeltine, teague ivry (2002)," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 31, No. 4, pp. 749–761, 2005.
- [28] M. D. Byrne, "Act-r/pm and menu selection: Applying a cognitive architecture to hci," *International Journal of Human-Computer Studies*, vol. 55, pp. 41–84, 2001.
- [29] D. Bothell, "Act-r 6.0 reference manual," Working Draft, 2004.
- [30] H. Pigot, J. Bauchet, and S. Giroux, *Assistive devices for people with cognitive impairments*, ser. The Engineering Handbook on Smart Technology for Aging, Disability and Independence. John Wiley and Sons, 2007, ch. 12.
- [31] D. Lussier-Desrochers, Y. Lachapelle, H. Pigot, and J. Bauchet, "Apartments for people with intellectual disability: Promoting innovative community living services," in *Proceedings of the 2nd International Conference on Intellectual Disabilities/Mental Retardation.*, 2007.
- [32] B. E. John and D. E. Kieras, "The goms family of user interface analysis techniques: Comparison and contrast." *ACM Transactions on Computer-Human Interaction*, vol. 3, pp. 320–351, 1996.
- [33] D. E. Kieras, S. D. Wood, K. Abotel, and A. Hornof, "Glean: A computer-based tool for rapid goms model usability evaluation of user interface designs," *Proceedings of the 8th ACM Symposium on User Interface Software and Technology*, no. ACM, pp. 91–100, 1996.
- [34] A. Serna, H. Pigot, and V. Rialle, "Modeling the performances of persons suffering alzheimers disease on an activity of the daily living," in *18th Congress of the International Association of Gerontology*, 2005.
- [35] A. Dion and H. Pigot, "Modeling cognitive errors in the realization of

an activity of the everyday life," in *Cognitio*, 2007.



Belkacem Chikhaoui received the Eng. degree in computer science from the University of Boumerdès, Algeria, in 2004. He is currently pursuing the Master degree in computer science at the University of Sherbrooke, Sherbrooke, QC, Canada. His research interests include smart homes, cognitive modeling and evaluation of Human-Machine Interfaces.



Dr. Hélène Pigot is professor in computer science at Sherbrooke University, Canada. She co-founded the DOMUS laboratory. She is director of the Center on smart environment of the Sherbrooke University. Hélène Pigot received his PhD on speech recognition in 1985, at P.M. Curie University, France. She was then graduated in occupational therapy at Montreal University, Canada. Professor at Montreal University during five years. she worked on spatial orientation deficits in Alzheimer disease. Since 2000, she dedicates her research on smart home, cognitive modeling and cognitive assistance for people with cognitive deficits.

Chapitre 4

Résultats et discussion

4.1 Résultats

Les résultats obtenus par les trois modèles, et l'étude expérimentale sont présentés dans le tableau 2 :

Tableau 2 – Résultats obtenus par l'étude expérimentale et les trois modèles

Objets	Étude expérimentale (s)	ACT-R (s)	GOMS (s)	Loi de Fitts (s)
CHERCHER UN OBJET (1)	5,299	2,230	4,250	1,328
OUVRE BOITE	2,291	2,165	2,550	1,228
PASSOIRE	2,966	2,250	2,550	1,228
CUILLERE A MESURER	2,167	2,250	2,550	1,228
LOUCHE	2,847	2,165	2,550	1,228
PETITE CASSEROLE	1,980	2,080	2,550	1,228
CUILLERE DE BOIS	2,590	2,250	2,550	1,228
COUTEAU	2,328	2,165	2,550	1,228
GRANDE CASSEROLE	1,779	2,165	2,550	1,228
PLANCHE A DECOUPER	2,000	2,165	2,550	1,228
M'AIDER A FAIRE LA TACHE (1)	2,039	2,165	2,550	1,328
SUIVANT	2,142	2,165	2,550	1,168
CHERCHER UN OBJET (2)	1,955	2,165	2,650	1,328
POIVRON	2,448	2,250	2,550	1,228
SPAGHETTI	1,939	2,165	2,550	1,228
BOITE DE TOMATES	1,794	2,165	2,550	1,228
BOEUFHACHE	2,491	2,165	2,550	1,228
OIGNON	2,021	2,165	2,550	1,228
SOUPE AUX TOMATES	1,970	2,165	2,550	1,228
SEL ET POIVRE	2,490	2,165	2,550	1,228
HUILE	1,965	2,165	2,550	1,228

CHAMPIGNONS	1,809	2,250	2,550	1,228
SUCRE	1,774	2,250	2,550	1,228
EPICES ITALIENS	1,736	2,250	2,550	1,228
M'AIDER A FAIRE LA TACHE (2)	2,432	2,165	2,550	1,328

Les résultats de la loi de Fitts, sont obtenus par un ajustement du premier modèle présenté dans le chapitre 2, en ajoutant la deuxième composante physique relative à l'interaction de l'utilisateur avec le PDA. Le temps de sélection d'un objet dans l'interface de l'assistant contextuel est calculé en additionnant le temps de reconnaissance de l'objet sur le PDA, et le temps de sélection de l'objet sur l'écran tactile selon la formule suivante :

$$TM = \text{TEMPS}_{\{PDA\}} + \text{TEMPS}_{\{ÉCRAN TACTILE\}} \quad (1)$$

Avec :

$$\text{TEMPS}_{\{PDA\}} = a_1 + b_1 \text{Log}_2 (A_1 / W_1 + 1) \quad (2)$$

$$\text{TEMPS}_{\{ÉCRAN TACTILE\}} = a_2 + b_2 \text{Log}_2 (A_2 / W_2 + 1) \quad (3)$$

Tels que : a_1 , b_1 , a_2 et b_2 sont des constantes qui peuvent être déterminées empiriquement.

A_1 : est la distance entre le point de départ et la cible sur le PDA

W_1 : est la largeur de la cible sur le PDA.

A_2 : est la distance entre le point de départ et la cible sur l'écran tactile.

W_2 : est la largeur de la cible sur l'écran tactile.

Une représentation graphique des résultats obtenus est illustrée par les figures 2 et 3.

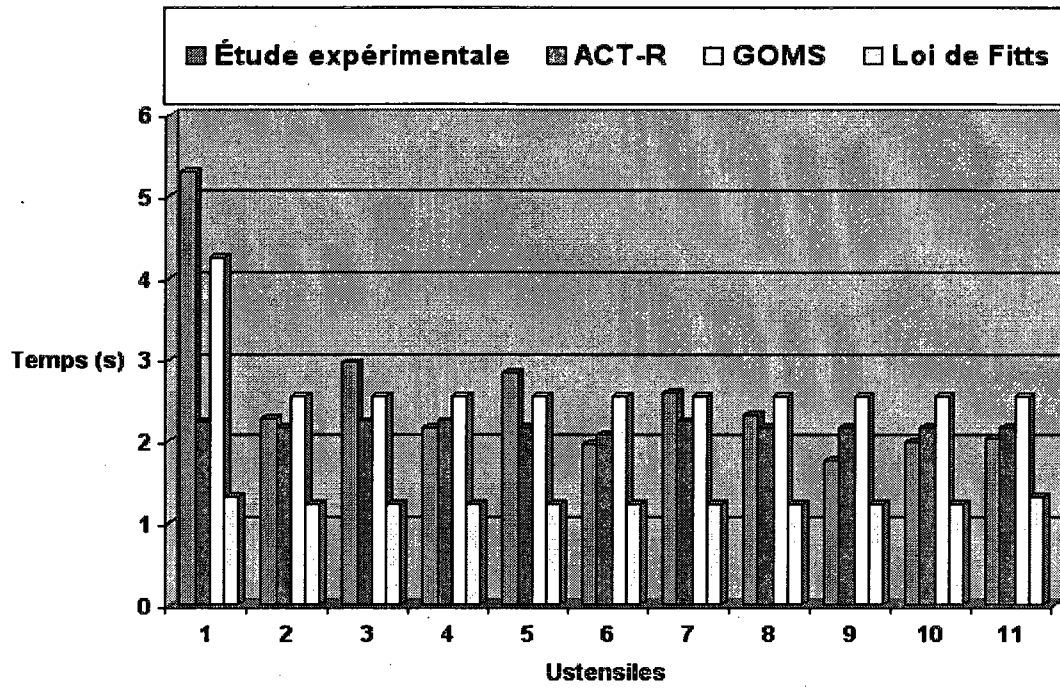


Figure 4 - Prédiction de temps de sélection des objets de la première tâche

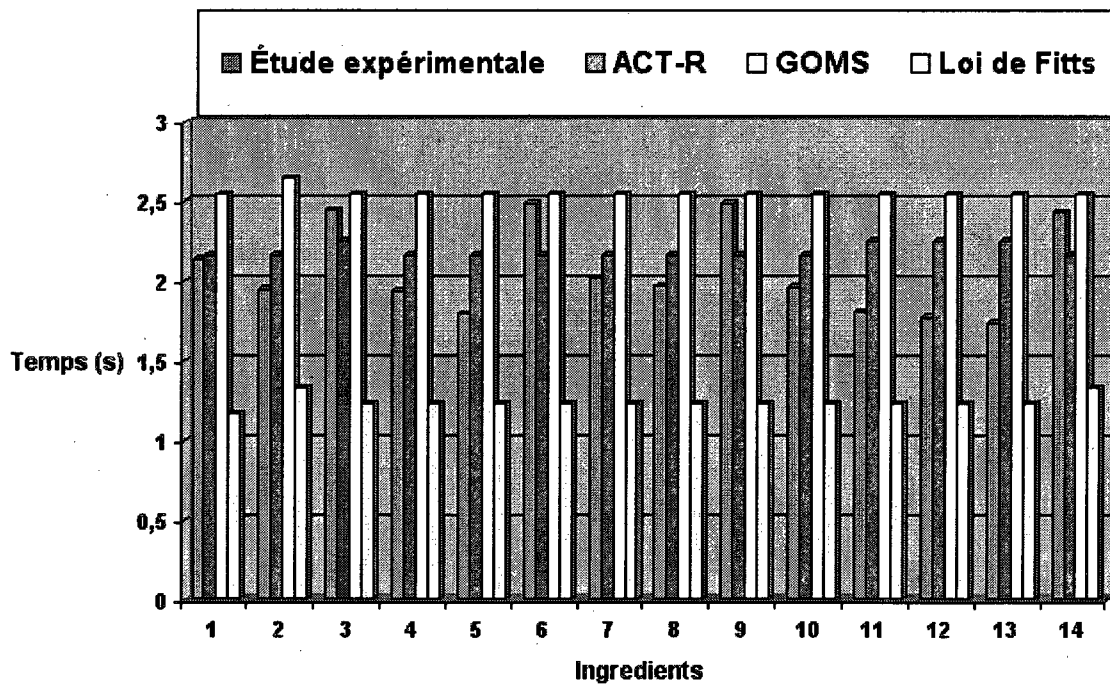


Figure 5 - Prédiction de temps de sélection des objets de la deuxième tâche

4.2 Discussion

Dans notre projet, on a utilisé trois modèles pour effectuer une évaluation analytique adéquate des interfaces homme machine de l'assistant contextuel. Parmi ces trois modèles, deux sont basés sur des modèles cognitifs et le troisième est basé sur le modèle de la loi de Fitts.

Tous les modèles se montrent performants et efficaces dans notre projet. Les modèles cognitifs sont beaucoup plus performants que le modèle de la loi de Fitts, et donnent une très bonne prédiction des performances utilisateurs à un niveau de détail plus élevé.

Les résultats obtenus nous renseignent d'une part, sur la prédiction du temps de sélection des objets, et d'autre part, sur la prédiction du temps d'exécution des tâches. Deux niveaux de prédiction de temps sont alors à mettre en avant.

4.2.1 Prédiction au niveau des objets

Au niveau des objets, le temps de prédiction pour sélectionner chaque objet utilisant la loi de Fitts est nettement petit en comparaison avec celui obtenu par les modèles cognitifs ACT-R et GOMS. Par exemple, l'objet « Ouvre-boite » requiert un temps de (2.165 secondes) pour être sélectionné en utilisant le modèle ACT-R, et requiert un temps de (2.550 secondes) en utilisant le modèle GOMS. Cependant, le même objet requiert uniquement un temps de (1.228 secondes) pour être sélectionné en utilisant le modèle de la loi de Fitts. En conséquence, Le temps de prédiction pour sélectionner cet objet en utilisant les modèles GOMS, respectivement ACT-R est proche, respectivement très proche du temps de la sélection de l'objet dans l'étude expérimentale, qu'est égal à (2.291 secondes).

Le temps de prédiction avec la loi de Fitts, dépend principalement de la taille des objets affichés sur l'écran, et la distance à ces objets. Autrement dit, le temps de prédiction est proportionnel à la distance aux objets, et inversement proportionnel aux tailles de ces

derniers. Dès lors, la loi de Fitts s'intéresse à la composante physique de l'interaction homme machine, i.e. elle modélise uniquement l'act de pointage.

À l'inverse de la loi de Fitts, les modèles ACT-R et GOMS mettent l'emphase sur les processus cognitifs impliqués lors de l'interaction avec l'interface de l'ACD tels que le processus visuel, cognitif et manuel, et non plus sur la taille des objets.

Quelques différences dans le temps de prédiction sont observées. Tel que présenté dans le Tableau 2, l'objet « Chercher-un-objet (1) » requiert plus de temps pour être sélectionné en utilisant le modèle GOMS (4.250 secondes) qu'avec le modèle ACT-R (2.230 secondes). Cette différence significative peut être interprétée par le fait que, le modèle GOMS inclut un opérateur mental au début de chaque tâche, qui sert à l'initialisation de cette tâche. Et comme la première tâche commence par sélectionner l'objet « Chercher-un-objet (1) », l'opérateur mental est alors ajouté, qui durera (1.2 secondes).

La même explication est valable pour l'objet « Chercher-un-objet (2) » qui requiert un temps de (2.650 secondes) pour être sélectionné en utilisant le modèle GOMS. Cet objet constitue le début de la deuxième tâche.

4.2.2 Prédiction au niveau des tâches

Les résultats obtenus par les trois modèles sont proches à ceux obtenus par l'étude expérimentale, et particulièrement les modèles ACT-R et GOMS qui donnent une très bonne prédiction à ce niveau, tel que présenté dans le Tableau 3.

Tableau 3 – Prédiction du temps d'exécution des tâches dans chaque modèle

Tâches	Étude expérimentale (s)	ACT-R (s)	GOMS (s)	Loi de Fitts (s)
Faire sortir tous les ustensiles	27.193	24.050	29.750	13.708
Faire sortir tous les ingrédients	27.236	30.650	35.800	17.332

Les résultats obtenus sont présentés graphiquement dans la figure 6.

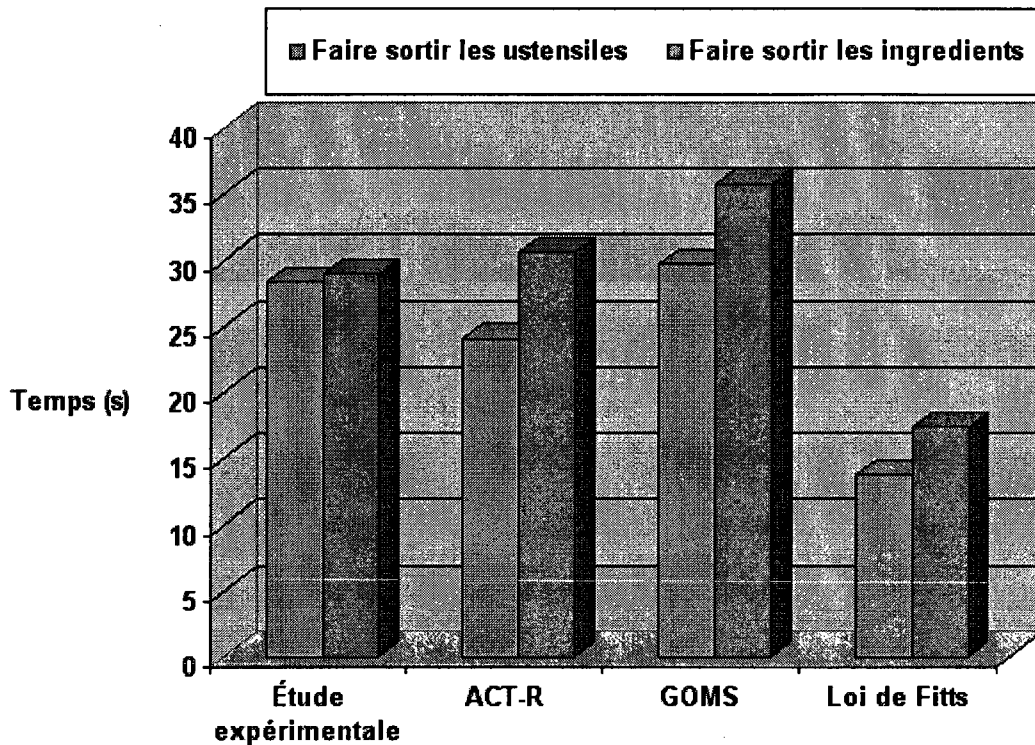


Figure 6 - Prédiction du temps d'exécution des tâches dans chaque modèle

Le Tableau 3 fait apparaître un constat très intéressant concernant le temps d'exécution obtenus pour les deux tâches durant l'étude expérimentale. On remarque que le temps d'exécution nécessaire pour accomplir la première tâche (27.193 secondes) est très proche de celui nécessaire pour accomplir la deuxième tâche (27.236). Deux interprétations possibles :

- 1) La première interprétation est que l'utilisateur apprend la position des objets ainsi que la navigation dans l'interface de l'ACD avec l'avancement du temps. Cela veut dire qu'avec cet apprentissage, la deuxième tâche sera plus rapide que la première, et les dernières expérimentations seront plus rapides que les premières.
- 2) La seconde interprétation est liée beaucoup plus aux objets en tant que tels, affichés dans l'interface de l'ACD. Les objets sont affichés chacun avec une image

correspondante. Quelques objets de la première tâche ne sont pas facilement identifiables, et la ressemblance entre quelques autres objets, créent une lenteur dans l'exécution, et particulièrement pour la première tâche de la recette comme le montre la Figure 7, ce qui explique le temps pris par les utilisateurs pour sélectionner l'objet en question tel que présenté dans le Tableau 2.

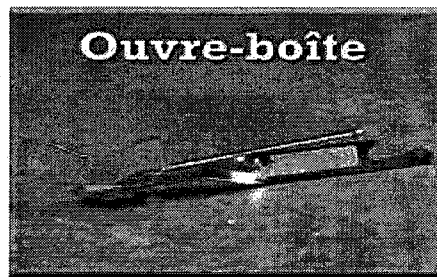


Figure 7 - Exemple d'un objet difficilement identifiable

Conclusion

Dans notre recherche menée sur les interfaces homme machine, développées dans le contexte des habitats intelligents au profit des personnes présentant des déficits cognitifs, une évaluation ergonomique de ces interfaces est indispensable. Après avoir dessiné le contour général de la problématique d'évaluation d'interfaces, différentes approches d'évaluation dans la littérature et qui peuvent mettre en œuvre cette problématique ont été introduites, ainsi que les méthodes sur lesquelles se basent ces approches.

Après une étude approfondie des approches d'évaluation d'interfaces, on a opté pour les approches analytiques qui sont basées sur l'examen et l'analyse des interfaces, sans faire recours aux utilisateurs, et qui permettent de mettre en œuvre cette évaluation en minimisant à la fois le coût et le temps observés dans les autres approches.

Notre recherche s'est intéressée en premier lieu à la simulation des interfaces homme machine, qui constitue l'étape préliminaire pour le processus d'évaluation, ainsi que le choix approprié des méthodes d'évaluation. Durant cette recherche, on a pensé à modéliser non pas seulement la composante physique d'interaction homme machine qui peut être obtenue en appliquant la loi de Fitts, mais plutôt d'aller encore plus loin en intégrant et modélisant les composantes perceptuelle et cognitive impliquées lors de l'interaction homme machine. Ces deux composantes ne peuvent être modélisées qu'avec des modèles et des méthodes dites cognitives. L'intégration de toutes ces composantes dans l'interaction homme machine justifie le choix des méthodes adoptées dans notre recherche.

La suite de notre recherche s'est intéressée aux travaux d'analyses et de modélisation. Après avoir introduit les concepts théoriques de chaque méthode utilisée, ces concepts ont été mis en œuvre selon les caractéristiques de chaque méthode, en commençant par la paire de

méthodes ACT-R et la loi de Fitts. Dans ce premier travail de modélisation, la loi de Fitts a été utilisée pour supporter et valider les résultats obtenus par le modèle d'ACT-R.

Afin de valider les performances du premier modèle cognitif d'ACT-R, on a opté pour un autre modèle cognitif basé sur le modèle GOMS très utilisé dans la littérature. En fin, Les résultats obtenus par les modèles ont été comparés avec ceux obtenus par une étude expérimentale menée au sein du laboratoire DOMUS avec des personnes saines. Les résultats obtenus ont été très satisfaisants, plus spécifiquement ceux obtenus par les modèles cognitifs qui donnent des meilleures prédictions sur le temps d'exécution des tâches.

On peut ainsi conclure que, si la mise en œuvre de cette évaluation analytique illustre la consistance et la performance des modèles développés ainsi que le réalisme de l'approche retenue, ceci ne constitue qu'une première étape du processus très complexe de réalisation d'activités dans un environnement intelligent. Toutefois, ce travail constitue une base solide non négligeable pour le reste du processus d'évaluation.

Au final, les modèles développés restent à être pleinement améliorés, dans le sens d'inclure plusieurs aspects de la cognition humaine d'un côté, et le dynamisme dans l'exécution de la tâche par l'association de l'environnement de l'utilisateur. Il aurait été intéressant de faire notre évaluation sur toute l'activité de cuisine en modélisant les erreurs cognitives possibles qui pourront être commises tout au long du processus de réalisation de la tâche. Parallèlement à ça, et vu que l'utilisateur interagit avec son environnement pour réaliser sa tâche, il aurait été très intéressant d'ajouter un modèle de l'environnement qui simule les actions et déplacements de l'utilisateur dans l'environnement intelligent.

Bibliographie

- ACM. (1992). *ACM SIGCHI curricula for human-computer interaction*. ACM Report. New York, NY, USA, p. 1-162.
- Anderson, J.R. (1993). *Rules of the Mind*. Lawrence Erlbaum Associates édition, Hillsdale, NJ, 336 p.
- Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C. et Qin, Y. (2004). An Integrated Theory of the Mind. *Psychological Review*, vol. 111, 136-1060.
- Baecker, R.M. (1969). *Interactive Computer-Mediated Animation*. PHD Thesis, Massachusetts Institute of Technology.
- Bajolle, L. (2002). *Amélioration, optimisation et humanisation de la médecine de ville par l'usage de l'internet et des nouvelles technologies (étude de faisabilité, mise en pratique médicale et sociale de l'habitat intelligent pour la santé)*. PHD Thesis, Université de Joseph Fourier de Grenoble, France.
- Bothell, D. (2004). *ACT-R 6.0 Reference Manual*. Technical Report, Carnegie Mellon University, p. 1-255.
- Bovair, S., Kieras, D.E. et Polson, P.G. (1990). The Acquisition and Performance of Text-Editing Skill: A Cognitive Complexity Analysis. *Human-Computer Interaction*, vol. 5, n° 1, p. 1-48.
- Byrne, M.D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. *International Journal of Human-Computer Studies*, vol. 55, p. 41-84.
- Carlson, W. (2003). *A Critical History of Computer Graphics and Animation, Section 17: Virtual Reality*. Technical Report, Ohio State University.
- Davies, D.K., Stock, S.E. et Wehmeyer, M.L. (2001). Enhancing Independent Internet Access for Individuals with Mental Retardation through Use of a Specialized Web Browser: A Pilot Study. *Education and Training in Mental Retardation and Developmental Disabilities*, vol. 36, p. 107-113.
- Direction de l'amélioration de l'exercice. (2000). *La Télémédecine*. Rapport du centre de recherche pour l'étude et l'observation des conditions de vie, France, p. 1-16.

- Fitts, P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, vol. 47, n° 6, p. 381-391.
- Grant, S. (1996). Developing cognitive architecture for modelling and simulations of cognition and error in complex tasks. *Paper presented at a meeting of the RoHMI project, Valenciennes*, (unpublished).
- ISO. (1998). *Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability*. International Organization for Standardization 9241-11, p. 1-16.
- Istrate, D.M. (2003). *Détection et reconnaissance des sons par la surveillance médicale*. PHD Thesis, Institut National Polytechnique de Grenoble, France.
- Just, M.A., Carpenter, P.A. et Keller, T.A. (1996). The capacity theory of comprehension: new frontiers of evidence and arguments. *Psychological Review*, vol. 103, n° 4, p. 773-780.
- Kaye, J. et Sengers, P. (2007). The Evolution of Evaluation. In *Proceedings of the alt.chi, ACM SIGCHI*.
- Kieras, D.E. (2001). *Using the Keystroke-Level Model to Estimate Execution Times*. Technical Report, University of Michigan, p. 1-11.
- Kieras, D.E. (2003). GOMS models for task analysis. *Handbook of task analysis for human-computer interaction*, Lawrence Erlbaum Associates, p. 83-116.
- Kieras, D.E. et Meyer, D. (1997). An Overview of the EPIC Architecture for Cognition and Performance With Application to Human-Computer Interaction. *Human-Computer Interaction*, vol. 12, n° 4, p. 391-438.
- Kieras, D. et Polson, P.G. (1999). An approach to the formal analysis of user complexity. *International Journal of Human-Computer Studies*, vol. 51, n° 2, p. 405-434.
- Kitajima, M. et Polson, P.G. (1997). A comprehension-based model of exploration. *Human-Computer Interaction*, vol. 12, n° 4, p. 345-389.
- Lussier-Desrochers, D., Lachapelle, Y., Pigot, H. et Bauchet, J. (2007). Des Habitats Intelligents Pour Promouvoir L'autodétermination et L'inclusion Sociale. *Revue Francophone De La Déficience Intellectuelle*, vol. 18, p. 52-63.
- Weiser, M. (1991). The computer for the 21st century. *Scientific American*, p. 94-104.
- Myers, B.A. (1998). A Brief History of Human Computer Interaction Technology. *ACM interactions*, vol. 5, n° 2, p. 44-54.

- Myers, B.A. et Rosson, M.B. (1992). Survey on user interface programming. *In Proceedings of the SIGCHI conference on Human factors in computing systems* (p. 195-202), ACM Press.
- Newell, A. (1990). *Unified Theories of Cognition*. Cambridge (Mass), Harvard University Press, 549 p. (Collection The William James lectures).
- Nielsen, J. (1993). *Usability Engineering*. Boston, MA, Academic Press, 362 p.
- Nielsen, J. et Molich, R. (1990). Heuristic evaluation of user interfaces. *In Proceedings of the SIGCHI conference on Human factors in computing systems* (p. 249-256), ACM New York, NY, USA.
- Pigot, H., Giroux, S., Mabillean, P. et Bouchard, F. (2007). L'assistance cognitive dans les habitats intelligents pour favoriser le maintien à domicile. *Les Publications du CRIR, PUB-003 -Défis technologiques*, p. 14-25.
- Radio-Canada, Québec. (2002). *Avantages de la télémédecine*.
<http://www.radiocanada.ca/regions/quebec/nouvelles/2002/archives/index.asp?val=3759>
- Radio des Nations Unies. (2008). *Journée des personnes âgées : promouvoir leurs droits et leur participation pleine et entière*.
<http://www.unmultimedia.org/radio/french/detail/6487.html>
- Ravden, S. et Johnson, G. (1989). *Evaluating usability of human-computer interfaces: a practical method*. New York, NY, USA, Halsted Press.
- Réalités Canadiennes. (2008). *Réalité Canadienne : Vieillesse de la Population Canadienne*. Report of the Centre for Canadian Studies.
- Rivière, J. (2007). Locomotion autonome et cognition spatiale : le paradoxe de l'amyotrophie spinale infantile. *Archives de Pédiatrie*, vol. 14, p. 279-284.
- Senach, B. (1990). *Évaluation ergonomique des interfaces homme-machine: Une revue de la littérature*. Technical Report, INRIA, France, n.1180, p. 1-68.
- Simon, M. et Fronteau, A. (1996). *Les conditions du maintien à domicile des personnes âgées dépendantes*. Rapport du centre de recherche pour l'étude et l'observation des conditions de vie, France.
- Statistique Canada. (2006). *Rapport sur la santé : Utilisation des services de santé par les personnes âgées*.
<http://www.statcan.gc.ca/daily-quotidien/060207/dq060207a-fra.htm>

Société Alzheimer. (2006). *Médias, Communiqués et dossiers de presse. Dossier de presse—100 ans de découvertes.*

<http://www.alzheimer.ca/french/media/100years06-adfacts.htm>

Sutherland, I.E. (1963). SketchPad: A Man-Machine Graphical Communication System. *In Proceedings of the AFIPS Spring Joint Computer Conference*, p. 329-346.

Weiser, M. (1993). Some Computer Science Problems in Ubiquitous Computing. *Communications of the ACM* (p. 74-84).