

Post-It haptiques par codes-barres

Rapport de stage

Thomas Pietrzak

Encadré par le Dr. Benoît Martin



Laboratoire d'Informatique
Théorique et Appliquée



Jury

Pr. Dominique Méry
Pr. Didier Galmiche
Pr. Noëlle Carbonell

29 Juin 2005

Remerciements

Je tiens particulièrement à remercier mon tuteur Benoît Martin qui a su me diriger tout au long de mon stage. Il m'a fait découvrir l'haptique, et m'a présenté quelques uns des plus grands spécialistes européens du domaine par l'intermédiaire de deux meetings du projet européen MICOLE à Metz et à Glasgow. Je remercie les autres membres du thème IHM Isabelle Pecci et Kamel Chelghoum pour leurs conseils et les discussions que j'ai pu avoir avec eux. Je suis reconnaissant envers mon collègue Nicolas Noble pour m'avoir fait profiter de ses qualités techniques et de sa bonne humeur. Je dois le féliciter ainsi que mon frère de DEA Serge Gaspers pour avoir supporté de travailler dans le même bureau que moi. Enfin je remercie infiniment Stephen Brewster pour m'avoir consacré deux journées de son temps lors de mon séjour à Glasgow.

Table des matières

1	État de l'art	1
1.1	Périphériques	1
1.1.1	Les périphériques à retour tactile	1
1.1.2	Les périphériques à retour d'effort	3
1.2	Applications	6
1.2.1	Guidage	6
1.2.2	Palpation	7
1.2.3	Localisation	9
1.3	Affichage d'information	10
1.3.1	Pourquoi ?	10
1.3.2	Existant	11
2	PICOB	13
2.1	Définitions	13
2.2	Repérage	14
2.3	Déclenchement et arrêt	15
2.4	Lecture	15
2.4.1	Codage	15
2.4.2	Représentation	16
2.4.3	Mode de lecture	17
3	Expérimentations	19
3.1	Déroulement et dispositif	20
3.2	Expérience 1	20
3.3	Expérience 2	21
3.4	Expérience 3	22
3.5	Synthèse	24
4	Circuits électriques	25
4.1	Visuel	26
4.2	Noeuds	26
4.3	Arcs	27
A	Détails sur les circuits électriques	31
A.1	Spécifications	31
A.2	Principes utilisés	31
A.3	Améliorations futures	32

Table des figures

1.1	Principe utilisé pour faire sentir un objet	2
1.2	Souris IFeel et Joypad	2
1.3	Gant Teletact	2
1.4	Souris VTplayer	3
1.5	Optacon et Tongue Display Unit	3
1.6	PHANToM	4
1.7	Pantograph	4
1.8	Souris à retour d'effort	4
1.9	PHANToM	5
1.10	Montages à base de PHANToM	6
1.11	Prototype de Comtouch	8
2.1	Mode de lecture non-guidé	17
2.2	Mode de lecture semi-guidé	18
2.3	Mode de lecture guidé	18
3.1	Impulsions dans six directions différentes	19
3.2	Principe utilisé pour guider l'utilisateur en aller-retour	20
3.3	Amplitudes à progression linéaire	22
3.4	Amplitudes à progression linéaire et exponentielle	23
4.1	Schéma de circuit électrique simple	25
4.2	Impulsions dans un noeud	26
4.3	Composant sur une ligne	27
4.4	Représentation haptique d'un circuit électrique	27
A.1	Zone d'attraction d'une ligne	32
A.2	Zone d'attraction d'un noeud	32

Liste des tableaux

3.1	Expérience 1 : erreurs en direction	21
3.2	Expérience 2 : erreurs en amplitude	22
3.3	Expérience 3, série 1 : erreurs en amplitude	23
3.4	Expérience 3, série 2 : erreurs en amplitude	23
3.5	Expérience 3, série 3 : erreurs en amplitude	24

Introduction

Mon stage s'est déroulé au LITA (Laboratoire d'Informatique Théorique et Appliquée) à Metz. J'ai été intégré à l'équipe CMI (Calculs, Modèles et Interface) qui compte cinq thèmes. Mon travail se situe dans le thème IHM (Interface Homme-Machine). Le but était d'étudier les possibilités de l'haptique pour représenter des informations. Tan [Tan00] explique brièvement ce qu'est l'haptique : de manière générale il s'agit du sens du toucher, qui se divise en deux catégories. La première est le retour tactile, qui est ressenti par la peau : c'est grâce à ce sens que l'on peut lire le braille par exemple. Cette catégorie regroupe autant les sensations de pression, de contact, de texture, de volume, que celles de température ou de rugosité. L'autre catégorie est le retour d'effort (appelé aussi kinesthésique) qui est ressenti par les muscles : c'est ce que l'on sent en tenant une pomme dans la main : on ressent des forces qui nous permettent de déduire des informations comme la forme. En déplaçant la main on peut sentir d'autres forces comme le frottement. Certaines forces sont la conséquence d'un effet tactile. Par exemple une rugosité, qui est due à des aspérités sur une surface, va entraîner un frottement qui est une force de résistance au mouvement. Les deux aspects sont donc extrêmement liés. Le sens du toucher permet de recueillir des informations topologiques locales pour pouvoir appréhender l'environnement avant même de l'avoir visualisé. D'après Robles-de-la-Torre et Hayward [RDLTH01], *dans les perceptions des formes par le toucher, les forces de résistance comptent plus que les déplacements de la main*. Donc on reconnaît plus facilement une forme en tenant un objet qu'en le palpant. Dans le cadre de la déficience visuelle le but est naturellement de remplacer le support visuel par un support haptique, éventuellement en synergie avec le support auditif. On voit apparaître la notion de multimodalité, c'est à dire l'utilisation de plusieurs modalités dans les interfaces. Selon le modèle CASE ceci peut se faire de plusieurs manières :

- Concurrente : plusieurs tâches sont effectuées en même temps et n'interfèrent pas entre elles
- Alternée : une tâche utilise plusieurs modalités mais l'une après l'autre
- Synergique : une tâche utilise plusieurs modalités mais en même temps cette fois
- Exclusive : une seule tâche à la fois est effectuée et avec une seule modalité

Les premières interfaces étaient strictement visuelles, ensuite est venu l'auditif : il est temps de doter les interfaces d'une dimension haptique. Le but est de permettre de proposer plus d'interactions avec l'utilisateur sans toutefois que celui-ci soit perdu : au contraire l'objectif est de l'assister dans ses tâches quotidiennes.

Par exemple Chu [[Chu02b](#), [Chu02a](#)] propose d'utiliser l'haptique pour la création musicale. En effet le retour haptique est important lorsqu'on joue sur un vrai instrument alors pourquoi pas dans l'édition sonore? Par exemple les auteurs de l'article envisagent une expérience où l'utilisateur doit trouver le début d'une séquence sonore, ou un index dans un son. Le but est de savoir si l'haptique peut aider le visuel et l'auditif et non de savoir s'il peut le remplacer.

Les membres du thème IHM du LITA travaille actuellement au sein du projet MICOLE (IST-2003-511592) qui a pour but de créer un environnement multimodal collaboratif d'apprentissage pour enfants déficients visuel. Dans le cadre de ce projet un programme permettant de parcourir des circuits électriques doit être réalisé. Ce stage a donc permis de créer une technique d'interaction haptique permettant de donner des informations utiles au parcours de ces circuits électriques et de créer un prototype d'un tel programme.

Le premier chapitre présente un état de l'art, tant sur l'haptique que sur l'affichage d'information. Le deuxième chapitre décrit la technique d'interaction haptique que l'on nomme PICOB (pour Post-It haptiques par codes-barres). Dans le troisième chapitre vous trouverez les expérimentations qui ont été conduites sur PICOB. Enfin le quatrième chapitre donne une brève description de l'application de circuits électriques dans laquelle est décrite l'intégration de PICOB.

Chapitre 1

État de l'art

Dans le domaine de l'Interaction Homme-Machine (IHM) l'haptique est naturellement étudiée à travers les périphériques haptiques d'entrée/sortie. Tout d'abord quelques périphériques haptiques seront présentés. Ensuite quelques applications actuelles de l'haptique seront exhibées. Et enfin le sujet qui nous intéresse ici sera introduit : l'affichage d'informations par un périphérique haptique.

1.1 Périphériques

Il existe différents périphériques permettant de créer un retour haptique. On peut principalement les classer dans deux catégories associées aux deux principales catégories haptiques : le tactile et le retour d'effort.

Hayward et al. [HACH⁺04] ont fait une liste des périphériques haptiques les plus répandus et étudié leurs possibilités, que ce soit au niveau technique ou au niveau des applications possibles. Outre les souris, trackballs, claviers et joysticks on peut remarquer les bras articulés et les exosquelettes. Ce sont des périphériques d'entrée et de sortie.

Le fait d'avoir des périphériques haptiques permet d'envisager des interfaces complètement haptiques. De Boeck et al. [DBRC⁺04] ont conçu un framework permettant de faire des menus, des boutons etc. haptiques. Ruspini et al. [RKK97a, RKK97b] ont développé une librairie haptique (HL) indépendante de tout périphérique. Le principe utilisé pour faire ressentir des objets est celui qui est implémenté dans toutes les bibliothèques pour le retour d'effort (figure 1.1) : on compare la position « théorique » du pointeur à sa projection sur la surface en contact (proxy). Grâce à ça on peut calculer la force à appliquer sur le périphérique pour faire sentir un objet.

Voici quelques périphériques tactiles et à retour d'effort. Ceci n'est bien évidemment pas une liste exhaustive mais un aperçu de ce qui existe et est utilisé.

1.1.1 Les périphériques à retour tactile

Parmi les périphériques haptiques ceux à retour tactile sont les plus utilisés, notamment grâce à leur coût raisonnable. On utilise généralement l'aspect vibratoire car il est facile à réaliser : il suffit d'un ou plusieurs moteurs à cou-

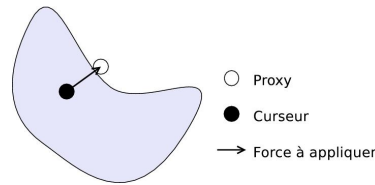


FIG. 1.1 – Principe utilisé pour faire sentir un objet

rant continu pour émettre des vibrations. Dans ce créneau on retrouve la souris IFeel de Logitech (un moteur, figure 1.2) et les joy pads pour les consoles de jeux (deux moteurs). Par exemple, grâce à ce type de périphériques des non voyants peuvent jouer à certains jeux comme des jeux de combat car il peuvent ressentir les coups qu'ils donnent et les coups qu'ils reçoivent. Un autre périphérique vibratoire courant est le vibreur sur les téléphones portable. Cela permet de signaler un appel dans un environnement où l'on ne peut pas utiliser de sonnerie (cours, réunion, bibliothèque, environnement bruyant, etc.) ou même lorsque l'utilisateur ne veut pas utiliser de sonnerie.



FIG. 1.2 – Souris IFeel et Joypad

D'autres périphériques jouent sur l'aspect pression du tactile. Par exemple certains utilisent de petits ballonnets qui, en se gonflant, exercent une pression sur le bout des doigts afin de donner le retour tactile. C'est le cas des gants Teletact de l'Advanced Robotics Research Ltd (figure 1.3).

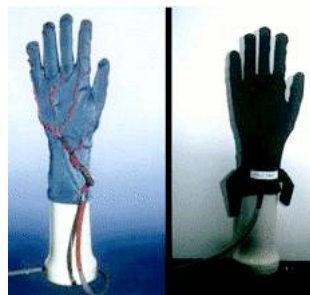


FIG. 1.3 – Gant Teletact

On trouve aussi des périphériques dédiés à la technique du braille comme la souris VTPlayer (figure 1.4) qui est une souris conventionnelle sur laquelle il y a des cellules braille. Elle permet donc de faire sentir des pressions sur la pulpe des doigts. Des équipes de recherche travaillent actuellement avec ce périphérique pour faire ressentir des graphiques et des cartes géographiques.



FIG. 1.4 – Souris VTplayer

Un autre périphérique utilisant les pressions est l'Optacon (figure 1.5 à gauche) : il s'agit d'un périphérique constitué d'une mini-caméra et d'une petite zone qui fait des pressions sur le doigt afin de ressentir ce que filme la caméra. Cet appareil permet donc de faire ressentir des formes. Il permet de lire des lettres conventionnelles mais encore faut-il qu'un non-voyant les « comprenne ».

Une autre technique employée est de placer une matrice d'électrodes sur le corps. Les électrodes délivrent des petites décharges électriques pour exciter les nerfs sensitifs. Le Tongue Display Unit (figure 1.5 à droite) est un périphérique utilisant cette technique. L'idée est que la langue est une des zones du corps qui contient le plus de nerfs sensitifs. Des chercheurs ont donc naturellement essayé d'utiliser cet organe pour un affichage tactile.



FIG. 1.5 – A gauche un Optacon et à droite un Tongue Display Unit

1.1.2 Les périphériques à retour d'effort

Les périphériques à retour d'effort peuvent se classer par degrés de liberté. Les plus simples en ont qu'un seul, comme les volants à retour d'effort (figure 1.6 à gauche). Ils sont utilisés dans certains jeux vidéo et dans des simulations de conduite. Les Joysticks (figure 1.6 à droite) ont deux ou trois degrés de liberté (deux en translation et parfois un en rotation) avec retour d'effort sur les deux degrés en translation. Ce type de périphérique est utilisé dans les jeux et les simulateurs de vol. Ils ont tous les deux un coût raisonnable et sont faciles à utiliser car ils sont destinés au grand public.

Le Pantographe (figure 1.7) est un périphérique de pointage haptique 2D sur une surface de $10\text{cm} \times 16\text{cm}$. Son principe est expliqué par Ramstein et Hayward [RH94]. L'exemple donné est le bouton : pour donner une illusion



FIG. 1.6 – volant et joystick à retour d'effort

de passer sur un bouton il suffit de freiner quand on monte sur le bouton, et accélérer quand on en descend. Plus tard Ramstein [Ram96] a combiné un Pantograph avec un Optacon pour y ajouter le sens tactile. Le but de l'auteur est de proposer un moyen de lire du braille en se déplaçant simultanément dans le texte. Le Pantograph est utilisé pour contrôler la position et l'Optacon pour la lecture tactile. Dans la même idée Wagner et al. [WLD04] ont conçu un volumineux périphérique qui permet de sentir des formes avec le bout du doigt, donc qui permet de sentir des pressions. Le périphérique se déplace sur des roulettes et le mouvement est capté par un capteur optique de souris, ainsi on peut palper des surfaces.



FIG. 1.7 – Pantograph

Les souris Wingman Force Feedback et FEELit (figure 1.8) sont en fait des souris montées sur un Pantograph. C'est donc le même principe qui est utilisé. Leur avantage par rapport aux souris conventionnelles est qu'elles fonctionnent en absolu. Ceci permet à des non-voyants de se repérer et donc d'utiliser une souris comme périphérique de pointage. Malheureusement ces périphériques ne sont plus fabriqués.



FIG. 1.8 – Wingman Force Feedback et FEELit

Le PHANToM (figure 1.9) est un périphérique de pointage en 3D inventé par Massie et Salisbury [MS94]. Il a six degrés de liberté (trois en translation et trois en rotation) avec retour d'effort sur trois ou six degrés de liberté selon le modèle. Sur certains modèles l'utilisateur tient en main un stylet qui est accroché au bout d'un bras articulé. Sur d'autres modèles une sorte de dé à coudre est fixé au bout de l'appareil et l'utilisateur doit l'accrocher au bout du doigt. Son succès est dû à un bon rapport satisfaction/prix. Il peut être programmé avec plusieurs API parmi lesquelles GHOST, Reachin et OpenHaptics.



FIG. 1.9 – PHANToM version Desktop et Omni

Des montages ont été faits avec le PHANToM (figure 1.10 à gauche) : l'Haptic Workbench est constitué d'un PHANToM et d'un dispositif stéréographique (avec un écran et des lunettes). Le but est de plonger l'utilisateur dans un environnement en trois dimensions. Une technique que l'on retrouve dans plusieurs études est un assemblage de trois PHANToMs : un est attaché à l'index, un au pouce et un au majeur. Ceci est possible avec la version 1.0 de ce périphérique car on peut soit l'attacher au bout du doigt soit utiliser le stylet. Cet assemblage permet de compenser une lacune du PHANToM à trois degrés de liberté : c'est un périphérique monopoint, c'est-à-dire que le retour d'effort n'est senti qu'à travers un point. Un inconvénient est qu'il est impossible de sentir les angles et donc très difficile d'apprécier les formes. Le fait d'avoir trois PHANToMs permet d'avoir trois points de contact et donc de mieux sentir des formes. Un autre avantage des assemblages de PHANToM est que l'on peut utiliser le pincement avec un PHANToM sur le pouce et un ou plusieurs sur d'autres doigts. Une étude de Michelitsch et al. [MWO⁺04] utilise cette configuration pour examiner la possibilité de faire des boutons et autres commandes qui changent de forme. Cela permet de simplifier les interfaces. Par exemple pour un bouton sur un magnétoscope on utilise trois formes : circulaire, rectangulaire et demi-circulaire. Il peut servir pour l'avance rapide ou de l'image par image. Chaque forme est associée à une action.

Un autre montage plus impressionnant est l'Haptic Workstation (figure 1.10 à droite). Il s'agit de deux bras articulés style PHANToM couplés à des exosquelettes, deux gants pour avoir le retour tactile, plus un casque de réalité virtuelle et un fauteuil. Lemoine et al. [LGV^T04] l'utilisent pour effectuer des simulations diverses. Ils définissent la notion de *médiateurs* qui correspond à une interface de pilotage à distance. Les avantages sont nombreux, notamment on peut disposer les commandes avec des contraintes de dimensions physique et on peut paramétrer l'interface beaucoup plus facilement qu'une interface conventionnelle.



FIG. 1.10 – Haptic Workbench et Haptic Workstation

Un aspect important et intéressant des périphériques à retour d'effort est qu'ils peuvent aussi simuler le sens tactile notamment avec des vibrations. En effet les vibrations pouvant être émises par des moteurs, et le retour d'effort étant réalisé par des moteurs, on peut faire des vibrations avec un périphérique à retour d'effort. Naturellement les périphériques dédiés au vibratoire sont potentiellement plus performants dans ce domaine car ils sont conçus pour ça. De plus les joypads qui ont deux moteurs pourront faire des vibrations qu'un périphérique n'ayant qu'un seul moteur ne pourra pas faire. Cependant dans certains cas les vibrations créées par un périphérique à retour d'effort sont suffisantes.

1.2 Applications

L'haptique est utilisé dans divers domaines comme par exemple le médical, les communications ou encore le militaire. Trois techniques d'interaction sont présentées ci-dessous : le guidage, la palpation et la localisation. Le guidage consiste à attirer ou contraindre l'utilisateur où on le souhaite. La palpation est une technique où l'utilisateur parcourt des formes pour essayer d'en extraire des informations. Enfin la localisation est tout simplement un moyen de se repérer dans l'environnement de travail.

1.2.1 Guidage

Des travaux intéressants ont été menés pour montrer l'utilité de l'haptique dans un contexte de guidage. Pour ce faire Ramstein et al. [RAD96] ont doté une interface d'une intelligence artificielle. Le but est d'apprendre les mouvements classiques des utilisateurs pour pouvoir les guider en anticipant leurs mouvements ultérieurs. Deux techniques sont mises en oeuvre : celle du « Magicien d'Oz » et un réseau de neurones, avec comme périphérique de pointage un Pantograph. La première technique consiste à prendre le contrôle du Pantograph quand l'utilisateur est en difficulté tandis que la seconde méthode consiste à placer une grille de neurones sur l'espace de travail et à analyser les trajectoires de l'utilisateur.

Dennerlein et al. [DMH00] utilisent une souris FEELit Mouse pour tester l'utilité de l'haptique pour diriger un curseur. Les auteurs ont fait des tests où des utilisateurs doivent traverser des tunnels de 150 à 550 pixels de long et 15 à 50 pixels de large, sans sortir du tunnel entre les extrémités. L'aide

haptique consiste en une attraction vers le centre du tunnel lorsqu'on est à proximité de celui-ci. Sans le retour d'effort les utilisateurs mettent entre 0,5 et 3 secondes selon la difficulté alors qu'avec le retour d'effort ils mettent entre 0,2 et 1 secondes.

Sjöström [Sjö01] donne des conseils pour une interface haptique destinée à des déficients visuels et propose d'ajouter des éléments par rapport à la réalité, comme par exemple aimanter un objet pour le situer plus facilement.

Nesbitt et al. [NGO01] ont fait une étude pour savoir si des informations multisensorielles permettent une meilleure interprétation d'un environnement virtuel. Pour cela les auteurs ont utilisé l'Haptic Workbench (figure 1.10) dans le but d'interpréter des données mathématiques sur un haut fourneau en fonctionnement. Il apparaît que des concepts simples comme les textures, surfaces ou vibrations, sont facilement perçues par les utilisateurs mais que des concepts plus abstraits comme le flux du métal en fusion par exemple n'est pas très bien ressenti et interprété.

Williams et al. [WM03] introduisent de la notion d'*inverted damping*. C'est l'inverse d'un effet de viscosité : plus on va vite et moins ça accroche. Des tests ont été faits avec trois PHANTOMs. Le but était d'attraper des informations disposées dans l'espace de travail. Il apparaît qu'avec l'*inverted damping* les utilisateurs s'acquittent de leur tâche plus rapidement que sans. Ceci vient du fait que l'utilisateur peut aller rapidement vers la zone approximative de la donnée car en déplacement rapide la force est faible. Ensuite une fois dans le voisinage de la donnée l'utilisateur fait des plus petits mouvements donc le système exerce une force plus élevée, ce qui permet de faire des mouvements plus précis. L'idée de guidage est indirecte car l'utilisateur se guide lui même en quelque sorte.

Dans le milieu médical Williams et al. [WSCN04] ont conçu un système haptique utilisant le PHANTOM pour apprendre la palpation à des étudiants ostéopathes à l'aide d'un guidage haptique. Deux expériences sont menées, qui consistent à suivre le mouvement d'un expert lors de la palpation d'une colonne vertébrale. Dans la première le mouvement de l'expert est enregistré et l'étudiant le « rejoue » sans avoir de retour haptique de la palpation, il est juste guidé haptiquement : le but est d'apprendre la trajectoire. Dans la deuxième l'étudiant doit suivre un curseur visuel qui représente le chemin enregistré par l'expert. Il ressent ainsi à peu près ce que ressentait l'expert. Deux bancs de tests ont été réalisés : un banc avec juste la deuxième expérience, un avec la première puis la deuxième. Il se trouve que ceux qui ont fait les deux se trompent moins dans la deuxième que les autres. On ne peut pas entièrement attribuer ce résultat au fait que les sujets avaient plus d'expérience sur le périphérique car les résultats de la deuxième expérience pour les utilisateurs qui n'avaient pas fait la première expérience sont tous au dessus de ceux qui ont fait la première. Il y a notamment une différence significative entre les derniers essais de ceux qui n'avaient pas fait la première expérience et les premiers de ceux qui ont fait la première expérience. Cette étude montre donc deux techniques intéressantes : le guidage et la palpation.

1.2.2 Palpation

D'autres études ont été menées pour comprendre avec quelle efficacité un humain perçoit des formes avec des périphériques haptiques. Jansson et al. [JIs01],

[JL02] ont étudié avec quelle efficacité un humain reconnaît des formes à l'aide d'un PHANToM. Dans un premier temps une étude a été faite avec des formes géométriques de dimensions diverses comme des sphères ou des cylindres. Dans un deuxième temps une expérience sur la reconnaissance de formes du visage a été menée (bouche, nez etc.) avec des visages à morphologies diverses. La proportion de réponses correctes est toujours supérieure à 50% et monte jusqu'à 80% quand la complexité est faible, quand au temps d'exploration de la scène il varie entre 45 et 80 secondes.

Yu et al. [YRB01] ont étudié la perception de graphiques sous forme haptique avec un PHANToM. L'expérience proposée consiste à représenter une courbe montante puis descendante glissante et une droite collante dans un environnement en trois dimensions. L'utilisateur pouvait choisir d'utiliser une grille pour se repérer ou non. La première tâche à accomplir étant de déterminer quelle courbe est la collante et laquelle est la glissante. Seulement deux utilisateurs sur quinze ont fait des fautes. La deuxième tâche consiste quant à elle de trouver le minimum et le maximum de chaque courbe. Tout d'abord il s'avère que la grille n'a pas aidé à mieux réussir le test. Ensuite les résultats montrent que la plupart des utilisateurs ont réussi à accomplir la tâche avec un taux de réussite au dessus des 50%.

Dans le domaine des communications, Chang et al. [COJ+02] ont inventé un périphérique de dialogue haptique : le ComTouch (figure 1.11). C'est un périphérique qui permet de transmettre des pressions à distance. Il fonctionne en complément de la parole. D'après les tests effectués, lors d'une conversation vocale les utilisateurs utilisent le périphérique principalement pour signaler qu'ils vont parler et pour signaler qu'ils ont fini. Un interlocuteur peut signaler qu'il a reçu le message en envoyant des messages d'acquiescement. Les utilisateurs établissent des codes pour oui, non etc. Le but est d'intégrer ceci à un téléphone portable conventionnel pour lui donner une dimension tactile en entrée. Contrairement à ce qui doit être fait dans le cadre de PICOB, ici l'utilisateur est à l'initiative des messages.



FIG. 1.11 – Prototype de Comtouch

Ziat et al. [ZGLS04] utilisent deux cellules braille (matrice 4×4) pour lire des formes en se guidant avec un stylet graphique. L'expérience menée consiste à reconnaître des E majuscule de différentes tailles. Les picots représentent les pixels. Le taux de reconnaissance est assez élevé : 80%.

1.2.3 Localisation

Arsenault et Ware [AW00] ont étudié l'utilité de l'haptique et de la position de l'oeil lors de tâches habituelles. L'expérience proposée consiste à demander à des utilisateurs de toucher des objets dans un environnement en trois dimensions. Deux paramètres sont testés : le retour d'effort et l'efficacité de l'« eyetracking », c'est à dire la possibilité de pouvoir regarder la scène sous un angle différent en bougeant la tête. Il apparaît qu'avec l'eyetracking et avec le retour d'effort la vitesse est optimale alors qu'à l'inverse sans l'eyetracking et le retour d'effort la vitesse est la plus faible. Par contre il y a le moins d'erreurs avec le retour d'effort et sans eyetracking, et le plus d'erreurs sans le retour d'effort et avec l'eyetracking. La vue et le retour d'effort sont donc tous les deux très utiles pour appréhender un environnement en trois dimensions.

Des études concernant la localisation de boutons par le sens tactile ont été faites. D'après Challis [Cha00] les non voyants peuvent apprendre la musique avec le sens auditif par des partitions enregistrées vocalement, ou par le sens haptique avec du braille. Il propose une interface tactile permettant de lire de la musique avec un touchpad Intellikeys d'Intellitools avec des « touches » haptiques pour l'haptique et un synthétiseur vocal et une sortie MIDI pour l'auditif. L'Intellikeys est un grand touchpad qui est généralement utilisé comme un clavier en mettant différentes étiquettes dessus pour donner des configurations de touches différentes. Dans ce cas les étiquettes ont différents effets tactiles selon ce qu'il y a d'imprimé dessus. Il apparaît qu'il y a une confusion dans les zones où il n'y a pas d'informations.

Fukumoto et Sugimura [FS01] ont ajouté un vibreur sur un PDA pour donner un signal lorsque l'utilisateur clique sur un bouton ou un autre objet. Ce dispositif permet de remplacer ou suppléer les bips. Ceci est pratique dans les cas d'utilisation en milieu bruyé par exemple. Les tests montrent que l'efficacité est accrue de 5% en milieu silencieux et de 15% dans un milieu bruyé. Dans cet ordre d'idées, Nashel et al. [NR03] proposent d'ajouter un retour tactile aux téléphones mobiles et PDA qui utilisent des boutons virtuels. Ceci permettrait d'appuyer sur les touches sans regarder le combiné. L'appareil vibre faiblement si le doigt de l'utilisateur se trouve au dessus d'une touche, une vibration un peu plus forte quand le doigt se trouve sur le bord d'une touche pour faire ressentir le bord. Lorsque l'utilisateur appuie sur une touche l'appareil donne une forte impulsion vibratoire pour ressentir la pression sur la touche virtuelle. Différentes fréquences de vibration sont utilisées selon la colonne de la touche sur laquelle se trouve l'utilisateur. La plupart des utilisateurs arrivent à discriminer les fréquences et donc à deviner dans quelle colonne de touches ils se trouvent ou s'ils ne se trouvent pas sur une touche. Par rapport aux travaux de Fukumoto et Sugimura il s'agit là de permettre la localisation des touches à la place de savoir uniquement si elles sont enfoncées.

Une étude de Jeong [Jeo01] nous dit que l'affichage haptique est plus efficace que l'affichage auditif pour la lecture d'une carte géographique. On donne des valeurs à différents territoires sur une carte, cela peut par exemple représenter une population, le nombre de jours ensoleillés dans une année ou tout simplement une numérotation des territoires. Le but peut être de trouver le territoire qui a la plus grande valeur, ou de sentir tout simplement les différences entre plusieurs zones. Avec l'expérience sonore plus la valeur est élevée et plus le son est aigu. Pour l'haptique, plus la valeur est élevée plus la force de la vibration

est élevée. Le périphérique haptique utilisé est la souris Wingman Force Feedback de Logitech. Lors de cette expérience deux données ont été étudiées : la réussite et le temps ajusté de complétion de la tâche. La réussite correspond au nombre de réponses correctes divisé par le nombre de valeurs différentes (10 dans cette expérience). L'ajustement du temps consiste à rajouter 10% de temps supplémentaire pour chaque erreur, cela correspond au temps nécessaire estimé de correction de l'erreur. L'expérience sur l'haptique révèle que les utilisateurs ont une meilleure réussite et mettent moins de temps à accomplir la tâche que dans l'expérience sonore.

1.3 Affichage d'information

Le champ d'investigations pour mon stage est relativement vaste. Au vu des textes que j'ai étudiés plusieurs questions se posent. Tout d'abord pourquoi proposer une interaction haptique ? C'est ce dont nous allons parler en premier lieu. Ensuite nous allons nous intéresser à ce qui se fait déjà en matière d'affichage d'informations et plus particulièrement en ce qui concerne l'auditif, le tactile et le retour d'effort.

1.3.1 Pourquoi ?

Il y a plusieurs buts poursuivis dans cette étude : premièrement il serait intéressant de savoir si l'haptique peut répondre aux besoins de situations où le visuel ou le sonore ne sont pas appropriés. Ceci est le cas pour des interfaces destinées à des déficients visuels ou auditifs, où des interfaces utilisées dans un environnement bruyant par exemple. De plus le but n'est pas seulement de pouvoir faire de nouvelles choses, mais d'être plus rapide et faire moins d'erreurs dans ses tâches quotidiennes. Par exemple Miller et Zeleznik [MZ98] ont eu l'idée de faire des menus, des fenêtres et des icônes haptiques. Ils restreignent les mouvements du PHANToM dans un rectangle de 50mm x 40mm x 2mm pour une résolution de 1280 x 1024. Le but est d'utiliser l'haptique pour éviter les erreurs comme glisser sur un bouton ou une barre de défilement. En effet le PHANToM n'est pas facile au premier abord. Oakley et al. [OMBG00] ont effectué des tests pour savoir avec quelle efficacité un utilisateur peut cliquer sur un bouton avec un PHANToM. La première expérience consiste à disposer plusieurs boutons dans l'espace de travail et de demander à l'utilisateur de cliquer sur un des boutons, choisi par le programme. Dans l'expérience divers effets ont été mis sur les boutons comme de la gravité ou de la texture par exemple, avec un seul effet pour tous les boutons lors de chaque test. Avec les boutons qui ont une propriété de texture les utilisateurs glissent sur le bouton, ce qui provoque beaucoup d'échecs. Par contre avec l'effet de gravité les résultats sont bons. Une autre expérience a été menée avec une liste défilante où les testeurs doivent rechercher et sélectionner des données. Les résultats sont similaires à ceux de l'expérience précédente.

Les techniques de guidages citées plus haut [RAD96, DMH00, AW00, Sjö01, NGO01, WM03] permettent d'éviter un grand nombre d'erreurs. Cela montre l'utilité de l'haptique en matière d'efficacité. On peut donc utiliser cette modalité pour faciliter les interactions et d'en créer de plus rapides et plus efficaces. Le but de PICOB est donc de proposer des telles interactions, d'en implémenter

quelques unes et de les tester.

1.3.2 Existant

Dans un contexte plus large, des études ont déjà été faites pour l'affichage d'informations. On retrouve des études sur le visuel, l'auditif mais très peu sur le retour d'effort.

En ce qui concerne l'auditif on trouve les earcons, créés et analysés par Brewster [BWE93]. Il fait varier plusieurs paramètres comme le rythme, la mélodie et le volume. L'idée est de hiérarchiser l'information : au premier niveau seul le rythme change, au deuxième niveau c'est la mélodie qui change et au troisième c'est le tempo. Pour comparer deux codes il suffit donc de comparer ces trois paramètres et donc de déterminer si les messages sont en rapport ou pas. Les tests proposés demandent à des utilisateurs de reconnaître des icônes et des menus. Dans le cas des menus avec apprentissage il y a 80% de réponses correctes alors qu'il y en a environ 50% dans les autres situations.

Gaver [Gav93] étudie quant à lui les Auditory Icons. Il s'agit de sons associés à des objets ou des actions. Par exemple pour la suppression d'un fichier il joue le son du froissement d'une feuille de papier. La différence avec les travaux de Brewster c'est que ce dernier utilise des notes alors que Gaver utilise des sons.

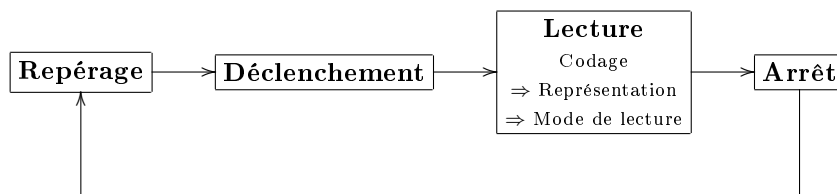
Ramstein [RMD⁺96] décrit son interface haptique : PC-Access. Il utilise un Pantograph pour permettre de parcourir l'environnement avec retour haptique. Cela offre la possibilité de sentir les fenêtres, les menus, les boutons etc. Il relève l'idée d'utiliser des métaphores sonores pour les actions courantes et propose de faire des métaphores haptiques pour augmenter la sémantique de ces icônes auditives. De plus le curseur de pointeur est en position absolue pour pouvoir se repérer sans voir. Plus tard MacLean et al. [MED03, ME03] ont développé des icônes haptiques utilisant un retour d'effort. Ils utilisent un moteur à courant continu qui délivre des forces sur un axe en rotation : le signal envoyé peut se paramétrer en magnitude (amplitude de déplacement), en forme (signal sinusoïdal, carré, etc.) et en fréquence. Les études publiées montrent des tests utilisateurs où les participants devaient classer des icônes utilisant ces paramètres en catégories de leur choix. Il semble que le seul critère unanimement utilisé pour la classification est la fréquence, les autres paramètres ne sont pas discriminants. Cette étude utilisant un périphérique spécifique, il est difficile de reprendre ces résultats.

Des travaux plus approfondis ont été menés par Brewster et Brown [BB04b, BB04a] avec les Tactons. Il s'agit d'icônes tactiles. Le principe est le même que pour les earcons cités plus haut : il s'agit d'une transposition tactile de ces derniers. Cependant il apparaît que la discrimination est moins évidente qu'avec les earcons. Toujours dans la même idée Rovers et van Essen [RvE04] ont créé un framework pour la messagerie instantanée haptique (HIM). En effet il est courant de nos jours d'utiliser la messagerie instantanée pour communiquer sur internet. Les protocoles les plus usités sont IRC, ICQ et MSN. Lors de conversations, il est courant de mettre des smileys (ou émoticônes) qui sont des icônes textuelles ou graphiques permettant d'ajouter une information d'émotion au texte. Ceci permet d'éviter des quiproquos ou tout simplement de partager des émotions comme la joie ou la colère. Grâce à HIM on peut transmettre des symboles haptiques représentant ces émotions lors des conversations. Le système utilisé par les auteurs est constitué d'un périphérique fait maison axé

sur un petit moteur à courant continu. Un gros problème de ce système est que la sémantique des messages haptiques doit être connue des utilisateurs. De plus il faut soit construire un périphérique spécifique à cet usage soit porter cette idée à des périphériques déjà existants.

Chapitre 2

PICOB



PICOB est un projet financé par la CA2M (Communauté d'Agglomération de Metz Métropole), qui s'intègre dans le projet MICOLE. Le but est de créer des systèmes permettant de représenter des informations. Comme on peut le voir sur le schéma ci-dessus il y a quatre étapes importantes pour la gestion de telles informations. La première section va détailler le repérage de l'information. Dans la deuxième section la problématique du déclenchement et de l'arrêt de la lecture est exposée. La troisième section introduit la plus grosse partie du travail effectué dans ce stage : la lecture. Trois notions sont importantes dans cette phase : le codage, la représentation et le mode de lecture.

2.1 Définitions

Avant toute chose je vais préciser ce que j'entend par *Post-It* et par *code-barres*. En effet nous venons de voir que des systèmes d'icônes (notamment haptiques) existent déjà, les notions de *Post-It* et de *code-barres* apportent une originalité intéressante.

Définition 2.1.1 *Un Post-It est une information disposée sur l'environnement de travail de l'utilisateur.*

Ce sont des informations consultatives que l'utilisateur va lire, notamment pour se situer dans le contexte de son espace de travail. Dans le monde physique il s'agit généralement de numéros de téléphone ou de rappels sur des tâches à faire. On pourrait comparer ce type d'informations aux panneaux sur la route qui indiquent autant des directions que les modalités de circulation. Ce sont des informations simples mais utiles et qu'il faut représenter de manière simple.

Définition 2.1.2 *Un code-barres est une information codée lue par un geste.*

Les premiers lecteurs de code-barres étaient des stylos que l'on passait sur le code-barres pour le scanner. Maintenant on place juste le lecteur sur le code-barres et le faisceau se déplace dessus pour le scanner. Notons que les codes-barres ne sont pas tous monodimensionnels : il en existe sous forme de matrice de points en deux dimensions. Nous ne ferons donc pas d'hypothèse a priori sur la représentation du code-barres.

Les notions de Post-It et de code-barres évoquent l'idée d'informations que l'utilisateur choisit de lire ou ne pas lire, c'est à dire qu'il est à l'initiative de la lecture. Avec un Post-It les informations sont « à portée de main » : l'utilisateur les consulte quand il sent qu'il en a la nécessité. Quand aux codes-barres : l'information est codée donc il faut la volonté de lecture pour en saisir le sens. Ce genre de système n'est donc pas adapté à des notifications par exemple car ce n'est pas une information que l'utilisateur lit volontairement. Par contre cela correspond parfaitement aux bulles d'informations. L'intérêt de ces systèmes est de ne pas encombrer l'espace de travail : l'information ne s'affiche que lorsque l'utilisateur le demande.

Des études déjà citées dans la section 1.3.2 ont déjà été menées pour le tactile. Les études concernant l'affichage d'informations à l'aide du retour d'effort sont très limitées [MED03, ME03]. Le périphérique utilisé n'est pas courant, n'est pas un périphérique de pointage et n'a qu'un seul degré de liberté. Les tests effectués ne permettent de dégager que la fréquence du signal comme paramètre discriminant. Le but ici est donc de chercher des valeurs discriminantes sur plusieurs paramètres et avec un périphérique courant.

Le périphérique haptique utilisé est le PHANToM. Nous disposons de deux de ces périphériques : un de type Desktop et un de type Omni.

2.2 Repérage

Nous allons disposer des informations sous forme de code-barres sur l'environnement de travail. Deux cas sont possibles : soit il y a des informations partout, soit elles sont parsemées. Le cas où il y en a partout correspond à des informations portant sur des zones : par exemple si on donne des informations sur ce qui se trouve sous le curseur (barre de titre, de menu, icône, barre défilante, bouton, etc.). Dans ce cas il n'y a pas besoin de repérer les informations vu qu'il y en a partout. Il est plus intéressant d'étudier le cas des informations parsemées. On peut par exemple donner une information sur les boutons et les menus comme le font les bulles d'informations. C'est à ce niveau que se pose le problème du repérage. Plusieurs solutions s'offrent à nous : il est possible d'afficher une sorte d'icône, éventuellement en transparence. Une autre éventualité est d'émettre un son (earcon, etc.) ou une vibration (tacton, etc.). Il se peut que le repérage soit déjà possible avec ce qu'il y a sur l'écran par exemple ; dans ce cas aussi aucun dispositif de repérage supplémentaire n'est nécessaire. Concernant le visuel et le sonore, vu que le projet vise à proposer une interaction haptique, utiliser le visuel ou l'auditif contredit nos motivations.

Des études concernant la localisation d'objets par le retour tactile ont déjà été mentionnées [Cha00, FS01, NR03].

2.3 Déclenchement et arrêt

Une fois que l'utilisateur est « sur » le code-barres il décide de le lire ou de ne pas le lire. Le cas qui nous intéresse est naturellement celui où il décide de lire. Dans ce cas comment va-t-il déclencher la lecture? Plusieurs propositions à ce sujet sont envisageables. La première est la technique utilisée pour les bulles d'informations : la temporisation. L'utilisateur reste quelques secondes à l'endroit où il y a l'information, le système en déduit qu'il veut plus d'informations et passe en mode lecture. Une seconde solution est le clic : le PHANToM étant doté de boutons (un dans la version Desktop et deux dans la version Omni) il est possible de les utiliser pour déclencher la lecture. Une autre solution est d'utiliser le clavier ou un autre périphérique. Chaque solution a ses avantages et ses inconvénients. La temporisation nuit à la rapidité d'interaction, contrairement aux autres solutions. De plus cela induit des risques de lecture non souhaitée, ce qui peut être gênant si la lecture ne peut pas être interrompue par l'utilisateur. Par contre le fait d'utiliser le clic fait que le bouton ne peut plus être utilisé par l'interface (la version Omni du PHANToM ayant deux boutons, elle est moins contraignante sur ce point de vue). Le fait d'utiliser un autre périphérique impose une multimodalité, ce qui complique donc l'interaction et pose des problèmes si le système de base est déjà multimodal. Le choix est donc un compromis à faire, et qui dépend du contexte d'utilisation.

La même question se pose de façon symétrique avec l'arrêt de la lecture. En effet il est possible de requérir une interaction pour l'arrêt de la lecture, sachant que cette interaction n'est pas forcément liée à celle du déclenchement ; cependant les mêmes concepts peuvent être utilisés.

2.4 Lecture

La lecture est la phase la plus intéressante et la plus compliquée du processus. Elle impose de définir plusieurs choses : tout d'abord le codage que l'on souhaite utiliser ; puis la ou les représentations de ce codage que l'on souhaite développer ; et enfin la manière de lire ces représentations avec un périphérique.

Pour bien expliquer la différence entre codage et représentation prenons un exemple : le code morse. Ce codage possède plusieurs représentations : visuellement on utilise des traits et des points et auditivement on utilise soit des bips longs et des bips courts, soit des « ti » et des « ta ».

2.4.1 Codage

L'affichage d'information se fait par le biais d'un codage. On appellera *digit* l'élément atomique d'un *code*. Le *codage* correspond aux règles définies afin d'organiser les digits pour coder les messages.

Exemple : si on désire coder le message « SOS » à l'aide du codage morse on a besoin du code de « S » et du code de « O ». Les digits utilisés pour le codage morse sont *long* et *court*. Le code de « S » est *court court court*, et le code de « O » est *long long long*. Le message « SOS » codé en morse donne donc *court court long long long long court court court*.

On peut utiliser plusieurs méthodes pour créer des codages. Une manière simple de créer un codage consiste à donner un code pour chaque lettre de l'alphabet, comme dans le cas du morse et du braille. Une autre approche consiste à coder des notions ou des métaphores. Pour illustrer ces deux principes imaginons que l'on doive signaler des directions à un utilisateur : nord, sud, est, ouest. Le codage de type alphabet consisterait à faire un code pour chaque lettre : 'n', 'o', 'r', 'd', 's', 'u', 'e' et 't'. Avec ces codes on va former les mots nord, sud, est et ouest. Dans le cas d'un codage par icônes il faudra créer une icône qui symbolise le nord, une pour le sud, une pour l'est et une pour l'ouest. On aura donc besoin de quatre codes contre huit dans le cas précédent. Dans cet exemple le codage par icônes semble plus approprié. Rien ne dit que ce sera le cas dans toutes les situations.

2.4.2 Représentation

La représentation d'un digit est appelée un *effet*, et celle d'un code est appelée une *lettre* ; un message est composé d'une ou plusieurs lettres.

Les représentations sont basées sur les effets, et un codage a besoin d'un certain nombre d'effets discriminables. Une représentation est un jeu d'effets au service du codage. Par exemple l'ensemble $\{-, \bullet\}$ est un jeu d'effets visuels pouvant être utilisé pour représenter le code morse. On peut aussi le représenter avec ces jeux d'effets auditifs : $\{bip, biiip\}$ et $\{ti, ta\}$. En règle générale, le morse pourra être représenté par tout jeu de deux effets discriminants. D'autres langages sont plus exigeants. Notre but dans un premier temps est de découvrir des jeux d'effets discriminants les plus vastes possibles pour pouvoir proposer des représentations au plus grand nombre possible de codages. Naturellement il est question de trouver des représentations basées sur l'haptique donc nous allons nous pencher sur les possibilités d'affichage de codes par l'haptique en trouvant des effets haptiques.

Les effets haptiques courants sont la dureté (rigide/mou), le frottement (collant/glissant), le poids (lourd/léger). Il faut trouver des valeurs suffisamment discriminantes pour pouvoir les utiliser dans nos codages. De plus il est important de considérer la taille des échantillons de lecture et le nombre de gestes à faire pour discriminer les effets correctement.

Des études ont été faites pour déterminer quelles types de surfaces sont facilement discriminées lors d'une palpation. Nesbitt [Nes02] a effectué des tests avec un Haptic Workbench (donc un PHANTOM comme périphérique de pointage haptique) sur trois critères pour l'affichage de données abstraites : rugosité, dureté et inertie. L'expérience proposée consiste à classer dix cubes ayant ces propriétés, du plus faible au plus fort. La discrimination n'étant pas assez forte, l'expérience a été modifiée pour n'utiliser plus que cinq cubes. Pour permettre une bonne discrimination, la valeur est doublée entre chaque cube. Les utilisateurs ont mis environ deux minutes pour chaque tâche. Sur dix utilisateurs, un seul s'est trompé avec la rugosité et un autre sur l'inertie alors qu'aucun ne s'est trompé avec la dureté. De plus selon les utilisateurs le plus facile à discriminer est la rugosité. Aucune information sur la taille des cubes n'est donnée.

Wall et Bruster [WB03a, WB03b] ont étudié trois paramètres : la texture, la rigidité et le frottement. Ils ont retenu deux valeurs pour ces paramètres et

le testeur devait sentir avec un PHANToM deux surfaces qui ne diffèrent que par l'un des trois paramètres. Naturellement le but est de deviner lequel. Il apparaît que le paramètre le mieux distinguable est le frottement. Là encore aucune information sur la taille des surfaces de test n'est donnée.

L'étude de Yu et al. déjà mentionnée [YRB01] nous dit que les utilisateurs arrivent à distinguer le collant et le glissant. Ce paramètre peut donc potentiellement être utilisé pour créer un jeu d'effets haptiques.

Une autre technique pouvant être utilisée est de donner des impulsions. Les différents paramètres pouvant être testés sont l'amplitude et la direction. Par exemple avec quatre directions on peut représenter les quatre points cardinaux. C'est cette représentation que nous allons étudier dans les chapitres suivants.

2.4.3 Mode de lecture

Après avoir décidé d'une représentation il faut déterminer comment cette représentation va être lue, notamment avec les contraintes du matériel que cela impose. Comme je l'ai déjà mentionné, le code-barres évoque la notion de lecture par un geste. Dans cette étude la lecture se fait par l'intermédiaire du PHANToM. Je propose d'étudier le mode de lecture à travers la notion de guidage. En effet ceci est une chose naturelle en terme de retour d'effort. L'idée est de guider l'utilisateur dans sa position ou dans son mouvement lors de la lecture. J'en déduis trois modes : le premier est un mode dans lequel l'utilisateur est totalement libre, c'est le mode *non-guidé*. Le second est un mode dans lequel l'utilisateur sera contraint en position sur le code barre tout en pouvant contrôler le mouvement de lecture, c'est le mode *semi-guidé*. Le dernier mode est un mode où le PHANToM guide entièrement l'utilisateur dans la lecture, c'est le mode *guidé*.

Mode non-guidé

Dans ce mode l'utilisateur est entièrement maître de la lecture : il n'est ni contraint en position, ni en mouvement. Le gros avantage de ce mode est que l'utilisateur contrôle entièrement la lecture. Cependant se pose le problème de la localisation du code-barres : soit il suit l'utilisateur quelque soit son mouvement (figure 2.1, à gauche), soit il reste en place (figure 2.1, à droite).

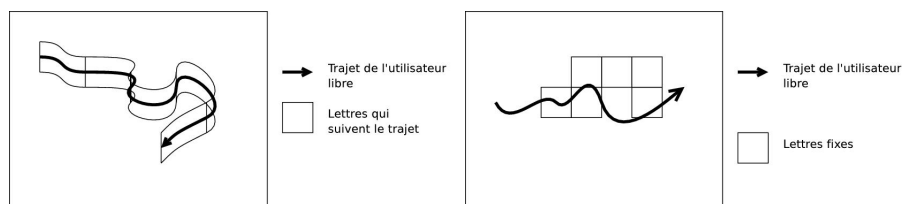


FIG. 2.1 – lectures en mode non-guidé

Pour le premier cas c'est la distance parcourue qui détermine l'effet à lire. Par exemple tous les centimètres parcourus on passe à l'effet suivant. Il faut prévoir le cas où l'utilisateur veut revenir en arrière. Ceci est possible par exemple en détectant l'angle du mouvement. Un angle aigu correspond à un retour en arrière. Une autre manière est de détecter les changements de courbure. Cela

reste à priori relativement complexe à mettre en oeuvre. Les API haptiques utilisent un rafraîchissement de 1000Hz, ce qui poserait des problèmes de mesures d'angles instantanés.

Dans le deuxième cas le code-barres est fixe, et comme l'utilisateur se déplace où il veut il peut très bien sortir du code. Comment interpréter cet événement : un désir d'arrêter la lecture ? une erreur ? Ceci doit être défini au moment de la conception du système. La séquence des lettres n'est pas fixée, et l'utilisateur peut très bien en rater quelques unes. C'est aussi une chose à prendre en compte lors du choix du mode de lecture à la conception.

Mode semi-guidé

Dans ce mode l'utilisateur est contraint en position mais pas en déplacement. Ceci permet de prendre le gros avantage du mode précédent, à savoir le contrôle de la lecture sans en avoir l'inconvénient majeur : risquer de rater un bout du code. En effet l'utilisateur est restreint dans sa position au code-barres, mais par contre il peut se déplacer librement sur celui-ci, notamment avec la possibilité de revenir en arrière (figure 2.2). Le séquençement des lettres est induit par la topologie du code. En donnant trop de liberté à l'utilisateur on se rapproche du mode précédent et on s'expose aux mêmes inconvénients.

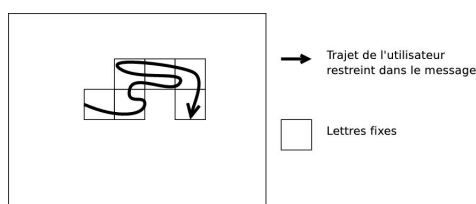


FIG. 2.2 – lecture en mode semi-guidé

Mode guidé

Dans ce mode l'utilisateur est contraint à suivre le mouvement imposé par le PHANToM (donc est aussi contraint en position). Le but est de faire sentir à l'utilisateur les directions ainsi que les effets placés sur le chemin (figure 2.3). L'inconvénient principal de ce mode est qu'il n'est pas adapté pour faire ressentir des effets comme l'inertie par exemple, car l'inertie est une force qui s'oppose au mouvement. Or le guidage consiste à donner un mouvement, donc ajouter une force contraire ne serait juste ressenti comme une décélération.

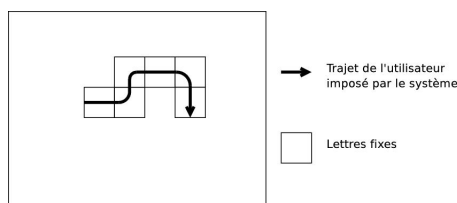


FIG. 2.3 – lecture en mode guidé

Chapitre 3

Expérimentations

Notre but est de trouver des représentations haptiques. Il faut donc trouver des jeux d'effets haptiques discriminables facilement et rapidement. On a décidé de tester un alphabet composé d'impulsions, c'est-à-dire des aller-retours dans une direction. Les paramètres de telles impulsions sont la direction, la hauteur et la durée. Seules les deux premiers ont été testés, le dernier étant fixé. La figure 3.1 représente des impulsions dans les six directions : haut, droite, arrière, bas, gauche et avant. Pour un périphérique à deux degrés de libertés on devra se limiter à quatre directions.

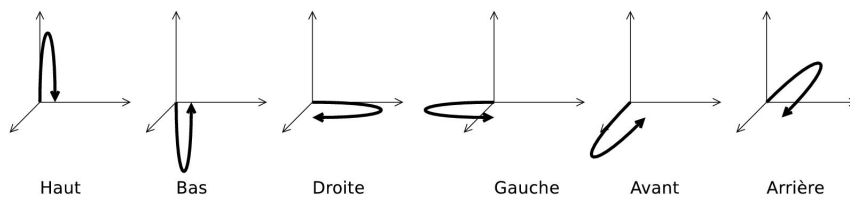


FIG. 3.1 – Impulsions dans six directions différentes.

Le principe utilisé pour réaliser ces impulsions est simple : l'idée est de tirer le curseur (\circ sur la figure 3.2) avec un ressort exerçant une force $\vec{F} = k\Delta_l$ (k : la constante de raideur ; Δ_l : distance entre les deux points) vers une position théorique de l'impulsion (\bullet sur la figure). En effet on ne peut pas contrôler directement la position du PHANToM, on peut juste lui appliquer des forces. Pour pouvoir atteindre une position on doit définir un point de convergence et exercer une force dirigée vers lui pour y déplacer le périphérique. Celui-ci ne fera que vibrer autour de ce point en réalité, et généralement le fait de tenir le stylet en main permet d'éviter que l'appareil n'entre en résonance. Dans le cas contraire il faut amortir le mouvement, en créant une inertie artificielle par exemple.

Les tests décrits ici portent uniquement sur le mode guidé et ont été soumis pour un article court à la conférence IHM 2005 [PMP05]. Les expérimentations sur le mode semi-guidé sont en cours. Trois tests sont proposés : dans le premier il s'agit de reconnaître la direction, dans le second il faut déterminer l'amplitude et dans le troisième il faut trouver les deux. Ces tests n'ont pas pour but d'étudier un apprentissage mais une utilisabilité immédiate.

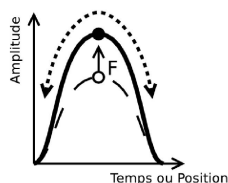


FIG. 3.2 – Principe utilisé pour guider l'utilisateur en aller-retour

Les participants étaient occultés afin de supprimer l'aide visuelle. Chaque expérimentation consistait à faire une ou plusieurs séries tirées aléatoirement et les utilisateurs faisaient tous les mêmes séries afin de n'avantager personne a priori. Le nombre de tests par série a été calculé après des tests préliminaires d'étalonnage afin de faire des séries de cinq minutes. Cet étalonnage a révélé qu'un test prend environ deux secondes pour une impulsion de $250ms$, donc chaque série contient 150 impulsions. Les séries étaient regroupées en sessions, sachant que tous les utilisateurs faisaient les mêmes sessions au rythme de une par jour maximum. Après chaque session un questionnaire était distribué pour recueillir les impressions des participants.

3.1 Déroulement et dispositif

Après avoir expliqué le principe de l'expérience à l'utilisateur, l'expérimentateur déclenche les impulsions une à une. L'utilisateur dit la direction ou l'amplitude qu'il a ressenti et un quart de seconde plus tard l'impulsion suivante est déclenchée. Cette expérience a été réalisée avec un PHANToM Desktop comme périphérique de pointage à retour d'effort. C'est un périphérique à six degrés de liberté (trois en translation et trois en rotation) et à retour d'effort sur les trois axes de translation. Le programme utilisé a été réalisé avec l'API Reachin [rea] en utilisant VRML et Python. Six utilisateurs ont été invités à participer aux tests. L'échantillon était composé de trois enseignants chercheurs en informatique (deux hommes et une femme) et trois étudiants en informatique de sexe masculin. Les chercheurs et un des étudiants sont habitués à manipuler le PHANToM. Les utilisateurs avaient entre 23 et 47 ans.

3.2 Expérience 1

Le but de cette expérience est de déterminer si un utilisateur arrive à reconnaître des impulsions selon les six directions : haut, bas, gauche, droite, avant et arrière (figure 3.1). L'utilisateur devait faire cinq séries de tests avec des amplitudes différentes : $0,4cm$, $0,7cm$, $1cm$, $1,5cm$ et $2,25cm$. Ainsi on peut déterminer si cette reconnaissance est possible avec plusieurs amplitudes.

Résultats et discussions : Les résultats sont rassemblés dans la table 3.1. La première colonne représente l'amplitude de la série en centimètres, et la dernière colonne représente le pourcentage d'erreurs sur la série. Quatre utilisateurs sur les six se sont trompés, et qu'une seule fois sur les 750 tests que totalisaient

les cinq sessions. Deux erreurs ont été faites par des utilisateurs habitués au PHANToM et deux par des utilisateurs non habitués au PHANToM. Avec une erreur sur 750 tests il est clair que cette erreur est due à l'inattention.

Amplitude (cm)	Utilisateur						Erreurs (%)
	1	2	3	4	5	6	
0,4	0	0	0	0	1	1	0,22%
0,7	0	0	0	0	0	0	0,00%
1	0	0	0	0	0	0	0,00%
1,5	0	0	1	0	0	0	0,11%
2,25	0	1	0	0	0	0	0,11%
total	0	1	1	0	1	1	0,09%

TAB. 3.1 – Erreurs de direction dans l'expérience 1

Un seul utilisateur (non habitué au PHANToM) avoue avoir eu parfois des difficultés à sentir la différence entre les impulsions, et trois autres utilisateurs (dont un non habitué au PHANToM) ont eu des hésitations sur certaines directions. Ces hésitations portaient sur les directions haut/avant, haut/arrière, bas/avant et bas/arrière. Cependant d'après les remarques des utilisateurs et les résultats cela ne les empêche pas de discriminer les directions. Par contre deux utilisateurs disent avoir eu l'impression que l'amplitude n'était pas la même dans des directions différentes sur la même série : un utilisateur a eu l'impression que la droite était plus grande que haut et bas, alors qu'un autre utilisateur a eu l'impression que haut, bas, gauche et droit était plus grand que avant et arrière. D'après le sentiment de ces utilisateurs, une explication possible pourrait être la tenue du stylet. Celle-ci diffère selon les utilisateurs et selon l'amplitude.

Trois utilisateurs trouvent que la cinquième amplitude est trop grande mais un d'entre eux évoque la possibilité que ça dérange moins lors d'une lecture de plusieurs impulsions successives. Une amplitude trop grande provoque des effets gênants : perte de la prise en main du stylet et déplacements involontaires dans la scène (dus à des réflexes). Deux utilisateurs suggèrent d'orienter la direction en fonction de l'orientation du stylet. Il ne s'agirait donc plus de directions absolues mais relatives. Il faudrait donc déterminer si les buts de ces deux solutions sont les mêmes.

On retiendra de cette expérience que la discrimination d'impulsions entre $0,4cm$ et $2,25cm$ dans six directions est claire.

3.3 Expérience 2

Cette expérience a pour but de tester la discrimination d'amplitude. Elle utilise trois impulsions d'amplitude différentes : $4mm$, $9,5mm$ et $1,5cm$, la progression est linéaire (figure 3.3). L'utilisateur doit reconnaître celle qui lui est proposée. Cette expérience n'a été réalisée qu'avec une seule direction : vers le haut.

Résultats et discussions : Les erreurs sont représentées dans le tableau 3.2. On constate 6% d'erreurs en moyenne sur six utilisateurs. C'est trop grand pour pouvoir affirmer que c'est discriminable. Les erreurs ne sont pas réparties

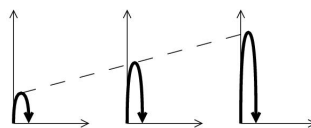


FIG. 3.3 – Trois impulsions d’amplitudes à progression linéaire.

équitablement entre les amplitudes : en moyenne 1% des petites amplitudes provoquent des erreurs contre 7% pour les moyennes et 11% pour les grandes.

Amplitude (cm)	Utilisateur						Erreurs (%)
	1	2	3	4	5	6	
0,4	0	1	0	0	1	1	1 %
0,95	3	2	4	5	2	6	7 %
1,5	3	10	7	10	1	2	11%
total	6	13	11	15	4	9	6 %

TAB. 3.2 – Résultats de l’expérience 2 : chaque ligne correspond à une amplitude proposée, la dernière ligne correspond au total de la série.

Si on regarde les amplitudes ressenties par les utilisateurs, on se rend compte que quand une petite ou une grande amplitude était proposée, les réponses erronées étaient toujours « amplitude moyenne » comme on peut s’y attendre. Par contre le fait marquant concerne les erreurs de reconnaissance des amplitudes moyennes : les réponses étaient toujours « grande amplitude », ce qui fait qu’aucun utilisateur ne s’est trompé quand il disait « petite amplitude ». La discrimination entre la moyenne et la grande amplitude est donc très problématique, ce qui se vérifie dans l’expérience suivante.

3.4 Expérience 3

Dans cette expérience il s’agit de tester la reconnaissance de plusieurs directions et plusieurs amplitudes simultanément. Trois séries ont été effectuées. La première consiste à déterminer si on peut discriminer six directions et deux amplitudes. Les six directions sont toujours les mêmes et les amplitudes sont de $4mm$ et $1,6cm$. Les deux autres séries avaient pour but de tester la discrimination de six directions et trois amplitudes avec respectivement une progression linéaire et une progression exponentielle (figure 3.4). Les amplitudes étaient de $4mm$, $9,5mm$ et $1,5cm$ pour la deuxième série et $4mm$, $8mm$ et $1,6cm$ pour la troisième. La série avec les valeurs linéaires utilise volontairement les mêmes amplitudes que dans l’expérience 2 afin de déterminer si l’utilisation simultanée de plusieurs directions ajoute une difficulté supplémentaire à la discrimination d’amplitude. Le tests sur une série à progression de valeurs exponentielle est motivée par Nesbitt [Nes02] qui suggère de multiplier les valeurs par deux pour la discrimination de paramètres haptiques.

Résultats et discussions : Les erreurs de direction sont toujours minimales : seuls deux utilisateurs ont fait une faute sur les trois séries ; c’est pourquoi les valeurs ne seront pas détaillées. On peut conclure assez facilement que le

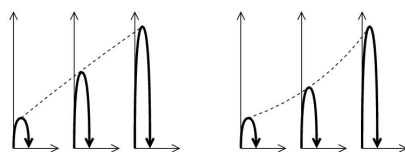


FIG. 3.4 – Trois impulsions d’amplitudes à progression linéaire et trois à progression exponentielle.

fait d’utiliser plusieurs amplitudes en même temps que plusieurs directions ne perturbe pas la discrimination des directions. Nous concentrerons donc l’analyse des résultats sur les erreurs d’amplitude.

Les résultats de la première série sont reportés dans la table 3.3. La moitié des utilisateurs ont fait une erreur sur les 150 impulsions, les autres en n’ont fait aucune. On peut donc affirmer que l’on peut discriminer des impulsions dans les six directions et d’amplitudes $4mm$ et $1,6cm$. Il serait intéressant de rapprocher ces valeurs pour voir à quelle intervalle les erreurs ne sont plus négligeables. Par contre les utilisateurs avaient déjà le sentiment de ne pas pouvoir discriminer une valeur intermédiaire.

Amplitude (cm)	Utilisateur						Erreurs (%)
	1	2	3	4	5	6	
0,4	0	0	1	0	0	0	0,11%
1,6	0	1	0	0	1	0	0,22%
total	0	1	1	0	1	0	0,33%

TAB. 3.3 – Expérience 3, série 1 : erreurs en amplitude. Chaque ligne correspond à une amplitude proposée, la dernière ligne correspond au total de la série.

La seconde série utilise la progression de taille linéaire avec les mêmes valeurs d’amplitude que dans l’expérience 2 ($4mm$, $9.5mm$ et $1.5cm$). Vous pouvez consulter les résultats dans la table 3.4. La moitié des utilisateurs n’ont aucun problème pour reconnaître la petite impulsion, et les autres utilisateurs font peu d’erreurs. En moyenne il y a 3% d’erreurs quand une petite impulsion était proposée. La majorité des erreurs se font sur les deux autres amplitudes : en moyenne 19% des médianes sont mal interprétés et 27% des grandes. C’est environ trois fois plus que dans l’expérience 2. On peut donc affirmer que l’utilisation de plusieurs directions perturbe la discrimination d’amplitude, comme on pouvait s’y attendre.

Amplitude (cm)	Utilisateur						Erreurs (%)
	1	2	3	4	5	6	
0,4	0	1	3	5	0	0	3%
0,95	10	10	19	4	5	2	19%
1,5	9	17	20	18	13	9	27%
total	19	28	42	27	18	11	16%

TAB. 3.4 – Expérience 3, série 2 : erreurs en amplitude. Chaque ligne correspond à une amplitude proposée, la dernière ligne correspond au total de la série.

La troisième série, dont les résultats sont disposés dans la table 3.5, permet

de comparer les valeurs à progression linéaire de la série précédente à des valeurs à progression exponentielle. On constate que pour les petites amplitudes on a toujours 3% mais par contre pour les moyennes et les grandes amplitudes on descend respectivement à 11% et 19%. Ceci est une nette amélioration car de manière globale il y a un tiers d'erreurs en moins, un quart en moins pour les amplitudes moyennes et presque 40% pour les grandes amplitudes. Un seul utilisateur a fait plus d'erreurs que dans la série précédente. Par contre un autre utilisateur n'a fait qu'une erreur par amplitude. Cette amélioration permet d'espérer qu'il est possible de diminuer encore les erreurs en ajustant les valeurs, en augmentant la précision des impulsions et avec un apprentissage. Une discrimination de trois amplitudes dans six directions est donc peut être possible mais pas encore atteinte.

Amplitude (cm)	Utilisateur						Erreurs (%)
	1	2	3	4	5	6	
0,4	1	0	2	1	3	1	3 %
0,8	2	23	8	2	1	1	11%
1,6	7	24	13	5	3	1	19%
total	10	47	23	8	7	3	11%

TAB. 3.5 – Expérience 3, série 3 : erreurs en amplitude. Chaque ligne correspond à une amplitude proposée, la dernière ligne correspond au total de la série.

3.5 Synthèse

L'apprentissage est jugé insuffisant par les utilisateurs ayant fait beaucoup d'erreurs. Cependant le but de cette expérience était justement de tester une utilisabilité immédiate. Il serait toutefois intéressant de tester un apprentissage. Le manque d'apprentissage peut expliquer beaucoup d'erreurs car généralement lorsqu'il y avait beaucoup de fautes l'utilisateur décalait les valeurs. Le fait de ne pas savoir si leur réponse était juste ou pas leur empêchait de se corriger pour ne plus faire de fautes. Ceci est le second vœu des utilisateurs : avoir la bonne réponse une fois qu'ils ont donné la leur. En effet beaucoup d'erreurs ont été faites car les utilisateurs se basaient sur leurs réponses précédentes donc s'ils se trompaient une fois ils avaient tendance à refaire la même erreur. Ceci explique les écarts importants entre différents utilisateurs dans la troisième série de l'expérience 3.

La prochaine série de tests consiste à faire les mêmes expériences mais en mode semi-guidé. La différence principale est que l'on aura deux directions en moins car un axe est utilisé pour le déplacement de lecture.

Des tests plus approfondis pourraient être menés avec le mode guidé. Par exemple on pourrait tester un apprentissage sur la deuxième série de l'expérience 3. Une autre approche intéressante serait de diminuer l'écart de l'expérience 4 jusqu'à avoir des erreurs significatives pour déterminer un seuil de discrimination.

Une fois ces tests sur les effets faits nous pourrions commencer à expérimenter la reconnaissance de mots entiers, donc des suites de lettres. Ceci sera effectué par l'intermédiaire des circuits électriques.

Chapitre 4

Circuits électriques

Une application intéressante des PICOB et du guidage haptique est la réalisation d'un logiciel de manipulation de circuits électriques haptiques. Dans le cadre du projet européen MICOLE j'ai réalisé avec mon collègue Nicolas Noble une maquette d'un tel logiciel.

Ce programme est destiné à des enfants déficients visuel de niveau collègue. La notion de topologie est importante pour permettre à ces utilisateurs d'utiliser cette application. Dans un circuit la notion de composant est naturellement très importante. C'est pourquoi il est nécessaire de les indiquer.

Pour ce faire nous avons décidé de faire des circuits haptiques. Le circuit est magnétique, de sorte que lorsqu'on se déplace à proximité on soit attiré pour se coller dessus. Une fois dans le circuit il faut forcer pour en sortir, ainsi l'utilisateur peut se repérer plus facilement. De plus pour trouver le circuit l'utilisateur a la possibilité de demander au programme de le guider par un simple clic.

La figure 4.1 montre une capture d'écran du logiciel avec un circuit simple. Les détails techniques sur le programme sont exposés dans l'annexe A.

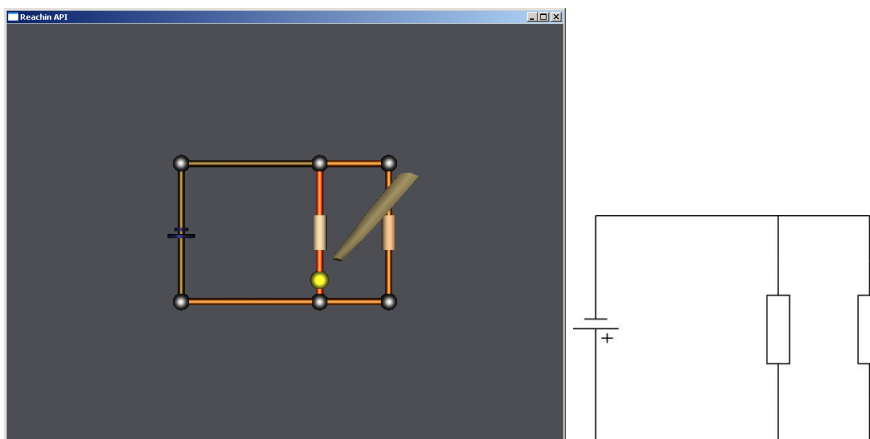


FIG. 4.1 – Exemple de schéma d'un circuit électrique simple

Je vais tout d'abord expliquer la nécessité du visuel. Ensuite la dimension

haptique sera expliquée. La restriction de l'utilisateur dans le circuit est gérée par des noeuds et des arcs. Dans ce programme nous avons naturellement utilisé PICOB pour annoter les circuits. Dans les noeuds il y a des PICOB guidés et dans les arcs des PICOB semi-guidés. Leur utilité est décrite ci dessous.

4.1 Visuel

Ce programme répond à un besoin pour les enfants déficient visuels. Cependant l'usage n'est pas et ne doit pas y être limité. En effet d'une part les enseignants doivent pouvoir aider l'enfant dans sa tâche et notamment voir les arcs déjà parcourus tout comme l'enfant peut les ressentir. D'autre part le programme doit aussi être utilisable par des enfants non déficients car les enfants mal voyants n'ont pas envie d'utiliser des outils qui leurs sont dédiés. En effet ça ne fait que souligner leur différence, ce qui les met mal à l'aise face aux autres enfants. C'est pour ces raisons que le visuel est tout de même important dans ce programme.

En conséquence, comme on peut le constater sur la figure 4.1, les lignes déjà parcourues sont affichées plus foncé. La ligne dans laquelle on se trouve est plus claire que les autres. Quand aux composants, ils ressemblent aux symboles couramment utilisés dans les manuels.

4.2 Noeuds

Les noeuds permettent à l'utilisateur de s'orienter dans le circuit, donc il y en a à chaque intersection. Ils contiennent des PICOB en mode guidé qui indiquent les directions des noeuds adjacents.

Le déclenchement de la lecture se fait en entrant dans le noeud, le PICOB étant lu qu'une seule fois. Par la suite il est possible de déclencher la lecture à nouveau par un clic pour que l'utilisateur n'ait pas besoin de sortir et de ré-entrer. En effet il faut limiter les déplacements pour éviter qu'il se perde.

Le système enregistre les arcs déjà parcourus et donc deux amplitudes sont utilisées pour donner les directions : faible si l'arc a déjà été parcouru et forte s'il ne l'a jamais été. Sur la figure 4.2 on constate que l'utilisateur ressent des petites impulsions vers le haut et le bas, ce qui veut dire qu'il y a des arcs qu'il a déjà parcouru dans ces directions. Par contre il ressent une grande impulsion vers la gauche, ce qui veut dire que dans cette direction se trouve un arc qu'il n'a pas encore visité.

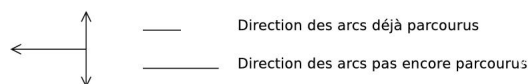


FIG. 4.2 – Impulsions dans un noeud

L'arrêt de la lecture est trivial car il correspond à la fin des impulsions. On pourrait imaginer un système permettant d'arrêter la lecture si l'utilisateur juge qu'il en sait assez pour choisir la direction qu'il compte emprunter.

4.3 Arcs

Les arcs relient les noeuds, et sont le support pour des composants ou de simples fils électriques. Des PICOB en mode semi-guidé sont intégrés dans les arcs afin de faire ressentir une représentation haptique des composants.

Les impulsions sont centrées sur les arcs, leur taille étant fixée par le nombre de lettre et la taille des effets. Le déclenchement et l'arrêt consiste tout simplement à atteindre les extrémités des PICOB en passant dessus. Aucune action spécifique de l'utilisateur n'est donc requise et donc pour lui il ne s'agit que d'une déformation de l'arc (figure 4.3).



FIG. 4.3 – Composant sur une ligne

Comme on est en mode semi-guidé l'utilisateur peut lire les codes dans les deux sens. Nous effectuerons des tests utilisateur qui nous permettront de savoir si on doit utiliser des PICOB symétriques (style palindromes) afin de comprendre le même code dans les deux sens. En effet si le mot est « grand petit », dans l'autre sens on ressent « petit grand ». Est-ce que le sens de parcours nous permet de fixer un repère et nous permet de comprendre « grand petit » ou comprenons nous « petit grand » ?

Un autre problème assez similaire concerne l'orientation des PICOB : peuvent ils être compris de la même façon qu'ils soient horizontaux ou verticaux ? Encore une fois des tests seront effectués pour le déterminer.

La figure 4.4 correspond à la représentation haptique du circuit de la figure 4.1. Il s'agit des limites que l'utilisateur sentira s'il essaye de sortir du circuit. Les résistances sont représentées par trois grandes impulsions et la pile par une petite et une grande impulsion. Cette dernière n'a pas un code symétrique et donc si la reconnaissance n'est pas la même d'un sens à l'autre nous détecterons des erreurs de compréhension lors des tests utilisateur.

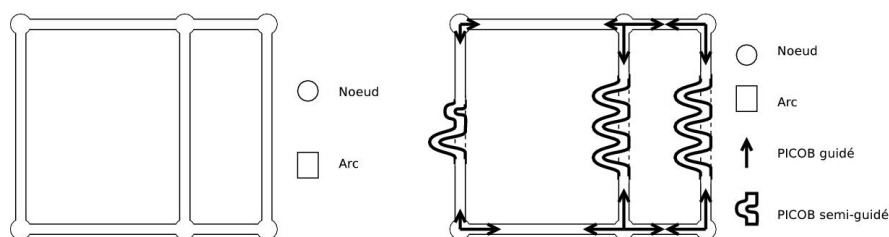


FIG. 4.4 – Représentation haptique d'un circuit électrique : à gauche sans PICOB, et à droite avec.

Conclusion

Ce stage avait pour but de créer une technique d'interaction haptique permettant d'afficher des messages. Je me suis tout d'abord intéressé à ce qui existe au niveau sonore et tactile. La technique utilisée par Brewster avec les earcons (auditif) et les tactons (tactile), consiste à utiliser plusieurs paramètres et de trouver plusieurs valeurs discriminantes pour chacun afin de construire des codes. Cette construction simple est intéressante et a donc été suivie pour définir les PICOB. Ainsi les paramètres possibles établis sont la direction, l'amplitude et la durée. Dans un premier temps la discrimination de directions a été testée. Il apparaît que l'on peut discriminer facilement six directions. Le test suivant portait sur les amplitudes : il s'agissait de reconnaître trois amplitudes. Les résultats ne permettent pas de dégager une discrimination. Ceci nous a conduit à revoir nos objectifs à la baisse dans la troisième expérience où il fallait déterminer simultanément la direction et l'amplitude. Toujours avec les six directions, l'utilisation de deux amplitudes donne de bons résultats : la discrimination est acquise. Cependant avec trois valeurs d'amplitude les résultats nous poussent à croire que sans apprentissage ce n'est pas utilisable avec les valeurs proposées. Or ces valeurs correspondent au seuil de confort des utilisateurs donc on ne peut pas chercher de valeurs plus grandes sans risquer de faire des interfaces potentiellement gênantes.

Ce travail est avant tout une introduction à un travail de plus grande envergure qui permettra de trouver plus de représentations et sous plusieurs formes. Tout d'abord nous devons tester les autres modes de lecture. L'application de circuits électriques utilise déjà les PICOB guidés et semi-guidés. Nous procéderons à des tests similaires à ceux décrits ci-dessus parallèlement à l'amélioration de ce programme. D'autres représentations, basées sur des effets haptiques comme les textures le frottement ou l'inertie sont envisageables. Nous en implémenterons, les testerons et les intégrerons si possible dans des interfaces haptiques.

Des expériences plus approfondies sur les impulsions seront effectuées, notamment pour tester des séquences de lettres. Le but étant de représenter des messages entiers, ce genre de tests est indispensable pour vérifier la faisabilité. Des paramètres comme le temps de pause entre chaque lettre et la longueur des lettres est à expérimenter.

Annexe A

Détails sur les circuits électriques

A.1 Spécifications

Le but est de faire passer les notions essentielles de physique de collège à travers une exploration des circuits :

- composants de base (résistances, condensateurs, générateurs, interrupteurs, etc.)
- connexions en série/parallèle (loi des noeuds)
- circulation du courant
- notion de conducteur et d'isolant
- mesure de la tension et de l'intensité
- notion de résistance (loi d'Ohm)
- notion de courant continu/courant alternatif
- utilisation d'un oscilloscope

A.2 Principes utilisés

Pour guider le PHANToM l'idée est toujours la même : utiliser un point attracteur qui va guider le périphérique où on le souhaite. Par exemple à proximité d'un arc on va projeter le point d'attraction sur l'arc proche afin de contraindre le pointeur sur le circuit. Un effet d'hystérésis a été créé pour éviter des vibrations (figure A.1). En effet si on utilise le même rayon pour la détection d'entrée ($r1$) et la détection de sortie ($r2$) de la zone d'attraction autour d'une ligne, la force appliquée peut changer brutalement et rapidement de sens et créer une vibration à cause du décalage entre la détection et le calcul de la force induit par l'API.

Pour les noeuds le principe est de faire sentir une surface sphérique trouée là où il y a des arcs pour pouvoir sortir. Tant qu'on est dans le rayon du noeud le pointeur est libre, par contre si on dépasse en dessous d'un epsilon prédéfini ($r2 - r1$), le point attracteur est projeté sur la sphère mais seulement dans le cas où il n'y a pas d'arc dans cette direction : ceci permet de créer des trous au niveau des arcs. La même technique d'hystérésis a été appliquée car le même

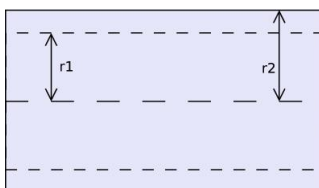


FIG. A.1 – Zone d'attraction d'une ligne

phénomène de vibrations indésirables peut se produire (figure A.2). Le rayon $r1$ correspond au rayon de détection d'entrée dans le noeud et le rayon $r2$ correspond au rayon de détection de sortie de la zone d'attraction du noeud.

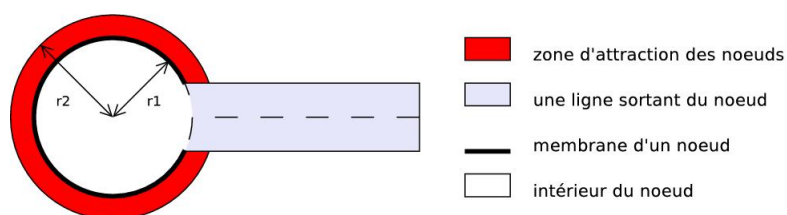


FIG. A.2 – Zone d'attraction d'un noeud

A.3 Améliorations futures

Le travail déjà effectué permet d'explorer le circuit et de reconnaître des composants. Dans un premier temps il faudra représenter tous les composants et tester la reconnaissance des codes. Dans un deuxième temps un guidage va être mis en place afin de permettre soit de prédéfinir un chemin, soit pour permettre à un enseignant de guider l'utilisateur.

Une autre amélioration prévue est de rajouter des codes permettant de lire les valeurs des composants par exemple. Ceci pourra se faire par un PICOB, par des vibrations ou par des sons. L'idée est d'exploiter la multimodalité.

Bibliographie

- [AW00] Roland Arsenault and Colin Ware. Eye-hand co-ordination with force feedback. In *CHI '00 : Extended abstracts on Human factors in computing systems*, pages 408–414, The Hague, The Netherlands, April 2000. ACM Press.
- [BB04a] Stephen A. Brewster and Lorna M. Brown. Non-visual information display using tactons. In *CHI '04 : Extended abstracts on Human factors in computing systems*, pages 787–788, Vienna, Austria, April 2004. ACM Press.
- [BB04b] Stephen A. Brewster and Lorna M. Brown. Tactons : structured tactile messages for non-visual information display. In *CRPIT '28 : Proceedings of the fifth conference on Australasian user interface*, pages 15–23, Dunedin, New Zealand, 2004. Australian Computer Society, Inc.
- [BWE93] Stephen A. Brewster, Peter C. Wright, and Alistair D. N. Edwards. An evaluation of earcons for use in auditory human-computer interfaces. In *CHI '93 : Proceedings of the conference on Human factors in computing systems*, pages 222–227, Amsterdam, The Netherlands, April 1993. ACM Press.
- [Cha00] Ben Challis. Design principles for non-visual interaction. In *CHI '00 : Extended abstracts on Human factors in computing systems*, pages 73–74, The Hague, The Netherlands, April 2000. ACM Press.
- [Chu02a] Lonny L. Chu. Haptic design for digital audio. In *ICME 2002 Proceedings*, 2002.
- [Chu02b] Lonny L. Chu. User performance and haptic design issues for a force-feedback sound editing interface. In *CHI '02 : Extended abstracts on Human factors in computing systems*, pages 544–545, Minneapolis, Minnesota, USA, April 2002. ACM Press.
- [COJ⁺02] Angela Chang, Sile O'Modhrain, Rob Jacob, Eric Gunther, and Hiroshi Ishii. Comtouch : design of a vibrotactile communication device. In *DIS '02 : Proceedings of the conference on Designing interactive systems*, pages 312–320, London, England, 2002. ACM Press.
- [DBRC⁺04] Joan De Boeck, Chris Raymaekers, Erwin Cuppens, Tom De Weyer, and Karin Coninx. Task-based abstraction of haptic and multisensory applications. In *Proceeding of the 4th International Conference EuroHaptics 2004*, pages 174–181, München, Germany, July 2004. ACM Press.

- [DMH00] Jack Tigh Dennerlein, David B. Martin, and Christopher Hasser. Force-feedback improves performance for steering and combined steering-targeting tasks. In *CHI '00 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 423–429, The Hague, The Netherlands, April 2000. ACM Press.
- [FS01] Masaaki Fukumoto and Toshiaki Sugimura. Active click : tactile feedback for touch panels. In *CHI '01 : Extended abstracts on Human factors in computing systems*, pages 121–122, Seattle, Washington, April 2001. ACM Press.
- [Gav93] William W. Gaver. Synthesizing auditory icons. In *CHI '93 : Proceedings of the conference on Human factors in computing systems*, pages 228–235, Amsterdam, The Netherlands, April 1993. ACM Press.
- [HACH⁺04] Vincent Hayward, Oliver R. Astley, Juan Manuel Cruz-Hernandez, Danny Grant, and Gabriel Robles De La Torre. Haptic interfaces and devices. *Sensor Review*, 24(1) :16–29, 2004.
- [Jeo01] Woo-Seob Jeong. Exploratory user study of haptic and auditory display for multimodal geographical information systems. In *CHI '01 : Extended abstracts on Human factors in computing systems*, pages 73–74, Seattle, Washington, April 2001. ACM Press.
- [JIs01] Gunnar Jansson and Anna Ivå s. Can the efficiency of a haptic display be increased by short-time practice in exploration? In *Proceedings of the First International Workshop on Haptic Human-Computer Interaction*, pages 88–97. Springer-Verlag, 2001.
- [JL02] Gunnar Jansson and Karin Larsson. Identification of haptic virtual objects with different degrees of complexity. In *Proceeding of the 2nd International Conference Eurohaptics 2002*, pages 57–60, Edinburgh, Scotland, July 2002. ACM Press.
- [LGV^T04] Patrick Lemoine, Mario Gutierrez, Frédéric Vexo, and Daniel Thalmann. Mediators : Virtual interfaces with haptic feedback. In *Proceeding of the 4th International Conference EuroHaptics 2004*, pages 68–73, München, Germany, July 2004. ACM Press.
- [ME03] Karon E. MacLean and Mario J. Enriquez. Perceptual design of haptic icons. In *Proceeding of the 3rd International Conference Eurohaptics 2003*, pages 351–363, Dublin, UK, July 2003. ACM Press.
- [MED03] Karon E. MacLean, Mario J. Enriquez, and Vincent DiLollo. Perceptual design of haptic icons : Towards an expressive haptic language. Submitted to CHI03, September 2003.
- [MS94] Thomas M. Massie and J. Kenneth Salisbury. The phantom haptic interface : A device for probing virtual objects. In *Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 295–301, Chicago, November 1994. vol. 1.
- [MWO⁺04] Georg Michelitsch, Jason Williams, Martin Osen, Beatriz Jimenez, and Stefan Rapp. Haptic chameleon : a new concept of shape-changing user interface controls with force feedback. In *CHI '04 :*

- Extended abstracts on Human factors in computing systems*, pages 1305–1308, Vienna, Austria, April 2004. ACM Press.
- [MZ98] Timothy Miller and Robert Zeleznik. An insidious haptic invasion : adding force feedback to the x desktop. In *UIST '98 : Proceedings of the 11th annual ACM symposium on User interface software and technology*, pages 59–64, San Francisco, California, USA, November 1998. ACM Press.
- [Nes02] Keith V. Nesbitt. Experimenting with haptic attributes for display of abstract data. In *Proceeding of the 2nd International Conference Eurohaptics 2002*, pages 150–155, Edinburgh, Scotland, July 2002. ACM Press.
- [NG001] Keith V. Nesbitt, Randall J. Gallimore, and Bernard J. Orenstein. Using force feedback for multi-sensory display. *Aust. Comput. Sci. Commun.*, 23(5) :64–68, 2001.
- [NR03] Andrew Nashel and Sharif Razzaque. Tactile virtual buttons for mobile devices. In *CHI '03 : Extended abstracts on Human factors in computing systems*, pages 854–855, Ft. Lauderdale, Florida, USA, April 2003. ACM Press.
- [OMBG00] Ian Oakley, Marilyn Rose McGee, Stephen A. Brewster, and Philip Gray. Putting the feel in look and feel. In *CHI '00 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 415–422, The Hague, The Netherlands, April 2000. ACM Press.
- [PMP05] Thomas Pietrzak, Benoît Martin, and Isabelle Pecci. Affichage d'informations par des impulsions haptiques. Submitted to IHM05, May 2005.
- [RAD96] Christophe Ramstein, Jean-François Arcand, and Martin Deveault. Adaptive user interfaces with force feedback. In *CHI '96 : Proceedings of the conference on Human factors in computing systems*, pages 406–407, Vancouver, British Columbia, Canada, 1996. ACM Press.
- [Ram96] Christophe Ramstein. Combining haptic and braille technologies : design issues and pilot study. In *Assets '96 : Proceedings of the second annual ACM conference on Assistive technologies*, pages 37–44, Vancouver, British Columbia, Canada, 1996. ACM Press.
- [RDLTH01] Gabriel Robles De La Torre and Vincent Hayward. La force au bout des doigts. *Pour la Science*, 288, 2001.
- [rea] *Reachin API 3.2 Programmer's Guide*. <http://www.reachin.se>.
- [RH94] Christophe Ramstein and Vincent Hayward. The pantograph : a large workspace haptic device for multimodal human computer interaction. In *CHI '94 : Proceedings of the conference on Human factors in computing systems*, pages 57–58, Boston, Massachusetts, United States, April 1994. ACM Press.
- [RKK97a] Diego C. Ruspini, Krasimir Kolarov, and Oussama Khatib. The haptic display of complex graphical environments. In *SIGGRAPH '97 : Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 345–352. ACM Press/Addison-Wesley Publishing Co., 1997.

- [RKK97b] Diego C. Ruspini, Krasimir Kolarov, and Oussama Khatib. Haptic interaction in virtual environments. In *Proceedings of the International Conference on Intelligent Robots and Systems, IROS '97*, pages 128–133, Grenoble, France, September 1997.
- [RMD⁺96] Christophe Ramstein, Odile Martial, Aude Dufresne, Michel Carignan, Patrick Chassé, and Philippe Mabillean. Touching and hearing gui's : design issues for the pc-access system. In *Assets '96 : Proceedings of the second annual ACM conference on Assistive technologies*, pages 2–9. ACM Press, 1996.
- [RvE04] A.F. Rovers and Harm A. van Essen. Him : a framework for haptic instant messaging. In *CHI '04 : Extended abstracts on Human factors in computing systems*, pages 1313–1316, Vienna, Austria, April 2004. ACM Press.
- [Sjö01] Calle Sjöström. Using haptics in computer interfaces for blind people. In *CHI '01 : Extended abstracts on Human factors in computing systems*, pages 245–246, Seattle, Washington, April 2001. ACM Press.
- [Tan00] Hong Z. Tan. Perceptual user interfaces : haptic interfaces. *Commun. ACM*, 43(3) :40–41, March 2000.
- [WB03a] Steven A. Wall and Stephen A. Brewster. Assessing haptic properties for data representation. In *CHI '03 : Extended abstracts on Human factors in computing systems*, pages 858–859, Ft. Lauderdale, Florida, USA, April 2003. ACM Press.
- [WB03b] Steven A. Wall and Stephen A. Brewster. Scratching the surface : Preliminary investigations of haptic properties for data representation. In *Proceeding of the 3rd International Conference Eurohaptics 2003*, pages 330–342, Dublin, UK, July 2003. ACM Press.
- [WLD04] Christopher R. Wagner, Susan J. Lederman, and Howe Robert D. Design and performance of a tactile shape display using rc servomotors. *Haptics-e*, 3(4), August 2004.
- [WM03] Jason Williams and Georg Michelitsch. Designing effective haptic interaction : inverted damping. In *CHI '03 : Extended abstracts on Human factors in computing systems*, pages 856–857, Ft. Lauderdale, Florida, USA, April 2003. ACM Press.
- [WSCN04] Robert L. Williams, Mayank Srivastava, Robert Conaster, and Howell John N. Implementation and evaluation of a haptic playback system. *Haptics-e*, 3(3), May 2004.
- [YRB01] Wai Yu, Ramesh Ramloll, and Stephen A. Brewster. Haptic graphs for blind computer users. In *Proceedings of the First International Workshop on Haptic Human-Computer Interaction*, pages 41–51. Springer-Verlag, 2001.
- [ZGLS04] Mounia Ziat, Olivier Gapenne, Charles Lenay, and John Stewart. Acuité perceptive via une interface pseudo-haptique. In *IHM 2005 : Proceedings of the 17th French-speaking conference of human-computer interaction*, pages 263–266, Namur, Belgium, August 2004. ACM Press.