

# Interfaces Réduites pour l'Entrée de Texte - État de l'Art et Présentation du Communicateur Sibylle pour l'Aide à l'Écriture de Personnes Handicapées

## *Reduced Interfaces for Text Input - State of the Art and Sibylle: AAC System for Disabled People*

**Franck POIRIER (1), Igor SCHADLE (2)**

(1) Laboratoire VALORIA, Université de Bretagne-Sud, France  
& Laboratoire TAMCIC, ENST de Bretagne, France

Franck.Poirier@univ-ubs.fr

(2) Laboratoire IRIT, Toulouse, France

schadle@irit.fr

**Résumé.** Cet article présente, dans une première partie, un état de l'art des méthodes de saisie de données adaptées aux dispositifs mobiles et aux interfaces réduites (assistant numérique, téléphone mobile, tablette PC...). Dans une deuxième partie, l'article détaille le communicateur Sibylle, qui est un système d'aide à la communication écrite pour personnes handicapées. Sibylle illustre particulièrement le problème de l'interaction dégradée où l'interface d'entrée est extrêmement réduite (bouton-poussoir). Les méthodes de saisie de données pour les dispositifs mobiles comme les communicateurs ont en commun de participer à l'objectif d'accessibilité en permettant à tous, valides ou handicapés, et partout, de produire de l'information et plus généralement de communiquer.

**Mots-clés.** Saisie de données, handicap, communicateurs, interaction dégradée, interface réduite, dispositif mobile, clavier virtuel.

**Abstract.** This article presents, in a first part, a state of the art of the text input methods for mobile devices with reduced interfaces (PDA, mobile phone, tablet PC...). In the second part, an AAC system for people with severe motor impairments (cerebral palsy), called Sibylle, is presented. In the first part, a typology of the different methods and devices is proposed and some results of evaluation are given. The second part is devoted to an example of degraded interaction. The design of Sibylle, the dynamic scanning keyboard and the letter and word predictors are more particularly described.

**Keywords.** Text input, AAC system, degraded interaction, reduced interface, mobile device, software keyboard.

## 1 Introduction

Les nouveaux appareils numériques de présentation, de traitement et de communication de l'information, plus simplement appareils d'info-communication, sont désormais plus nombreux que les ordinateurs personnels. Les interfaces de type WIMP qui équipent ces derniers ont été conçues, il y a 30 ans pour des usages professionnels standard et limités. De fait, les interfaces WIMP sont très mal adaptées aux systèmes apparaissant actuellement (Beaudoin-Lafon, 2000). Cette inadaptation s'explique par l'usage nomade des nouveaux dispositifs et les limitations physiques des périphériques (petit écran, petit clavier voire absence de clavier...), mais aussi par l'élargissement du cercle des utilisateurs à l'ensemble de la population (non-spécialistes, personnes jeunes, âgées, handicapées...).

Cet article illustre le double intérêt des nouvelles méthodes d'entrée de texte. D'une part, ces méthodes permettent d'interagir dans un contexte de mobilité par l'intermédiaire d'un clavier virtuel ou d'un dispositif porté, et d'autre part, elles facilitent la communication des personnes qui ne peuvent utiliser un clavier physique.

Pour traiter ce double enjeu des méthodes d'entrée de texte, cet article présente, dans une première partie, un état de l'art des méthodes de saisie alternatives et, dans une seconde partie, le cas particulier d'un communicateur pour l'aide à la communication écrite d'une personne handicapée – le système Sibylle.

## 2 Saisie de texte sur assistant numérique

Mark Weiser (Weiser, 1991) avait clairement annoncé en 1991 que la « machine » du XXI<sup>e</sup> siècle ne serait plus un ordinateur unique « bon » à tout faire, comme le PC, mais qu'elle serait remplacée par une multitude de dispositifs adaptés à une tâche spécifique.

Jef Raskin (2000) a qualifié ces systèmes « information appliances » à la fin des années 80, terme repris et popularisé ensuite par Don Norman (1998). En français, l'expression « appareil d'infocommunication » peut être retenue.

Depuis dix ans, plusieurs de ces appareils se sont effectivement développés : assistant personnel (PDA, pocket PC, smart phone), téléphone mobile, ordinateur ultra-portable, ordinateur tablette, appareil photo numérique, navigateur GPS... Contrairement au PC, ces systèmes ne s'utilisent pas sur un coin de bureau. Ce sont des objets nomades qui sont employés parfois debout, en marchant, posés sur les genoux, à une main... Le clavier, héritage de la machine à écrire, a souvent disparu de ces nouveaux systèmes ou a été remplacé par un mini-clavier dont la fonction n'est pas de saisir de longs textes.

L'entrée d'information, indispensable sur tout système interactif reste un problème difficile qui ne trouve pas toujours de solution satisfaisante (Stephanidis, 2003). L'entrée d'information est assurée généralement par un dispositif alternatif au clavier. La technique retenue pour saisir l'information joue un rôle crucial dans le succès ou l'échec de ces systèmes. La conception d'une bonne méthode de saisie est ainsi devenue un réel sujet de recherche en IHM et de multiples méthodes ont été proposées depuis dix ans.

Ces méthodes ne sont pas universelles, elles se positionnent par leurs réponses aux questions suivantes :

- quels dispositifs d'entrée (clavier, stylet, doigt...) et pour quels types de système (assistant, borne interactive, ordinateur nomade...)?
- quels contextes d'interaction (bureau, véhicule, domicile, loisirs, professionnel, en mouvement...)?

- quels utilisateurs (non expérimenté, expert, habile, personne handicapée, jeune, âgée...) ?
- quels organes et appendices sollicités (une main, deux mains, un doigt, dix doigts...) ?
- quelle modélisation du langage et quel type de production (texte, message, commande, chaîne numérique...) ?
- quelle évaluation et quelle performance (pas d'évaluation, évaluation quantitative ou qualitative, métrique retenue...) ?

Les réponses à ces questions permettent d'établir une typologie des différentes méthodes de saisie qui est présentée ci-après.

L'évaluation des méthodes de saisie repose principalement sur deux métriques : le nombre de mots saisis par minute ou *wpm* (word per minute) et le nombre d'appuis de touches par caractère ou *kspc* (keystroke per character) (Soukoreff et MacKenzie, 1995), (MacKenzie, 2002). Pour une entrée gestuelle, la métrique est le nombre de gestes par caractère ou *gpc* (gesture per character).

Il faut remarquer que certaines méthodes de saisie n'ont pas fait l'objet d'une évaluation rigoureuse ou que les performances annoncées varient sensiblement selon les sources. Aussi, pour certaines méthodes, soit aucune performance ne sera précisée, soit les performances les plus probables seront mentionnées.

### 3 Typologie des méthodes de saisie

La plupart des méthodes de saisie (Sirisena, 2002 ; Poirier et Schadle, 2004) ont pour objectif d'être une alternative à l'usage du clavier standard (82 à 108 touches). Ces méthodes sont conçues pour des saisies à partir d'une interface réduite dans des situations non-standard. Ces situations particulières relèvent soit du contexte, comme en mobilité (« one-handed data entry on mobile computer »), soit des caractéristiques de l'utilisateur, une personne handicapée par exemple. Dans les deux cas, la saisie s'effectue au moyen d'une interface réduite par rapport à celle fournie par un clavier complet.

L'analyse des différentes méthodes de saisie de données à interface réduite (MSDIR) permet de dégager les grandes catégories suivantes :

- clavier virtuel (soft keyboard),
- clavier physique réduit,
- clavier réduit ambigu ou à appuis multiples (multitap),
- clavier réduit à combinaison de touches ou à accord (chording keyboard),
- entrée par bouton unique,
- entrée gestuelle analogique,
- entrée gestuelle non analogique par désignation de cibles (flicking gesture entry),
- entrée tactile avec guidage tangible,
- entrée par mouvement d'inclinaison sans tracé (tilting keyboard),
- entrée gestuelle avec reconnaissance de l'écriture,
- entrée avec prédiction linguistique.

Cette liste n'est pas exhaustive. Ces différentes catégories ne sont pas indépendantes les unes des autres, par exemple un clavier peut combiner l'entrée par accord et par inclinaison ; de même, une entrée sur clavier ambigu peut recourir à une prédiction linguistique. Ainsi, dans l'espace de conception des MSDIR, les catégories précédentes se recouvrent largement.

Dans la suite, les grandes caractéristiques de chacune de ces catégories sont présentées.

### 3.1 Entrée avec clavier virtuel

Le clavier physique est remplacé par un clavier virtuel affiché à l'écran. Celui-ci pourrait représenter fidèlement un clavier d'une centaine de touches, mais il se limite souvent à un clavier faiblement réduit comprenant de l'ordre de 80 touches pour minimiser la surface d'affichage.

Si la saisie s'effectue au stylet, contrairement au clavier physique, l'usage de modes kinesthésiques (comme le mode « majuscule ») n'est pas possible. Il faut soit combiner la sélection au stylet et l'usage d'un bouton, soit découper le clavier en différents modes (numériques, caractères spéciaux, majuscules bloquées...). L'espace économisé à l'écran avec un clavier faiblement réduit ne représente pas un avantage suffisant au regard de l'inconvénient dû aux modes : une augmentation de la charge cognitive de l'utilisateur et des erreurs.

Un aspect pour le moins critiquable de cette approche est qu'elle reprend la configuration des touches définie par Christopher Sholes en 1868 et qui repose sur des considérations purement mécaniques. Déjà non optimisé pour la frappe (la première méthode de frappe remonte à 1888, soit vingt ans après l'invention du clavier qwerty), le clavier virtuel de type qwerty n'est pas davantage optimisé pour la saisie au stylet. Il conduit à une saisie lente, inadaptée à de longs textes. Il consomme entre le tiers et la moitié de la surface totale sur un PDA (figure 1). Il ne permet pas à l'utilisateur d'améliorer à l'usage sa vitesse de saisie et empêche la saisie « à l'aveugle ».



Figure 1. Clavier virtuel occupant 50% de l'écran sur PDA

Le clavier qwerty a un wpm de l'ordre de 30 (TextWare, 1998), mais la vitesse de saisie réelle à un doigt n'est que de 11 et rarement de plus de 15 sur ordinateur. Son seul intérêt est d'avoir une utilisation immédiatement compréhensible, même par un opérateur non expérimenté. Des versions pliantes (foldable keyboard) présentent un avantage pour la mobilité, mais n'ont pas d'intérêt en dehors d'une utilisation de bureau.

#### Claviers virtuels optimisés

À l'inverse du qwerty, le clavier Dvorak propose une disposition des touches étudiée pour la saisie rapide. Les touches sont organisées afin de minimiser le déplacement entre les rangées et de rapprocher des lettres se succédant fréquemment. En théorie, un clavier virtuel Dvorak est donc plus efficace par rapport à la loi de Fitts (1954), car il minimise le déplacement inter-rangée de la souris ou du stylet. En fait, l'organisation des touches n'est pas optimale par rapport à la saisie réelle de données à l'ordinateur, qui n'est pas seulement textuelle ; le gain supposé n'est donc pas significatif.

D'autres claviers moins connus proposent un ré-arrangement plus optimal des touches : le clavier Metropolis d'IBM (Zhai *et al.*, 2000), le clavier FITALY produit

par Textware Solutions (1998), le clavier OPTI (MacKenzie et Zhang, 1999) ou encore le clavier KNITS développé par l'équipe de recherche du centre Enkidu de New York (Lesher et Moulton, 2000).

L'organisation de ces différents claviers repose sur la minimisation d'une fonction de coût qui utilise la loi de Fitts (Zhai *et al.*, 2002).

Le clavier KNITS propose une organisation optimale des touches déterminée à partir de l'étude des fréquences bigrammiques des lettres sur un corpus de 3 millions de mots. Le clavier correspond à une grille de 5 x 6 touches (figure 2). La réduction de la longueur des déplacements par rapport à l'ordre alphabétique est de 32 % et de 35 % par rapport à l'ordre qwerty. L'évaluation du clavier KNITS annonce un wpm de l'ordre de 40.

Z	G	C	H	V	X
K	N	I	T	S	W
Espace	A	E	Espace		
Y	D	L	R	O	F
Q	B	U	M	P	J

Figure 2. Clavier optimisé KNITS

Le clavier Metropolis (figure 3) a la particularité d'avoir les voyelles organisées le long des médianes d'un triangle virtuel. La touche Espace, le caractère le plus utilisé (15 à 20% des caractères d'un texte) est au centre. L'évaluation donne un excellent wpm de 43, contre 30 pour un clavier qwerty.

X	K	W	M	U	Q	'
	C	H	T	O	F	Z
J	I	E	Esp	N	G	B
	V	R	S	A	D	Ret
,	X	P	L	Y	Sh	

Figure 3. Clavier optimisé Metropolis

### 3.2 Entrée avec clavier physique réduit

De très nombreuses études ont porté sur les claviers réduits et sur la définition d'un clavier optimisé par rapport au nombre de touches ou à la vitesse de saisie.

Le clavier fortement réduit le plus fréquent est le clavier téléphonique à 12 touches appelé clavier T12 (figure 4). Ces claviers sont qualifiés d'ambigus car une même touche sert pour entrer plusieurs caractères (la touche 2 sert aussi à entrer A, B, C et parfois a, b, c, à, ae, ä, ç...). La méthode de saisie est qualifiée de « multitap » car plusieurs appuis sont nécessaires pour entrer un caractère (dans l'exemple de la touche 2, sur certains modèles, il faut dix appuis pour entrer « ç »).



Figure 4. Clavier téléphonique T12

Les mesures théoriques donnent un wpm de l'ordre de 20 pour ce type de clavier, mais des évaluations réelles donnent plutôt un wpm de l'ordre de 5. La saisie d'un caractère comme « ç » conduit à un taux élevé d'erreurs et globalement à une saisie fastidieuse. Par ailleurs, les experts obtiennent un wpm proche de celui des débutants, supérieur d'une unité seulement, car les erreurs diminuent peu à l'usage et leurs corrections sont très coûteuses en temps.

Si la saisie est limitée aux seuls chiffres et lettres majuscules (dans ce cas, la touche 2 correspond seulement aux caractères A, B, C), la méthode multitap correspond à un kspc de l'ordre de 2. Dans le cas où une touche correspondrait à une dizaine de caractères, le kspc devient très mauvais, de l'ordre de 5.

Un autre inconvénient de l'entrée sur T12 est qu'elle est de type zoné, c'est-à-dire qu'un caractère ou un ensemble de caractères est associé à une zone de saisie, en l'occurrence une touche. L'attention de l'utilisateur doit donc être soutenue, l'usage à l'aveugle n'est pas possible. Pour arriver sans erreurs à sélectionner le n° caractère par n appuis rapides, le feedback visuel doit être de qualité, ce qui est difficile à réaliser sur un petit écran.

Et pourtant, malgré tous ces défauts, l'approche multitap reste la plus utilisée principalement pour envoyer quelques centaines de milliards de messages courts (SMS) chaque année.



Figure 5. Clavier Fastap et BlackBerry 7100

Il existe beaucoup d'autres claviers faiblement réduits (de 50 à 80 touches), mais qui n'offrent pas d'avantage en mobilité. À l'inverse, la figure 5 présente deux types de claviers faiblement réduits intéressants dans ce contexte, il s'agit du clavier Fastap™ et des mini-claviers (thumbboards) équipant des PDA communicants (par exemple le Treo 600, le BlackBerry 7100 ou le Sidekick 2).

Le clavier Fastap™ (Cockburn et Sirisena, 2003) est construit autour d'un clavier téléphonique standard à 12 touches (0 à 9, \*, #). Autour et entre ces 12 touches sont ajoutées 40 touches. L'entrée d'un numéro s'effectue normalement, directement sur les touches numériques et la saisie des caractères alphabétiques et des méta-caractères (espace, suppression, retour à la ligne, majuscule, @...) directement par appui simple sur les autres touches.

Ce clavier présente comme principaux avantages d'être compact, immédiatement compréhensible et de ne pas recourir à des modes (pas de mode numérique). En termes de performances, il a un kspc de 1 par conception et un wpm de l'ordre de 7.

### 3.3 Entrée par accord

Cette approche avait déjà été suivie par Douglas Engelbart (1962) dans le projet NLS (oNLine System) mené à partir du début des années 60 à Stanford. Elle utilise un clavier à quelques touches (de l'ordre de 5 à 10), chaque caractère correspondant à une combinaison de touches ou accord (chording keyboard). Ces claviers se prêtent bien à l'utilisation nomade en mouvement.

CyKey (Bellaire Electronics), Twiddler (Handikey Corp.), FrogPad (Frogpad Corp.) et in10did (in10did Corp.) sont des exemples de claviers à accords.

L'inconvénient de ces claviers est d'exiger un apprentissage préalable des accords et une certaine dextérité. Il n'est pas adapté à un usage occasionnel, car dans ce cas, les accords doivent être ré-appris à chaque utilisation. Il est à noter que le chord keyboard du système NLS n'a pas vraiment convaincu, contrairement à l'autre dispositif de saisie qui était la première souris-pointeur de l'histoire et dont chacun connaît le succès. Engelbart avait compris que l'interaction clavier-souris ne pouvait pas être efficace dans la mesure où le clavier sollicite déjà les deux mains et que l'ajout de la souris imposait un va-et-vient incessant d'une main entre le clavier et la souris. Le modèle GOMS - Keystroke Level Model (Card *et al.*, 1980) associe classiquement à cette opération de rapatriement de la main (homing) un temps de 0,4 s. Le chord keyboard du système NLS avait pour objectif de permettre une interaction efficace à deux mains, une pour les accords, l'autre pour la souris.



Figure 6. Clavier à accords in10did et Twiddler 2

CyKey repose sur une analogie entre la forme de l'accord et la forme du caractère pour faciliter l'apprentissage qui reste, malgré tout, assez difficile. Le clavier comprend 9 touches et un accord requiert au maximum 4 touches. CyKey est apparu en 1978 sur un des premiers organizers (Agenda de Microwriter).

Twiddler (figure 6) est plus récent. Il a été présenté comme un clavier pour l'informatique vestimentaire, qui tient dans la poche. Il comprend 6 touches activables avec le pouce et 12 touches avec les quatre doigts. Il émule un clavier complet de 101 touches. Il dispose également d'un Trackpoint™ pour déplacer le

pointeur de la souris. On peut facilement l'avoir dans une poche, jouer des accords pour entrer du texte et sélectionner avec la souris.

In10Did (figure 6) est un clavier de taille carte de crédit, qui possède 10 touches. Il est prévu pour un jeu à deux mains. Certains modèles permettent de déplacer un pointeur par inclinaison du clavier tout en appuyant sur une des touches avec le pouce (mode kinesthésique). Il émule également un clavier complet. Les accords respectent l'ordre alphabétique, par exemple si le caractère A est produit en appuyant sur la touche de l'index, le caractère B est produit en appuyant sur la touche du majeur. Le principe est simple, mais il n'est pas sûr que ce moyen mnémotechnique soit suffisant pour une saisie « sans réfléchir », sachant que les claviers physiques organisés selon l'ordre alphabétique sont déjà peu efficaces. Les touches semblent déjà trop nombreuses pour une mémorisation facile des accords.

### 3.4 Entrée à bouton unique

Il est également possible d'entrer du texte avec un seul bouton (bouton physique, ou objet d'interaction de type bouton cyclique, curseur...). Ce dispositif permet une sélection en dimension 1, en d'autres termes de balayer une liste. Pour la saisie de texte, la liste correspond à un alphabet de caractères. Cette approche est généralement employée en véhicule, par exemple pour entrer une destination sur un système de navigation.

Il existe des boutons plus évolués, par exemple le PowerMate (Griffin Technology) qui est clicable et distingue les clics courts et les clics longs. Avec un tel dispositif, on peut envisager de gérer un déplacement en 2D, le clic court permettant de changer d'axe et le clic long de valider. Il est ainsi possible avec un seul bouton de simuler une entrée gestuelle selon quatre ou huit directions de type EdgeWrite, T-Cube ou QuikWriting (Cf. *infra*).

L'inconvénient est principalement la lenteur de saisie et la difficulté de mémorisation des modes dans le cas d'un bouton clicable.

### 3.5 Entrée gestuelle analogique

Ces méthodes reposent sur la saisie de symboles avec un stylet sur une surface tactile. L'utilisateur effectue un geste dont le tracé 2D présente une analogie avec le symbole à saisir. Par exemple, pour entrer le caractère O, il trace un cercle, pour un I, un trait vertical ou encore pour la commande « effacer le dernier caractère » un trait horizontal vers la gauche.

Cette approche concerne uniquement les systèmes équipés d'un écran tactile (PDA) et n'est donc pas généralisable à l'ensemble des systèmes interactifs (téléphone mobile, GPS portable...).

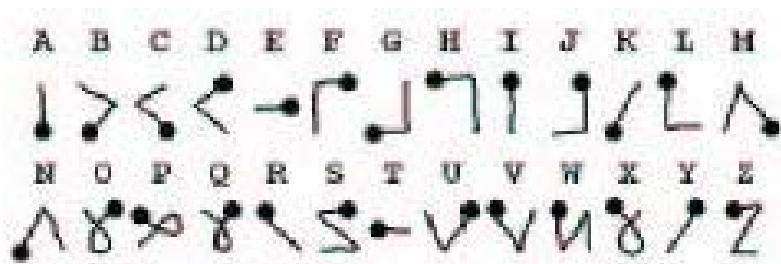


Figure 7. *Alphabet Unistrokes*

La plupart de ces méthodes sont de type « un trait par caractère » (unistroke). La carte de caractères de Unistrokes (Goldberg et Richardson, 1993) est présentée à la



figure 7, mais l'origine de cette approche est en fait ancienne (Dimond, 1957). Tracer un caractère d'un seul trait présente un triple avantage. Cela permet une saisie relativement rapide, consomme peu d'espace écran et simplifie considérablement le problème de la reconnaissance du caractère. L'inconvénient de l'approche unistroke est d'une part de fixer une forme standard et arbitraire pour chaque caractère ; d'autre part, sur un PDA, d'imposer une interaction à deux mains, une pour tenir l'appareil et l'autre pour écrire avec le stylet.

La méthode la plus connue de cette famille est Graffiti® (figure 8), inventée par Jeff Hawkins, fondateur de Palm Computing, quelques années avant la sortie du premier PDA Palm. La forte analogie entre le geste à effectuer et le caractère roman majuscule est le principal avantage de cette méthode. La mémorisation des gestes est ainsi relativement facile, l'alphabet est directement utilisable, tout au moins pour les lettres et les chiffres. La mémorisation des autres caractères (@, {, \*...) reste difficile. L'utilisation de ces caractères augmente les erreurs et détériore le wpm.

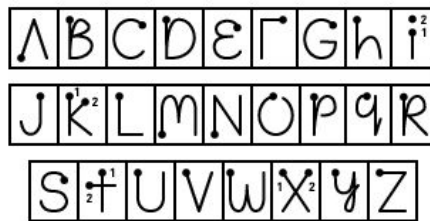


Figure 8. *Alphabet Graffiti 2*

Les inconvénients sont nombreux : certains caractères (e, é, v, n, u, x, k) sont sujets à de fréquentes erreurs qui ne diminuent pas à l'usage, le contrôle du stylet doit être précis et éviter tout rebond (sinon, le caractère n'est plus unistroke), Graffiti n'est donc pas adapté aux personnes peu habiles.

Graffiti implique différents modes (lettres, chiffres, ponctuations, symboles...) car certains caractères sont associés au même geste (i/1, o/0, 2/Z...), ces modes sont gérés soit par des zones de saisie différentes (lettres/chiffres), soit par un geste de changement de mode (tap au stylo en préfixe pour les ponctuations et les caractères spéciaux).

Les évaluations montrent que le wpm est de l'ordre de 6, ce qui est faible car l'écriture n'est pas un procédé rapide d'entrée de données ; le gpc est de 1,62, ce qui est élevé pour une approche unistroke.

### 3.6 Entrée gestuelle non-analogique

À l'inverse de l'approche précédente, il ne s'agit pas de tracer au stylet un caractère simplifié, mais d'effectuer un geste pour atteindre une cible ou une zone assignée au caractère à saisir. En général, le geste se compose d'un ou plusieurs segments, chaque segment suit une direction cardinale (Nord, Est, Sud, Ouest, noté N, E, S, O dans la suite) ou diagonale (NE, SE, SO, NO).

L'utilisateur doit mémoriser le geste associé à chaque caractère (par exemple NE + O pour le caractère « f » avec la méthode QuikWriting). Comme cette approche ne repose pas sur une analogie entre le caractère roman et le geste, l'apprentissage est plus difficile. Une première phase d'apprentissage (mode débutant) est nécessaire avec un feedback visuel du tracé et des cibles à atteindre. Dans cette phase, la consommation d'espace à l'écran est importante. En phase d'usage réel (mode expert), les cibles ne sont plus affichées et l'utilisateur trace en aveugle.

Les méthodes T-Cube (Venolia et Neiberg, 1994) et QuikWriting (Perlin, 1998) sont les plus connues. Dasher (Ward *et al.*, 2000) est une autre méthode de ce type à saisie continue.

T-Cube (figure 9) repose sur des menus à secteurs (pie-menus). Les gestes (flicks) sont caractérisés par un point de départ et une orientation (cardinales ou diagonales). Le premier segment vise un menu parmi 9 et le deuxième segment désigne un des 8 secteurs du menu. Les gestes sont de faible amplitude, ce qui permet une petite surface de saisie (un inch carré) et une entrée rapide de l'ordre de 60 à 100 caractères par minute.

Vu le nombre de caractères, il est difficile d'associer une cible à chaque caractère, T-Cube a donc recours à des modes (par exemple, les modes accent et majuscule). Chaque mode peut coder au plus 72 (9 x 8) caractères. Plusieurs organisations particulières des caractères (layout) ont été étudiées, par exemple, regrouper les voyelles sur un même menu *vs* mettre une seule voyelle par menu. Aucune organisation optimale n'a été trouvée, cependant les caractères « non-imprimables » (tabulation, retour arrière, entrée...) sont toujours disposés dans la zone centrale du menu.

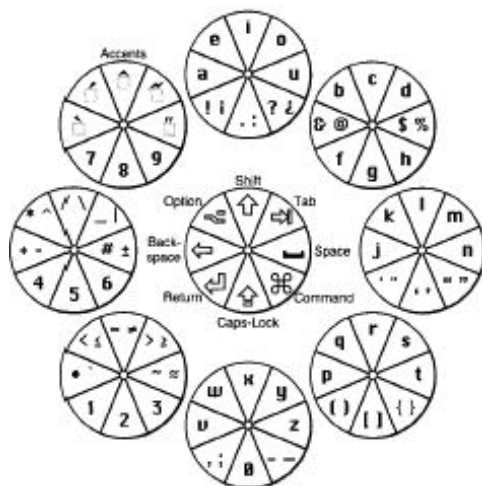


Figure 9. 9 cibles T-Cube

QuikWriting (figure 10) est dans la lignée de T-Cube. Il est plus rapide car il repose sur une saisie continue (T-Cube propose une saisie par caractère), ce n'est donc plus une approche de type unistroke. QuikWriting organise les caractères selon une rosace à 8 zones (NE puis O pour f), les caractères les plus fréquents sont associés à des cibles facilement mémorisables (par exemple E pour e).

Quatre modes sont nécessaires pour structurer l'ensemble de l'alphabet : lettre, lettre majuscule, numérique, ponctuation. Chaque mode est associé à une cible et le changement de mode correspond à la saisie d'un caractère particulier ( $\Delta$ ,  $\nabla$ ,  $\circ$ ). Comme la saisie est continue, la levée du stylet provoque la fin de la saisie QuikWriting et le retour à Graffiti. Après deux ou trois heures d'un apprentissage jugé laborieux, QuikWriting serait 3 fois plus rapide que Graffiti.

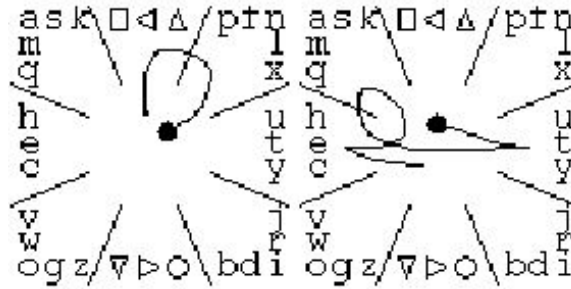


Figure 10. *QuikWriting* – saisie de « f » et de « the »

### 3.7 Entrée tactile avec guidage tangible

Cette approche reprend à la fois des idées de l'entrée gestuelle analogique et non-analogique. EdgeWrite (Wobbrock *et al.*, 2003) est l'exemple le plus connu.

EdgeWrite (figure 11), comme Graffiti, associe à chaque caractère un geste qui rappelle l'écriture de ce caractère en majuscule romaine. La surface de saisie est délimitée par une découpe rectangulaire (1,3 x 1,3 cm) posée sur la surface tactile. L'avantage de l'approche est triple : le scripteur est guidé par les bords physiques de la surface de saisie (edge), il peut faire des gestes rapides qui sont arrêtés par la rencontre d'un bord ou d'un coin et il peut ainsi plus facilement effectuer la saisie tout en bougeant (on the move). Il n'a donc pas à ajuster son geste à la taille de la zone de saisie comme avec Graffiti.



Figure 11. *EdgeWrite* – guide tangible

Par rapport à la loi de Fitts, le guide physique augmente indirectement la taille de la cible, c'est-à-dire le coin visé, par phénomène d'overshooting décrit initialement par Jef Raskin (2000).

L'avantage, du point de vue algorithmique, est que la reconnaissance du caractère est bien plus facile, dans la mesure où il est représenté par une séquence de coins et non par un modèle de geste assez complexe.

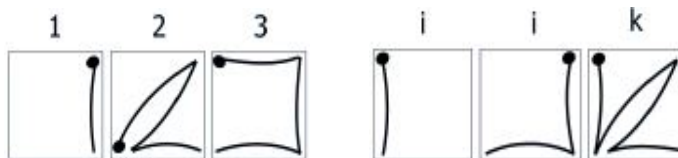


Figure 12. *Éléments de l'alphabet EdgeWrite*

EdgeWrite présente moins de modes que Graffiti (il existe un mode ponctuation), par exemple les caractères *i* et *l* sont dans un même mode et correspondent respectivement à un trait vertical de haut en bas le long du bord gauche et le long du bord droit (figure 12) ; les caractères majuscules sont introduits par un suffixe (trait vers le coin en haut à gauche).

L'évaluation de EdgeWrite donne un gpc de 1,26 et un wpm amélioré de 18% par rapport à Graffiti pour des débutants, mais du même ordre, un peu moins de 6, pour des experts.

Autre dispositif de saisie tactile (Doignon et Kamp, 1997), celui dont le brevet a été déposé en 1997 dans le cadre de la thèse de Jean-François Kamp (1998). Le dispositif est schématisé par la figure 13. Il a été conçu pour « communiquer » avec des systèmes embarqués tout en conduisant (par exemple, entrer une destination) en écrivant directement avec le doigt.

La différence avec EdgeWrite est que la surface tactile est activée au doigt et non pas au stylet et que le guidage n'est pas limité au bord, mais s'applique également aux directions latérales et diagonales de la surface de saisie. Ainsi, un caractère n'est pas représenté par une séquence de coins, mais par une séquence de segments. Seize segments sont reconnus sur la surface à l'aide de capteurs, deux segments par bord, par direction latérale et diagonale. Les combinaisons sont donc plus nombreuses, ce qui permet de limiter les modes et d'avoir des gestes qui présentent une meilleure analogie avec le caractère roman.

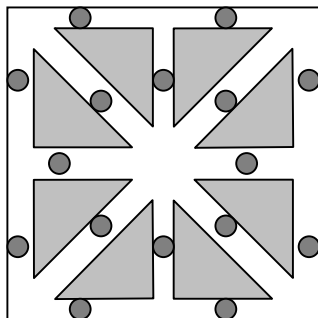


Figure 13. Saisie tactile au doigt

### 3.8 Entrée par mouvement sans tracé

Cette approche permet de saisir les caractères sans stylet en appliquant directement un mouvement au dispositif. Le mouvement de la main est en général une inclinaison (tilt) vers le haut, le bas, la gauche ou la droite, ou encore une secousse. On parle de clavier à inclinaison (tilting keyboard) ou d'écriture par inclinaison (tilt-to-write).

L'avantage premier est de permettre, contrairement aux méthodes précédentes, la saisie à une main, qui tient à la fois le dispositif et applique le mouvement de saisie. Cette approche est particulièrement indiquée si une main est déjà requise par une autre tâche.

Unigesture (Sazawal *et al.*, 2002) et MDITIM (Isokoski et Raisamo, 2000) sont des exemples de claviers à inclinaison.

Unigesture organise les caractères en 8 zones périphériques, 7 pour les lettres (3 à 4 lettres par zone) et une zone au sud pour l'espace. Unigesture est donc de type clavier ambigu, la désambiguïsation est faite par prédiction lexicale, déclenchée après la saisie de l'espace (inclinaison vers le bas). Il n'y a pas de feedback visuel, ce qui

permet de ne pas consommer d'espace à l'écran et de porter cette approche sur de petits dispositifs. En revanche, il est nécessaire de mémoriser la carte des caractères. Les utilisateurs familiarisés avec l'usage d'un PDA y arrivent sans difficulté.

Unigesture conduit à des erreurs en assez grand nombre, soit par mauvaise mémorisation de la carte des caractères, soit à cause de problème moteur ou de mauvaise discrimination entre inclinaisons proches (N et NO par exemple). Les utilisateurs comprennent spontanément le principe de la méthode. La saisie d'un caractère sur zone diagonale ne conduit pas à plus d'erreurs que sur zone cardinale. À l'usage, certains utilisateurs mentionnent une fatigue du poignet, Unigesture n'est donc pas adaptée à la saisie de longs textes. L'évaluation montre une vitesse de saisie relativement lente de 3 secondes par caractère (4 wpm).

### 3.9 Entrée gestuelle avec reconnaissance de l'écriture

Une autre approche consiste à écrire « naturellement » les caractères, sans contrainte, avec le stylet (ou le doigt) sur la surface tactile. L'origine de cette approche est ancienne, les premiers travaux de reconnaissance de caractères en ligne remontent au début des années 60 (Frishkopf et Harmon, 1961).

Dans ce cas, il n'y a pas de modèle de mémorisation des caractères pour le scripteur, les caractères sont écrits le plus souvent isolément en majuscules (logiciel MyScript de Vision Objects pour PDA) ou plus rarement en continu (reconnaisseur InkWell dérivé de Calligrapher de Apple). L'utilisateur est relativement libre de son geste, c'est au système de reconnaître son écriture. Contrairement à l'approche unistroke, un caractère ne se limite pas à un trait, il peut en comprendre jusqu'à 4 (E par exemple).

Ce type de saisie est immédiatement compréhensible par l'utilisateur, l'écriture est intuitive, caractère par caractère, et n'impose pas de mémoriser une carte des caractères.

La reconnaissance d'écriture peut être proposée sur de petits assistants numériques comme un téléphone mobile car elle requiert peu d'espace à l'écran (Anquetil et Bouteruche, 2004).

Les deux principaux inconvénients sont que, l'écriture étant plus lente que la frappe, ce mode d'entrée n'est jamais rapide, de l'ordre de 12 à 23 wpm (Card *et al.*, 1983), et que plusieurs modes (majuscule, numérique, clavier virtuel) sont obligatoires pour éviter les erreurs de reconnaissance dues à la similarité entre caractères (entre I et 1, o et O, 2 et Z, { et [...]).

### 3.10 Entrée à prédiction linguistique

La prédiction linguistique est une fonction qui peut être présente dans toutes les méthodes précédentes. Elle consiste à prédire la lettre suivante, la fin du mot, voire la fin de la phrase en cours de saisie. Ce service repose sur une modélisation du langage et relève donc de l'ingénierie linguistique.

T9™, de Tegic Communications, qui est disponible sur la majorité des téléphones mobiles, assure une prédiction lexicale.

SureType™ de Research In Motion effectue une prédiction lexicale à partir d'un dictionnaire de 30 000 mots. Cette méthode équipe la série BlackBerry 7100 (figure 5) et permet la saisie directe, sans multitap, de numéros de téléphone et de textes sur un clavier qwerty réduit (thumbboard) de 20 touches (Research In Motion, 2004).

D'autres méthodes fonctionnent par prédiction comme LetterWise (MacKenzie *et al.*, 2001) de Eaton Ergo pour la prédiction de lettres, Dasher (Ward *et al.*, 2000) et Unigesture (Sazawal *et al.*, 2002) pour la prédiction de mots.

T9 (taper avec 9 touches) permet d'appuyer une seule fois sur une touche d'un clavier ambigu T12 pour entrer un caractère. Par exemple, pour le mot « MON », les

trois lettres qui le composent se trouvent sur la même touche, la touche 6 qui est étiquetée par les lettres MNO. Les trois appuis sur cette touche vont afficher successivement M, ON, NON. Le mot affiché « NON » n'est pas celui attendu. Un quatrième appui, cette fois, sur la touche 0 permet finalement d'obtenir le mot désiré « MON ». La saisie a nécessité seulement 4 appuis au lieu de 6 (sans compter les pauses intermédiaires puisque les lettres sont sur la même touche) pour composer le mot. Les performances en kspc de T9™ sont de 1,0051 appui par lettre.

Comme d'autres méthodes, T9 présente un écart important entre performances théoriques (wpm de l'ordre de 40) et performances mesurées (wpm de 5 pour des débutants et de 10 pour des experts). On constate bien cependant, à l'usage, une amélioration des performances.

T9 présente quelques inconvénients. Certains n'arrivent pas à entrer du texte avec ce système, ils sont gênés par l'affichage qui n'indique pas le mot qu'ils veulent saisir et qui évolue au fil de la saisie du fait de la prédiction en continue. Il semble difficile pour d'autres d'accéder aux autres prédictions quand le mot affiché n'est pas le mot saisi.

Enfin, T9 ne permet pas de saisir des mots hors dictionnaire, dans ce cas, il faut changer de mode pour revenir au mode standard multimap du clavier T12. Les utilisateurs s'y perdent, oublient de passer en mode multimap ou plus souvent de revenir en mode T9. S'il est possible de rajouter des mots personnels dans le dictionnaire (néologisme, nom propre, abréviation...), très peu d'utilisateurs se servent de cette possibilité. On constate enfin que ces différents problèmes imposent à l'utilisateur une charge cognitive assez élevée.

Comme la prédiction linguistique permet de déterminer, à chaque instant, les touches les plus probables, la saisie peut devenir plus efficace en exploitant cette information. Ainsi, il est possible de rendre plus saillantes les touches les plus probables (Magnien, 2005) ou encore de réorganiser le clavier dynamiquement (Schadle *et al.*, 2002) pour minimiser l'amplitude des mouvements (Cf. *infra*, § 5.4).

#### 4 Comparaisons quantitatives

Pour certaines MSDIR, des résultats d'évaluation, le plus souvent quantitatifs, ont été communiqués (tableau 1). La comparaison reste difficile dans la mesure où ces méthodes sont évaluées sur des corpus et avec des protocoles différents, même si des méthodologies et des corpus de tests ont déjà été proposés (MacKenzie et Zhang, 1999).

L'écart entre les performances théoriques et celles mesurées en situation réelle est souvent important, ce qui montre que l'évaluation de ces méthodes avec de vrais utilisateurs reste indispensable.

Le tableau 1 récapitule les performances moyennes des méthodes présentées dans cet article. Ces performances, même en mode expert, sont bien inférieures à celles qui sont calculées dans certaines « compétitions » (speed contests) uniquement sur la saisie d'une seule ou de quelques phrases données. Ces « contests » indiquent par exemple 80 wpm pour le clavier Fitaly, 50 wpm pour Calligrapher ou 30 pour Graffiti 2.

Le tableau 1 précise également si une méthode peut être adaptée pour des personnes handicapées moteurs. En fait, les méthodes d'entrée de texte récentes sont conçues, soit pour permettre une saisie rapide (claviers optimisés), soit une saisie sur dispositifs mobiles ou portés (clavier virtuel ou mini-clavier) (Plouznikoff et Robert, 2004). Pour les claviers optimisés, l'adaptation aux personnes handicapées moteurs peut se faire par un système de défilement sur les touches du clavier (Cf. *infra*, § 5).

Cependant, cette seule technique conduit à une saisie excessivement lente de l'ordre de quelques mots par minute. En revanche, les claviers virtuels et les mini-claviers nécessitent un contrôle gestuel fin et ne sont donc pas adaptables aux personnes handicapées motrices. Ces remarques expliquent en partie les choix de conception faits pour le système d'aide à l'entrée de texte Sibylle présenté dans la deuxième partie de l'article. Sibylle propose une saisie sur un clavier virtuel dynamique à défilement avec prédiction linguistique.

MSDIR	wpm	kspc ou gpc	adaptation au handicap
Qwerty	de 11 à 30	1	clavier à défilement
Fitaly	de 10 à 30		clavier à défilement
Metropolis	43	1	clavier à défilement
Multitap	5	2,03	non : problème moteur
T9	de 5 à 10	1,0072	non : trop de touches
T12	de 5 à 20	de 2 à 5	non : trop de touches
Fastap	7	1	non : trop de touches
Unigesture	de 3 à 4	1	non : problème moteur
Graffiti	6	1,62	non : problème moteur
T-Cube (expert)	de 12 à 20	1	non : problème moteur
QuikWriting (expert)	de 15 à 18	1	non : problème moteur
EdgeWrite	6	1,26	oui : guidage tangible

**Tableau 1.** Comparatif de quelques méthodes de saisie

Comme mentionné précédemment (Cf. *supra*, § 3), les méthodes d'entrée de texte à interface réduite concernent principalement deux situations, les systèmes d'infocommunication portables (handheld) utilisés en mobilité et les communicateurs pour personnes handicapées.

La suite de cet article va décrire une méthode originale d'entrée de texte pour l'aide à la communication des personnes handicapées mise en œuvre dans le système Sibylle.

## 5 Sibylle : une méthode dynamique d'entrée de texte pour l'aide à la communication écrite

Pour les personnes atteintes d'un handicap limitant l'usage de la parole ou affectées d'un handicap physique lourd (tétraplégie), l'aide à la saisie de texte sur ordinateur est un moyen de communication alternatif. Un système d'aide à la saisie, dénommé par la suite « communicateur », est vital dans la mesure où il est parfois le seul moyen d'établir un échange entre la personne handicapée et son entourage.

Un communicateur est un système d'aide composé d'une interface d'entrée adaptée aux facultés motrices (bouton-poussoir, commande par souffle, joystick, souris, etc.) et d'un clavier virtuel remplaçant le clavier physique rendu inadapté. Les messages composés sont restitués par une synthèse vocale.

Comme les possibilités d'interaction offertes par l'interface d'entrée sont extrêmement réduites, typiquement une commande de type « tout ou rien », la sélection des lettres sur le clavier virtuel, encore appelé « clavier simulé », n'est pas possible par désignation directe. La sélection est réalisée par un système de défilement automatique du curseur sur les touches du clavier.

En défilement dit *linéaire*, les touches du clavier sont parcourues une par une et l'utilisateur valide lorsque la touche souhaitée est mise en évidence.

Un autre type de défilement plus rapide est le défilement *ligne/colonne* : le défilement parcourt d'abord les lignes, puis, après une première validation, les touches de la ligne. Bien que plus rapide, le défilement ligne/colonne introduit cependant une deuxième validation qui peut être source d'erreur pour des personnes dont le geste est souvent mal contrôlé.

Quel que soit le type de défilement, l'inconvénient majeur de ces aides est l'extrême lenteur de saisie. Là où l'oral permet un débit de 150 à 200 wpm, ces aides ne permettent qu'une saisie de l'ordre de 5 wpm pour des personnes entraînées. Si ces performances sont modestes, elles sont cependant du même ordre que celles obtenues par des utilisateurs novices ou peu expérimentés avec la plupart des MSDIR (Cf. tableau 1).

Le système Sibylle issu du travail de thèse d'Igor Schadle (2003) (Cf. aussi Schadle et Poirier, 2004), présenté dans la suite de l'article, a pour objectif d'augmenter cette vitesse de saisie et conjointement de minimiser le nombre de saisies en ayant recours à la prédiction linguistique.

Par rapport à la typologie des MSDIR présentée précédemment, Sibylle appartient à la catégorie des entrées par clavier virtuel, plus exactement à une saisie sur multi-claviers virtuels à défilement.

Sibylle est un communicateur développé au VALORIA et utilisé au CMRRF (Centre Mutualiste de Rééducation et Réadaptation Fonctionnelles) de Kerpape. Tout comme d'autres systèmes issus de la recherche tels que VITIPI (Boissière, 2000), HandiAS (Maurel et Le Pévédic, 2001), KeyGlasses (Raynal, 2005), il intègre des connaissances linguistiques. Dans Sibylle, ces connaissances agissent à deux niveaux : d'abord, une prédiction de lettres dont le rôle est de faciliter la sélection des lettres sur le clavier ; puis une prédiction de mots à travers une liste de mots pour économiser le nombre de saisies

### 5.1 Conception de Sibylle

L'objectif scientifique de Sibylle est de tester les apports des modules linguistiques en situation de saisie réelle par des utilisateurs handicapés. La phase de validation par les utilisateurs est une étape nécessaire de l'évaluation (Preece *et al.*, 1994), les résultats théoriques ne se traduisent en effet pas toujours par des gains effectifs. L'utilisation réelle de Sibylle a par ailleurs permis de valider les différentes solutions ergonomiques mises en œuvre.

Une partie de la conception de Sibylle a été réalisée en concertation avec le personnel du CMRRF de Kerpape. Comme ce centre intègre une école qui possède différents systèmes d'aide à la communication, il a été possible de comparer Sibylle à différents logiciels disponibles et de connaître leur utilisation dans divers contextes (utilisation personnelle, utilisation pédagogique en classe, ergothérapie) et leur manipulation par plusieurs personnes handicapées, en particulier des enfants IMC (Infirmes Moteurs Cérébraux). Pour mémoire, au sens strict, les enfants IMC ont un quotient intellectuel supérieur à 70 et représente 0,6 à 0,8% des nouveau-nés.

La conception d'un système d'aide à la communication répond à plusieurs exigences. D'abord, il convient de rappeler que quel que soit l'outil de communication, la composition de texte demande de la part des personnes handicapées un effort important. De plus, ces personnes sont généralement habituées à un système de communication et l'utilisation d'un nouveau système demande une phase d'apprentissage. Cet effort supplémentaire sera d'autant mieux accepté que l'interface est conviviale et intègre des commodités et des fonctionnalités dont l'utilisateur a déjà l'habitude.



Aujourd'hui, les jeunes personnes handicapées ont l'habitude d'utiliser un ordinateur pour avoir une « fenêtre ouverte sur le monde » et communiquer. Elles ont donc une certaine connaissance et des exigences quant aux interfaces proposées. Des expériences passées menées à Kerpape ont bien montré que certains logiciels et systèmes étaient rejetés pour des problèmes d'interface.

## 5.2 Présentation de l'interface

La figure 14 donne une illustration de l'interface de l'application Sibylle (la taille des caractères a été augmentée pour la lisibilité de la figure).



Figure 14. Exemple d'interface de Sibylle

Sur la figure 14, la zone d'édition se trouve en bas à gauche. Les caractères saisis par l'utilisateur apparaissent en noir, tandis que ceux complétés automatiquement à l'aide de la prédiction de mots apparaissent en grisé. L'illustration montre également que le clavier est découpé en plusieurs sous-claviers (ou pavés de touches) regroupant les touches par thème. En haut à gauche se trouve la liste de mots proposés par le module linguistique, au centre le clavier dynamique de lettres. Les claviers sur la droite sont successivement le clavier des commandes générales (correction, impression, enregistrement, vitesse de défilement, etc.), le clavier des touches de déplacement et le pavé numérique.

## 5.3 Saisie dans Sibylle

### *Organisation en sous-claviers*

L'interface présentée est principalement conçue pour le défilement linéaire (le défilement ligne-colonne est une option du système) et la décomposition du clavier en pavés de touches permet de minimiser le nombre de défilements.

L'accès aux différents sous-claviers est réalisé à l'aide de touches de « saut de clavier ». Ces touches sont placées en début de clavier et permettent d'accéder au clavier indiqué par la touche lorsque l'utilisateur clique dessus. La figure 15 montre les cinq premières touches de la première ligne du clavier de lettres.

Les deux touches du début sont des touches de saut de clavier. La première permet d'accéder au clavier général (qui donne lui-même accès aux autres claviers) et

la deuxième à la liste de mots. L'évaluation a montré que ce principe de navigation entre les claviers est simple d'utilisation et immédiatement compréhensible.



Figure 15. Exemple de touches de saut de clavier sur le clavier de lettres

**Saisie sur clavier dynamique avec prédiction de lettres**

La première aide apportée par Sibylle est une prédiction de lettres. Son rôle est d'augmenter la vitesse de saisie en donnant accès en priorité aux lettres estimées comme les plus probables.

Les estimations sont fonction des premières lettres du mot, le pseudo caractère *début de mot* (noté Ø) étant également une information. Après chaque saisie (lettre ou mot), les estimations sont recalculées en fonction du nouveau contexte, l'ordre des lettres est donc modifié sur le clavier après chaque frappe de touche. Le curseur retourne en position initiale (sur la première touche) et le défilement reprend. Une temporisation initiale est programmable pour donner le temps à l'utilisateur de lire les premières lettres.

Lors des tests avec les utilisateurs, il a été constaté que cette temporisation pouvait être remplacée par une ou plusieurs touches fixes en début de clavier, cette constatation justifie le placement fixe de deux touches de saut décrites précédemment en début de clavier de lettres.

Afin d'illustrer le fonctionnement, le tableau 2 présente un exemple d'utilisation avec la saisie du mot « POUR ». La première colonne indique le contexte pour la prédiction, la deuxième la lettre à saisir et la troisième affiche les premières lettres du clavier de lettres (avec le curseur positionné sur la lettre désirée).

Contexte	Lettre à taper	Clavier de lettres
Ø	P	d   p
ØP	O	a r o
ØPO	U	u   s
ØPOU	R	r v s
ØPOUR	espace	_ s n

Tableau 2. Saisie du mot « POUR » sur le clavier de lettres

Sur cet exemple, au début de la saisie de la lettre *p*, le contexte est vide (début de mot). Le clavier affiche donc les lettres les plus probables en tout début de mot : *d*, *l*, *p*, etc. La présence des lettres *d* et *l* s'explique par la fréquence très élevée en français de la préposition *de* et de l'article *le* et de leurs formes dérivées. Une fois la lettre *p* saisie (5<sup>e</sup> touche soit quatre défilements), le curseur retourne en position initiale, l'ordre est actualisé pour afficher les lettres les plus probables après la lettre *p* en début de mot : *a*, *r*, *o*, etc. La saisie se poursuit ainsi pour chaque lettre. Le caractère espace fait partie de la prédiction et est représenté par le caractère *tiret bas* (*\_*).

En moyenne, la lettre désirée se situe au 3<sup>e</sup> rang des lettres présentées (le modèle de prédiction et les résultats de l'évaluation sont donnés dans la suite).

### **Saisie avec la prédiction de mots**

La deuxième aide apportée par Sibylle est une prédiction de mots dont le but est d'accroître la vitesse de composition des messages en économisant le nombre de saisies. La liste des meilleures propositions établie par le système s'affiche dans une liste de mots, les mots proposés tenant compte à la fois des premières lettres du mot et du contexte gauche de la phrase en cours. Ces propositions sont actualisées après chaque saisie.

L'exemple suivant (tableau 3) illustre cette prédiction contextuelle sur le début de phrase « *comme cette chose* ». Pour ces trois mots, la saisie de la première lettre *c* est nécessaire.

Contexte	Mot à saisir	Propositions
c	comme	c' ce <b>comme</b> cette ces
comme c	cette	celui ce celle <b>cette</b> c'
comme cette c	chose	cible chef <b>chose</b> chaîne commission

**Tableau 3.** Exemple de mots proposés en fonction du contexte

En début de phrase, après avoir entré la lettre *c*, le système propose des pronoms (*ce, c'*), des déterminants (*ce, cette, ces*) et la conjonction de subordination *comme*. Si l'on considère que le curseur est en position initiale sur le clavier de lettres, la sélection du mot *comme* est donc réalisée en deux étapes, d'abord accès à la liste de mots (1 défilement, 1 clic) puis sélection du mot (2 défilements, 1 clic).

Une fois la conjonction *comme* affichée, celle-ci disparaît de la liste des propositions, les déterminants et les pronoms sont estimés comme les plus probables. Enfin, après la saisie de *comme cette*, le système privilégie des noms au féminin singulier. Notons que, contrairement aux autres claviers, la sélection dans le clavier de mots n'entraîne pas un retour en position initiale dans la liste des mots mais dans le clavier de lettres. De plus, lorsque la liste des mots est actualisée, une temporisation (proportionnelle à la taille des mots) est ajoutée pour permettre la lecture des propositions.

### **5.4 Analyse du clavier dynamique**

Il est intéressant de s'interroger sur le choix d'un clavier dynamique. Depuis l'origine des interfaces utilisateurs graphiques (GUI), il est posé comme principe que l'utilisateur doit contrôler son écran, le ré-arrangement automatique d'éléments de l'interface est habituellement proscrit.

Ce principe remonte aux premiers GUI conçu par Xerox au Parc pour l'Alto (1973), puis pour le Star (1981). Il est repris par les auteurs de règles et critères ergonomiques. Ben Shneiderman (1987) pose le contrôle comme une règle d'or de la conception des interfaces ; Dominique Scapin (1990) identifie le « contrôle utilisateur » comme un des critères élémentaires ; enfin, plus récemment, Jeff Johnson (2000) rappelle que l'écran appartient à l'utilisateur et qu'il faut préserver l'inertie de l'écran.

Tout ré-arrangement automatique d'éléments d'interaction (widgets) à l'écran est à éviter car il perturbe l'utilisateur. Par exemple, les lignes multiples d'onglets sont toujours rejetées par les utilisateurs, qui ont le sentiment d'une totale désorganisation de l'écran à chaque sélection d'un des onglets.

Plusieurs justifications peuvent être avancées pour expliquer pourquoi, dans le cas de Sibylle, le clavier virtuel dynamique n'est pas rejeté par les utilisateurs.

Alors que, dans le cas des onglets, l'utilisateur ne perçoit pas la logique de la réorganisation ; pour Sibylle, il comprend aisément le sens du re-classement des touches. Il sait qu'elles sont présentées selon leur chance de suivre la lettre précédente. En d'autres termes, le ré-arrangement des touches respecte le principe cognitif de causalité posé par Norman (1988).

Comme chaque onglet est de taille différente selon la longueur de son libellé, le ré-arrangement des lignes modifie grandement l'affichage ; dans le cas du clavier virtuel, chaque touche est de même taille, la structure du clavier est donc ré-affichée sans changement, seul les libellés des touches sont modifiés.

Enfin, le clavier virtuel propose un défilement linéaire (ou ligne-colonne) qui se caractérise par un balayage des touches. A chaque instant le focus d'attention est sur une seule touche (ou une ligne, ou une colonne), l'attention de l'utilisateur est bien canalisée, ainsi le reclassement des touches après chaque saisie perturbe-t-il peu l'utilisateur, qui reste concentré sur la touche active et n'est pas dérangé par la modification de l'affichage.

L'avis des usagers a montré que le clavier dynamique était bien accepté. Il faut cependant indiquer que l'aspect dynamique du clavier a été la cause de l'échec de l'utilisation de Sibylle chez une personne qui a préféré revenir à son système de communication habituel.

## 5.5 Les modules de prédiction

Après avoir expliqué l'utilisation des prédictions dans la saisie, le principe des modèles et leur évaluation sont présentés succinctement ci-après.

### *Prédiction de lettres*

Dans Sibylle, la prédiction de lettres est fondée sur le modèle statistique n-gramme (Schadle *et al.*, 2001, 2002). Celui-ci permet d'estimer la probabilité d'occurrence d'un événement (ici des lettres) en fonction des n-1 derniers événements (dans Sibylle, après expérimentation, en fonction des 4 dernières lettres du mot en cours). L'estimation des paramètres du modèle a été réalisée sur le journal « Le Monde » sur un corpus de plus de cent millions de mots.

L'évaluation théorique a montré que, en moyenne, la lettre désirée apparaît au rang 2,9. Sur les mêmes corpus d'apprentissage et de test, les performances de trois autres modes de sélection ont également été calculées :

- défilement ligne/colonne sur un clavier AZERTY organisé en 3 x 10 correspondant au clavier « traditionnel »,
- défilement linéaire sur un clavier dont les touches sont classées en fonction de leur fréquence d'utilisation hors contexte, appelé clavier « fréquentiel »,
- défilement ligne/colonne sur le même clavier que précédemment.

Clavier	Défilement	Temps d'accès
AZERTY	ligne/colonne	7,3
fréquentiel	linéaire	7,1
fréquentiel	ligne/colonne	4,3
Sibylle	linéaire	2,9

**Tableau 4.** Performances de différents modes de sélection

Le tableau 4 donne les performances obtenues en nombre moyen de défilements pour accéder à la lettre souhaitée en ne tenant compte que des lettres.

Les performances obtenues montrent un gain en temps d'accès significatif pour Sibylle par rapport aux autres modes de sélection. Ce gain est d'autant plus appréciable que la sélection d'une lettre se fait en un seul clic contrairement aux modes de sélection utilisant le défilement ligne/colonne qui nécessite deux validations.

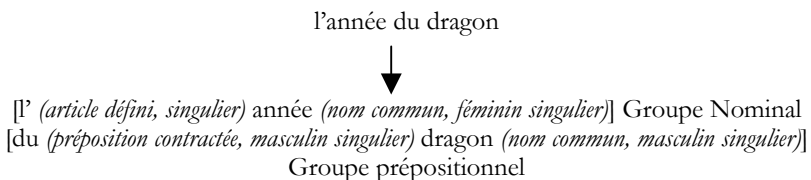
Le gain en efficacité apporté par la prédiction de lettres a été confirmé lors de l'évaluation auprès d'une dizaine de personnes IMC à Kerpape. Cette évaluation, sans la prédiction de mots, a montré une phase d'apprentissage rapide, une baisse des erreurs de frappe grâce au défilement linéaire et une baisse des fautes d'orthographe. Ce dernier objectif est important dans le contexte de l'usage scolaire de Sibylle au CMRRF de Kerpape.

### **Prédiction de mots**

En ce qui concerne la prédiction de mots, Sibylle utilise un modèle de langage avancé (Schadle, 2003, 2004). La prédiction est réalisée en deux étapes : une étape préliminaire d'analyse du contexte gauche de la phrase en cours, puis une étape de prédiction.

De manière schématique, l'étape d'analyse détermine pour chaque mot sa catégorie grammaticale (nom, adjectif, verbe, déterminant, etc.), ainsi que sa flexion (genre, nombre, etc.).

La phrase est ensuite segmentée en constituants comme le montre l'exemple suivant sur le syntagme « l'année du dragon » :



À partir de cette représentation, le système délivre une estimation de la probabilité d'apparition de chaque mot de son lexique comprenant 50 000 entrées. Le modèle permet une gestion partielle des accords et également une présentation de mots en relation avec le contexte. Sur l'exemple précédent, les verbes proposés après l'expression « l'année du dragon » sont *commencer*, *finir*, en relation avec *année*, à la troisième personne du singulier et prioritairement au présent de l'indicatif.

Le modèle a été évalué sur un corpus de test de 50 000 mots extrait du journal « Le Monde » après un apprentissage sur deux millions de mots. Les résultats de l'expérimentation ont montré, pour une liste de 5 mots, un taux d'économie de saisies de l'ordre de 57 %.

Certains systèmes disponibles dans le commerce proposent déjà une liste de mots. Cependant, les propositions sont toujours établies hors contexte. L'intérêt d'adopter un modèle de langage qui prenne en compte le contexte est double.

- Il permet d'accroître la performance (en économie de saisies) du système. Lors de l'évaluation, il a été calculé une économie de saisies de seulement 43 % avec une prédiction hors contexte.
- La prédiction en contexte supprime la présentation de mots incohérents avec le reste de la saisie, qui ont pour inconvénient de perturber l'utilisateur comme le mentionnent Boissière et Dours (2001).

À l'heure actuelle, les prédictions de Sibylle ne s'adaptent pas au vocabulaire particulier de l'utilisateur et aux textes déjà saisis, les propositions sont fondées uniquement sur sa base d'apprentissage du journal « Le Monde ». L'adaptation aux saisies de l'utilisateur est prévue dans les perspectives de Sibylle. Le taux réel d'économie de saisies est nécessairement inférieur au taux calculé sur la base du corpus de test composé d'articles du journal « Le Monde ».

## 5.6 Modularité de Sibylle

La dernière partie concernant Sibylle aborde l'utilisabilité de l'application, ses possibilités d'adaptation et l'analyse des besoins de ses différents usagers.

### *Typologie des utilisateurs*

Au CMRRF de Kerpape, Sibylle s'adresse à trois catégories d'usagers : les personnes handicapées, le personnel hospitalier et le personnel technique. Chacun à son niveau doit pouvoir configurer l'interface.

Les personnes handicapées n'ont accès qu'au réglage de quelques paramètres de la saisie et à quelques commandes, comme l'enregistrement ou l'impression du document. Un réglage important pour ces personnes est la possibilité de changer simplement la vitesse de défilement (vitesse plus, vitesse moins). En effet, la vitesse de défilement dépend de la capacité de l'utilisateur (expérimenté ou non), mais également de son état (réceptivité, fatigue), qui peut changer entre différentes sessions, ou en cours de session. Sur ce dernier point, Sibylle met en œuvre une vitesse de défilement auto-adaptative proposée dans le système EDITH (Pino, 2000).

Le rôle principal attribué au personnel hospitalier (typiquement un médecin, un ergothérapeute ou un infirmier) est de pouvoir configurer de manière simple les principaux paramètres « utilisateur » (couleurs, taille des caractères, temporisations, etc.). Ces paramètres sont peu fréquemment modifiés (typiquement à la création d'un nouvel utilisateur) et peuvent être simplement spécifiés à l'aide de boîtes de dialogue accessibles *via* la barre de menus.

Le dernier type d'usager, le personnel technique informaticien, a accès à l'ensemble des possibilités de configuration de l'application, en particulier aux fichiers de configuration qui permettent de modifier profondément l'interface (liste des claviers, nombre de touches, commandes associées aux touches, texte, image et son des touches, etc.).

### *Configuration de l'interface matérielle*

En dehors des options de configuration précédemment citées, Sibylle permet également de s'adapter à l'interface matérielle d'entrée. Les deux modes d'interaction proposés sont le mode « souris » et le mode « balayage automatique ». En mode « souris », la désignation de la touche est directe, le logiciel assure le suivi du curseur (mise en évidence de la touche sous le curseur souris). En mode « balayage automatique », il est possible d'utiliser le défilement linéaire ou ligne/colonne.

De plus, l'application offre toujours la possibilité de saisir du texte dans la zone d'édition à l'aide du clavier standard, sans avoir à changer de mode.

En raison d'une atteinte du système nerveux central, l'IMC contrôle mal sa motricité et a des difficultés pour commander un mouvement. Il dirige donc

difficilement l'interface matérielle, ce qui entraîne souvent des erreurs de frappe lors d'appuis non désirés sur le bouton-poussoir par exemple. Sibylle permet de corriger ces erreurs en réglant certains paramètres. Ainsi, il est possible de régler la durée minimale d'enfoncement des touches pour éviter les appuis intempestifs, la durée minimale entre deux clics pour éviter des successions de clics non désirées, la vitesse de défilement du balayage. À part le réglage de la vitesse de défilement, ces paramètres doivent être réglés manuellement par un assistant. Il existe toutefois des jeux de tests qui permettent de régler semi-automatiquement ce type de paramètre (Noirhomme-Fraiture *et al.*, 2000).

### ***Évolution de l'interface***

Dans le domaine de la santé et plus particulièrement du handicap, tenir compte de la parole des usagers n'est pas une simple recommandation, mais une réelle obligation prescrite par la Haute Autorité de santé. Dans le cas de Sibylle, il fallait que l'application offre de nombreuses possibilités de configuration pour permettre d'adapter l'interface, depuis le début de son utilisation à Kerpape, en fonction des remarques des utilisateurs.

Ainsi, l'écoute des usagers a conduit aux modifications suivantes pour rendre l'interface plus attractive :

- des modifications d'icônes, quand certaines ont été jugées peu représentatives de la commande ;
- l'ajout d'effets sonores, par exemple la touche de correction a été augmentée d'un son. De même, pour l'utilisation dans le contexte scolaire du CMRRF de Kerpape, une voix a été enregistrée pour la prononciation de chaque lettre, dès qu'une lettre est saisie, la lettre est ainsi prononcée ;
- le déplacement de la liste de mots. À la conception, la liste de mots avait été placée au plus près de la zone d'édition du texte. L'utilisation à Kerpape a montré qu'il est préférable de la positionner à gauche du clavier de lettres. En effet, sur le clavier de lettres ou sur le clavier de mots, lorsqu'une saisie est réalisée, le curseur retourne en position initiale sur le clavier de lettres. La position rapprochée de ces deux claviers permet donc de réduire l'effort de suivi du curseur.

En conclusion, les performances de Sibylle sont dues tout autant à la qualité de ses modules de prédiction, qui améliorent la vitesse de saisie, qu'à ses nombreuses possibilités de configuration. Ces dernières ont permis d'améliorer l'utilisabilité de l'application tout au long de son expérimentation et de proposer une interface satisfaisant pleinement les usagers.

## **6 Conclusion et perspectives**

Cet article a présenté, dans une première partie, une classification des méthodes d'entrée de données à interface réduite et une description des plus représentatives, et dans une seconde partie, la stratégie d'interaction et l'interface du système Sibylle d'aide à la communication pour personnes handicapées.

L'objectif principal de Sibylle est de restaurer partiellement la fonction de communication pour les personnes atteintes d'un handicap physique lourd. C'est par l'intégration de connaissances linguistiques et une interface à clavier dynamique qu'il est possible de compenser le handicap d'interaction et d'atteindre une vitesse de saisie du même ordre que celle obtenue avec des MSDIR très utilisées, comme Graffiti ou Multitap (pour un utilisateur novice ou intermédiaire).

L'évaluation de Sibylle se poursuit pour la prédiction de mots et par l'ajout d'un mécanisme de boîte noire pour recueillir des données quantitatives précises sur la

saisie. Les évolutions à venir porteront sur l'adaptation des données linguistiques aux saisies des utilisateurs.

Plus généralement, les différentes méthodes d'entrée de données à interface réduite, dont celle proposée par Sibylle, qui appartient à la catégorie des entrées par clavier virtuel à défilement, sont des réponses adaptées au problème de l'accessibilité numérique à l'information.

Il faut souligner que l'accessibilité ne concerne pas seulement l'accès à l'information, soit le processus de lecture ou de réception, mais tout autant l'accès à la communication, c'est-à-dire le processus de production.

Aujourd'hui, l'accessibilité tend à être considérée comme un droit universel (tout au moins dans le monde occidentalisé). Être accessible, c'est donc permettre à tous, les personnes mobiles ou handicapées, d'utiliser les systèmes d'infocommunication pour recevoir et produire de l'information dans des contextes choisis et variés.

À l'avenir, il faudra que les nouvelles méthodes d'entrée de données prennent mieux en compte que par le passé l'objectif d'accessibilité qui doit résulter d'une conception pour tous (design for all). L'enjeu est bien que tous les utilisateurs interagissent dans différents contextes d'usage et communiquent ensemble par-delà leurs différences.

## 7 Remerciements

Les travaux de recherche sur Sibylle ont été financés par le conseil régional de Bretagne.

## 8 Références bibliographiques

Anquetil, E., Bouteruche, F. (2004). Conception d'un micro éditeur d'encre électronique et embarquement d'un système de reconnaissance d'écriture manuscrite sur téléphone mobile. In *Conférence Francophone : Mobilité & Ubiquité'04*, 151-157, Nice, juin, France.

Beaudoin-Lafon, M. (2000). Ceci n'est pas un ordinateur - Perspectives sur l'Interaction Homme-Machine. *Technique et Science Informatiques (TSI), Numéro spécial "Informatiques - enjeux, tendances, évolutions"*, R. Jacquart (Ed.), 19(1-2-3), 69-74.

Bellaire. *Clavier CyKey*, site de Bellaire Electronics. Disponible à : <http://www.bellaire.demon.co.uk/>.

Boissière, P. (2000). VITIPI : Un système d'aide à l'écriture basé sur un principe d'auto-apprentissage et adapté à tous les handicaps moteurs. In *Actes de Handicap 2000 « Nouvelles technologies : assistance technique aux handicaps moteur et sensoriel »*, Paris, juin, 81-86.

Boissière, P., Dours, D. (2001). Comment VITIPI un système d'assistance à l'écriture pour les personnes handicapées peut offrir des propriétés intéressantes pour le TALN. In *Actes TALN'2001*, vol. 2 (tutoriels et conférences associées), 183-192, Tours, France.

Card, S., Moran, T., Newell, A. (1980). The keystroke level model for user performance time with interactive systems. *Communications of the ACM*, 23(7), 396-410, ACM Press.



Card, S., Moran, T., Newell, A. (1983). The psychology of human-computer interaction. *Lawrence Erlbaum Associates, Hillsdale, NJ*.

Cockburn, A., Siresena, A. (2003). Evaluating Mobile Text Entry with the Fastap Keypad. In *People and Computers XVII, Proceedings of HCI 2003: Designing for Society*, Bath, UK, September 2003, 77-80.

Dimond, T. L. (1957). Devices for reading handwritten characters. In *Proceedings Eastern Conference Computer*, p. 232-237.

Doignon, P., Kamp, J.-F. (1997). Procédé et dispositif de commande tactile d'équipements électroniques. *Brevet Renault n° FR9715757*.

Engelbart, D. Augmenting Human Intellect: A Conceptual Framework. *SRI Report*. Disponible à : <http://www.bootstrap.org/augdocs/friedewald030402/augmentinghumanintellect/ahi62index.html>

Fitts, P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47(6), 381-391.

Frogpad. *Clavier Frogpad*, site de Frogpad corporation, Disponible à : <http://www.frogpad.com/>

Frishkopf, L., Harmon, L. (1961). Machine Reading of Cursive Script. In C. Cherry (Ed.), *Information Theory (4 th London Symp.)*, 300-316, London, England: Butterworths.

Goldberg, D., Richardson, C. (1993). Touch typing with a stylus. In *Proceedings INTERCHI'93*, ACM Press, 80-87.

Handykey. *Clavier Handykey*, site de Handykey corporation, Disponible à : <http://www.handykey.com/>

In10did. *Clavier in10did*, site de in10did corporation, Disponible à : <http://www.in10did.com/inputnomen.html>

Isokoski, P., Raisamo, R. (2000). Device independent text input: a rationale and an example. In *Proc. of the working conference on advanced visual interfaces AVI'2000*, ACM Press, 76-83.

Johnson, J. GUI (2000). Bloopers : user-interface don'ts and dos for software developers and managers. *Morgan Kaufmann*.

Kamp, J.-F. (1998). *Interaction homme-machine à bord des véhicules automobiles : étude des modalités et des dispositifs d'interaction*. Thèse de l'ENST, février.

Lesh, G.W., Moulton B.J. (2000). A method for optimising single-finger keyboards. In *Proceedings of the RESNA 2000 Annual Conference*, Atlanta, GA. Arlington, VA: RESNA Press.

MacKenzie, I.S., Zhang, S. Z. (1999). The design and evaluation of a high performance soft keyboard. In *Proc. of the ACM Conference on Human Factors in Computing Systems - CHI '99*, ACM Press, 25-31.

MacKenzie, I.S., Kober, H., Smith, D., Jones, T., Skepner, E. (2001). Letterwise: Prefix-based disambiguation for mobile text input. In *Proc. ACM UIST'2001*, ACM Press, 111-120.

MacKenzie, I. S. (2002). KSPC (Keystrokes per Character) as a Characteristic of Text Entry Techniques. In *Proceedings of the Fourth International Symposium on Human Computer Interaction with Mobile Devices*, 195-210, Heidelberg, Germany: Springer-Verlag.

Magnien, L. (2005). *Saisie de texte sur dispositifs nomades : Propositions et évaluations de solutions pour la saisie au stylet sur claviers logiciels*. Thèse de l'université Paul Sabatier, Toulouse, France, décembre.

Maurel, D., Le Pévédic, B. (2001). The syntactic prediction with Token Automata: Application to HandiAS system. *Theoretical Computer Science*, 267, 121-129.

Noirhomme-Fraiture, M., Charrière C., Vanderdonckt J. (2000). A Laboratory of Ergonomic Analyses for Children Suffering form Cerebral Palsy. In J. Vanderdonckt, C. Farenc (Ed.), *Tools for Working With Guidelines TFWWG'2000*, Springer, London, 35-49.

Norman, D. (1988). *The Design of Everyday Things*. Basic Books.

Norman, D. (1998). *The invisible computer*. MIT Press.

Perlin, K. (1998). QuikWriting : continuous stylus-based text entry. In *Proc. of the 11th ACM symposium on User interface software and technology*, ACM Press, 215-216.

Pino, P. (2000). EDITH : Adaptation automatique du temps de défilement aux caractéristiques et intentions de l'utilisateur. In Actes de *Handicap 2000 « Nouvelles technologies : assistance technique aux handicapés moteur et sensoriel »*, Paris, juin, 125-130.

Plouznikoff, N. Robert, J.-M. (2004). Caractéristiques, enjeux et défis de l'informatique portée. In *Proceedings of IHM 2004, International Conference Proceedings Series*, Namur, Belgium, ACM Press, 125-132.

Poirier F., Schadle, I. (2004). Etat de l'art des méthodes de saisie de données sur dispositifs nomades. Typologie des approches. In *Proceedings of IHM 2004, International Conference Proceedings Series*, Namur, Belgium, ACM Press, 133-140.

Preece, J., Rogers, Y., Sharp, H., Benyon, D., Holland, S., Carey T. (1994). *Human-Computer Interaction*. Addison-Wesley ed, Wokingham.

Raskin, J. (2000). *The humane interface*. Addison-Wesley.

Raynal, M. (2005). *Systèmes de saisie de textes pour les personnes handicapées moteur : optimisation, interaction et mesure de l'utilisabilité*. Thèse de l'université Paul Sabatier, Toulouse, France, décembre.

Research In Motion. (2004). *Clavier à saisie SureType*. Site de Research In Motion. Disponible à : <http://www.rim.net/products/suretype/index.shtml>

Sazawal, V., Want, R., Borriello, G. (2002). The Unigesture Approach. In Paterno F. (Ed.), *Proceedings Mobile Human-Computer Interaction : 4th International Symposium, Mobile HCI 2002 (Pisa, Italy, September 18-20)*, Lecture Notes in Computer Science, Springer Verlag, 256 - 270.

Scapin, D.L. (1990). Des critères ergonomiques pour l'évaluation et la conception d'interfaces. *Actes du XXVIème Congrès de la SELF « Méthodologie et outils d'intervention et de recherche en ergonomie »*, Montréal, Canada, octobre.

Schadle, I. Le Pévédic, B. Antoine, J.-Y. Poirier, F. (2001). *Prédiction de lettre pour l'aide à la saisie de texte*. In Actes de JIM 2001, Metz, juillet.

- Schadle, I., Antoine, J.Y., Le Pévédic, B. Poirier, F. (2002). *SibyLettre* : Prédiction de lettre pour la communication assistée. *Revue d'Interaction Homme Machine (RIHM)*, vol. 3, n° 2, 115-133.
- Schadle, I. (2003). *Sibylle* : *Système linguistique d'aide à la communication pour les personnes handicapées*. Thèse de doctorat, Université de Bretagne-Sud.
- Schadle, I., Poirier, F. (2004). Sibylle : un système d'aide à la saisie de texte. In *Proceedings of IHM 2004, International Conference Proceedings Series*, Namur, Belgium, ACM Press, 141-146.
- Schadle, I. Antoine, J.-Y. Le Pévédic, B. Poirier, F. (2004). *SibyMot – Modélisation stochastique du langage intégrant la notion de chunk*. In *Actes de TALN 2004*, Fès, Tunisie, 19-21 avril.
- Shneiderman, B. (1987). *Designing the user interface*. Addison-Wesley.
- Sirisena, A. (2002). *Mobile Text Entry, rapport en informatique, université de Canterbury (NZ)*, Disponible à : [http://www.cosc.canterbury.ac.nz/research/reports/HonsReps/2002/hons\\_0201.pdf](http://www.cosc.canterbury.ac.nz/research/reports/HonsReps/2002/hons_0201.pdf).
- Soukoreff, R. W. et MacKenzie, I. S. (1995). Theoretical upper and lower bounds on typing speed using a stylus and soft keyboard. *Behaviour and Information Technology*, 14 (6), 370-379.
- Stephanidis, C. (2003). Towards Universal Access in the Disappearing Computer Environment. *Upgrade, Special issue "Human-Computer Interaction: Overcoming Barriers"*, vol. 4, n° 1, 53-59.
- TextWare. (1998). TextWare Solutions. *The Fitaly one-finger keyboard*. Disponible à : <http://www.twsolutions.com/domperignon/domperignon2.htm>.
- Venolia, D., Neiberg, F. (1994). T-Cube: a fast, self-disclosing pen-based alphabet. In *Proceedings on Human Factors in Computing System: celebrating interdependence*, ACM Press, 265-270.
- Ward, D.J., Blackwell, A.F., MacKay, D.J.C. (2000). Dasher – a data entry interface using continuous gestures and language models. In *Proceedings of the 13<sup>th</sup> ACM symposium on User Interface Software and Technology*, UIST'00, ACM Press, 129-137.
- Weiser, M. (1991). *The Computer for the 21<sup>st</sup> Century*. Scientific American.
- Wobbrock, J.O., Myers, B.A., Kembel, J.A. (2003). EdgeWrite : A Stylus-Based Text Entry Method Designed for High Accuracy and Stability of Motion. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '03, Vancouver, British Columbia, November 2-5, 2003)*, New York: ACM Press, 61-70.
- Zhai, S. et al. (2000). The Metropolis Keyboard – An Exploration of Quantitative Techniques for Virtual Keyboard Design. In *Proc. of ACM Symposium on User Interface Software and Technology*, San Diego, California, ACM Press, 119-128.
- Zhai, S., Hunter, M., Smith, B. (2002). Performance optimization of virtual keyboards. *Human-Computer Interaction*, 17(2-3), 89-129.