LOI DE fitts

RAPPORT TP

UTBM

GL40

Une image contenant jouet

Description générée automatiquement

Neil FARMER

Table des matières

[1) Contexte du TP 2](#_Toc37496539)

[1.1) Introduction 2](#_Toc37496540)

[1.2) Rappel de la loi de Fitts 2](#_Toc37496541)

[1.3) Cahier des charges 3](#_Toc37496542)

[1.4) Étape de réalisation du TP 3](#_Toc37496543)

[2) Modélisation de la loi de Fitts 4](#_Toc37496544)

[3) Interface 5](#_Toc37496545)

[3.1) Interface de paramétrage du test 5](#_Toc37496546)

[3.2) Interface de Test 6](#_Toc37496547)

[3.3) Interface de résultat 7](#_Toc37496548)

[4) Améliorations possibles et limites 8](#_Toc37496549)

[4.1) Amélioration possible de l’interface 8](#_Toc37496550)

[4.2) Les limites 8](#_Toc37496551)

[a) Limites de l’interface créée 8](#_Toc37496552)

[b) Limites de la formulation de Shannon 8](#_Toc37496553)

[4.3) Difficultés rencontrées 8](#_Toc37496554)

[Bibliographie 9](#_Toc37496555)

# 1) Contexte du TP

## 1.1) Introduction

Dans le cadre de l’UV GL40, nous devons réaliser une interface graphique à l’aide de QTCreator. Cette interface a pour but de comparer le temps réel que prend un utilisateur pour cliquer à l’intérieur d’un cercle à une distance et taille définis aléatoirement. Il faudra ensuite comparer les données récoltées à la loi de Fitts. On pourra pour cela présenter les deux ensembles de données sous forme de courbes et d’indicateurs statistiques.

## 1.2) Rappel de la loi de Fitts

En IHM (Interaction Homme-Machine), la loi de Fitts est un modèle de processeur humain. Elle permet notamment de calculer le temps théorique en secondes pour pointer avec le curseur une cible en fonction de la taille et la distance de la cible.

Une image contenant volant, noir, cerf-volant, oiseau

Description générée automatiquement

Plusieurs formulations de la loi de Fitts existent, chaque fonction possède ses avantages et ses limites. Les principales formulations sont les suivantes :

* La formulation de Fitts :

Où :

* + ID est l'indice de difficulté obtenu (en bits)
  + D est la distance entre le curseur et le centre de la cible ()
  + W est la largeur de la cible ()
* La formulation de Shannon :

Où :

* + D est la distance entre le curseur et le centre de la cible
  + W est la largeur de la cible
  + a et b sont des constantes
* La formulation de Welford :

Où :

* + D est la distance entre le curseur et le centre de la cible
  + W est la largeur de la cible
  + a et b sont des constantes

## 1.3) Cahier des charges

L’objectif sera de :

1. Proposer une interface graphique QT de test permettant le déroulement de l’expérience avec l’utilisateur.
2. Fournir à l’utilisateur les résultats statistiques de l’évaluation comparative sous forme de graphique aisément interprétable.
3. Les options (telles que le nombre de cibles lors du test, la taille des cibles, …) pourront être modifiées.

## 1.4) Étape de réalisation du TP

Pour commencer au mieux le TP, nous devrons choisir une formulation de la loi de Fitts pour le calcul du temps théorique. Nous ferons ensuite 3 interfaces, la première pour choisir ses paramètres de test, une seconde de test, et une troisième qui affiche les résultats obtenus. Ces 3 interfaces devront respecter les règles de l’IHM vu en cours, notamment la structuration de l’écran. Nous verrons pour finir les limites et les améliorations possibles.

# 2) Modélisation de la loi de Fitts

Comme vu précédemment, plusieurs formules sont possibles pour modéliser la loi de Fitts. Nous utiliserons la formulation de Shannon pour calculer la loi de Fitts dans le TP et le rapport :

Où, pour rappel :

* D est la distance entre le curseur et le centre de la cible
* L est la largeur de la cible
* a et b sont des constantes

Le résultat de la formule sera en secondes, nous multiplierons donc la valeur obtenue par 1000 pour la convertir en millisecondes pour correspondre au temps requis par l’utilisateur pour effectuer le mouvement et améliorer la lisibilité des résultats.

Pour commencer, comprenons ce que peuvent être les valeurs a et b. La valeur « a » peut être interprétée comme le temps de réaction de l’utilisateur et la valeur « b » est l’accélération du mouvement du point initial vers la cible.

On estimera tout d’abord la valeur de « a » à environ 284 ms. Cette valeur est le temps de réaction moyen des utilisateurs de « Human Benchmark », qui a recueilli des données sur le temps de réaction moyen sur plus de 81 millions de test[[1]](#footnote-1).

On estime ensuite de manière empirique la valeur de « b » à environs 0,101 ms. La valeur a été définie après plusieurs tests avec la valeur de « a » définie à 284 ms. Pour estimer au mieux « b », on réalise plusieurs séries de tests et on essaye de diminuer au maximum l’écart moyen et l’écart type moyen entre les valeurs théorique calculées par la loi de Fitts et les valeurs des tests.

La valeur de L sera le diamètre du cercle pour que, lorsque la souris se trouve à l’intérieur du cercle quand ce dernier s’affiche, la valeur de soit inférieure à 1. Cela donnera alors :

Le fait que la valeur soit égale à 0 indique qu’aucun micromouvement n’est nécessaire pour se rendre à l’intérieur du cercle. Seul le temps de réaction sera pris en compte dans la formule de la loi de Fitts.

La distance D séparant le point de départ ( au centre du cercle ( sera estimée grâce à la formule suivante (dérivé de la formule de Pythagore) :

Il ne reste plus qu’à stocker puis récupérer les valeurs des positions de la souris au départ et du centre du cercle pour faire les calculs.

# 3) Interface

## 3.1) Interface de paramétrage du test

L’objectif de cette interface est de pouvoir modifier les paramètres du test. On doit pouvoir modifier le nombre de cibles pour un test, la taille minimum et maximum d’une cible. Une fois les paramètres choisis, on doit pouvoir lancer la phase de test avec un respect des paramètres entrés.

Pour commencer, ces paramètres doivent pouvoir être changés facilement les uns à la suite des autres. On choisira donc de regrouper dans un GroupBox les éléments liés au test. Cela aura pour but d’avoir un diagramme de transition le plus faible possible, pour faciliter les modifications.

On va définir un intervalle pour chaque option. Ainsi, le nombre de cibles est un entier entre 1 et 99 compris, la taille minimum d’une cible est comprise entre 1 et 100 et la taille maximum entre 101 et 200.

On peut aussi réfléchir aux paramètres qui pourraient être ajoutés. On pourrait faire plusieurs interfaces de test : la première pour guider au mieux l’utilisateur testé, la seconde qui donnerait des détails supplémentaires et une dernière qui serait vide. De plus, étant donné que la taille et la position sont choisies aléatoirement, on pourrait proposer à l’utilisateur de choisir un seed pour répéter plusieurs fois le même test sur des individus différents. De ce fait, l’objectif est d’enlever une partie aléatoire qui peut expliquer des variations d’écart d’un utilisateur à un autre (on peut par exemple imaginer qu’une cible de petite taille dans un coin de l’écran est plus difficile à repérer et à atteindre).

On choisira de regrouper les différentes interfaces dans un même GroupBox. L’interface peut être choisie avec des boutons radio. Pour le seed, on le mettra dans le GroupBox des options du test. On ajoute aussi à coté de ce champ un check box, pour générer du pseudo aléatoire au lieu du seed.

Pour finir, un rappel de la loi de Fitts (formulation de Shannon) à l’utilisateur peut aussi être pertinent. Les paramètres sont enregistrés pour qu’ils soient identiques à l’exécution suivante grâce à QSettings.

L’interface finale est la suivante :

Une image contenant capture d’écran

Description générée automatiquement

## 3.2) Interface de Test

Bien qu’il y ait 3 interfaces possibles de test, les différences sont très mineures. On s’intéressera ici à une interface de test type, que les 3 interfaces ont en commun.

Avant de construire une interface de test, il faut transmettre les paramètres de test entrés précédemment vers la nouvelle fenêtre. Pour cela, on va modifier le constructeur généré par QT pour y faire transiter les informations rentrées dans la fenêtre précédente.

Pour le test, on lance le tracking de la souris avec QMouseEvent. On crée la fonction mousePressEvent qui s’activera à chaque clic de souris. On pourra alors tester si le curseur est dans le cercle ou non. Si le curseur est bien dans le cercle, on pourra alors arrêter le timer lancé précédemment, mettre la valeur en mémoire, stocker la position de la souris, lancer un nouveau timer et dessiner un nouveau cercle.

Pour les cercles, on passera par QPaint et la fonction paintEvent. On choisira aléatoirement les coordonnées du centre du cercle sur l’écran ainsi que le diamètre du cercle qui sera compris entre les valeurs entrées par l’utilisateur précédemment.

A noter que l’interface de test se met automatiquement sur toute la surface de l’écran pour avoir le meilleur test possible. En effet, cela permet d’avoir des distances moyennes plus grandes, ce qui sera plus intéressant dans les données car certaines distances seront faibles voire nulles alors que d’autre seront à l’opposé de l’écran.

On transmettra les données récoltées vers la fenêtre résultat par la même méthode que lorsque les paramètres ont été reçus, à savoir en modifiant le constructeur.

L’interface de test est visuellement proche d’une fenêtre vide avec un cercle rouge à une position et taille aléatoire :

Une image contenant oiseau

Description générée automatiquement

## 3.3) Interface de résultat

Pour la fenêtre de résultat, il sera possible d’ajuster manuellement les valeurs de « a » et « b » pour que les valeurs de la loi de Fitts correspondent au mieux aux valeurs recueillies. Par défaut, ces dernières sont aux valeurs définies plus haut (lien : Modélisation de la loi de Fitts)

Deux moyens sont disponibles pour visualiser les valeurs :

1/ Une table, où sont affichés le diamètre du cercle, la distance du curseur au centre du cercle, le temps mis par l’utilisateur et le temps théorique prédit par la formule dans la première partie.

2/ Un graphique (fait avec QCustomPlot) qui affiche le temps théorique (loi de Fitts) et le temps réel mis par l’utilisateur, les deux en millisecondes.

Des données statistiques basiques sont fournies, tel que l’écart moyen, l’écart type, l’écart maximum et minimum. Il est aussi possible de venir extraire les données dans un fichier csv ou de recommencer le test.

Une image contenant carte, texte

Description générée automatiquement

# 4) Améliorations possibles et limites

## 4.1) Amélioration possible de l’interface

Comme on a pu le voir, différentes formulations de la loi de Fitts existent, on aurait pu donc laisser à l’utilisateur le choix de la formule utilisée. Aussi, la possibilité d’afficher le temps en fonction de l’indice de difficulté (via de la régression linéaire) aurait pu être pertinent.

Les interfaces de test auraient pu être plus différentes et donc plus poussées, avec une fenêtre différente appelée selon l’interface choisie.

## 4.2) Les limites

### a) Limites de l’interface créée

Pour commencer, toutes les valeurs sont arrondies à l’entier le plus proche (en millisecondes), ce qui entraîne une légère perte de précision de moins de 1ms.

De plus, lorsque l’interface de test n’est pas vide, les ronds rouges peuvent se mettre sur le texte, ce qui augmentera alors artificiellement le temps mis par l’utilisateur. Aussi, quand l’interface n’est pas vide, le texte fait un élément visuel supplémentaire qui peut à nouveau augmenter le temps de l’utilisateur.

### b) Limites de la formulation de Shannon

« Les Analyse de Hoffman ont mené à la conclusion que la formulation de Shannon pour la loi de Fitts, index de difficulté, était erroné dû notamment au fait que les mouvements ne sont pas un signal continu ». (MacKenzie, 2013)

Des limites sont à apporter à chacune des formulations de la loi de Fitts. Aucune formulation ne permettra de donner une idée précise du temps mis réellement par l’utilisateur. Ces formules permettent de donner une approximation plus ou moins juste selon les paramètres et les individus.

## 4.3) Difficultés rencontrées

Pour réaliser le TP, il est nécessaire de lire beaucoup de documentation Qt, notamment QMouse et QPainter, qui sont uniquement disponibles en anglais. Cela peut s’avérer assez long pour en sortir toutes les subtilités et options qui nous seront utiles.

Il faut de même lire beaucoup de choses sur la loi de Fitts (majoritairement en anglais) pour avoir un maximum de détails sur son utilisation et ses limites.

Ces difficultés restent très relatives étant donné que la documentation disponible est très fournie.

# Bibliographie

MacKenzie, I. S. (2013). *A Note on the Validity of the Shannon Formulation for Fitts' Index of Difficulty*. Récupéré sur https://www.yorku.ca/mack/ojas2013.html

Vella, F. (2008). *Modèles psychophysiques d’atteintes de cibles pour les personnes souffrant de troubles neuromusculaires.* Récupéré sur Chapitre 2 : Modélisation des performances: https://tel.archives-ouvertes.fr/tel-00423539/document

1. Source : <https://www.humanbenchmark.com/tests/reactiontime/statistics> [↑](#footnote-ref-1)