



**Université  
de Lille**  
1 SCIENCES  
ET TECHNOLOGIES

Université de Lille

**MASTER 2 INFORMATIQUE**  
Premier semestre



**NIHM**

**Bubble cursor**

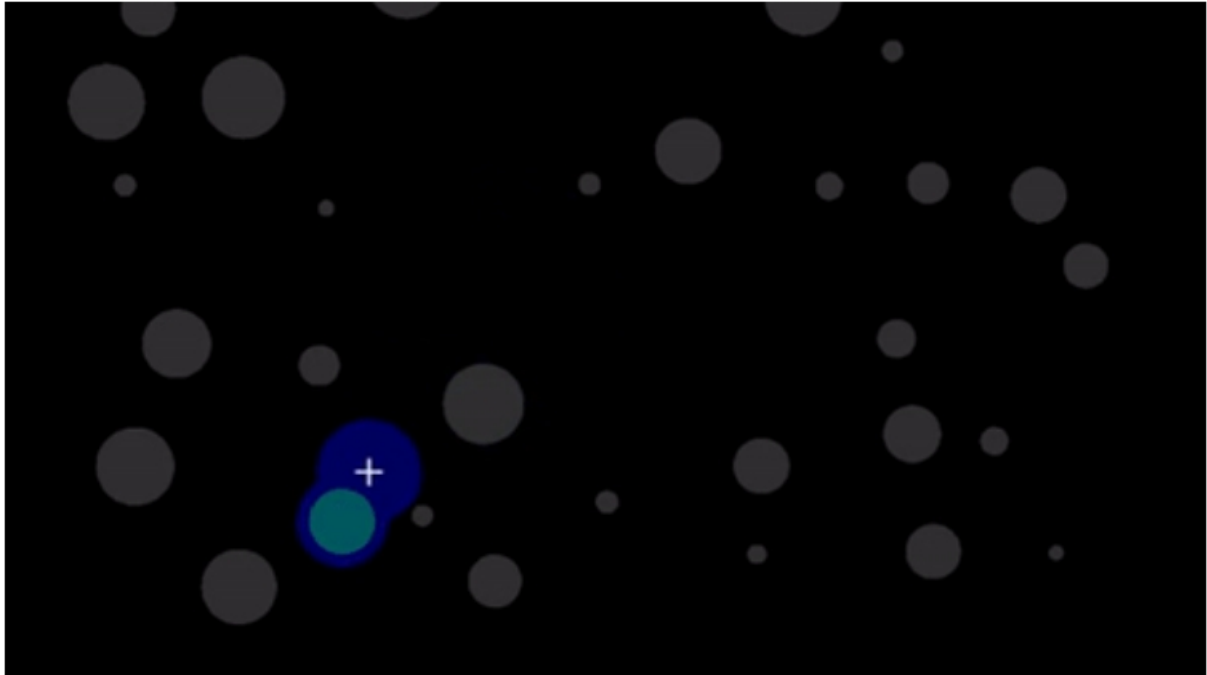
**BARCHID Sami**

Année académique 2019-2020

# Introduction

---

Ce rapport de TP vise à effectuer une évaluation d'une nouvelle technique de pointage : le « bubble cursor » (BC, en abrégé). Le BC est une technique de pointage qui consiste à trouver le point le plus proche du curseur et le rend cliquable.



*Exemple de Bubble Cursor*

Nous cherchons à étudier et comparer le BC à la technique de pointage classique. La question à laquelle nous voulons répondre est : « **Le Bubble Cursor apporte-t-il des avantages en terme de performances (rapidité et réduction d'erreurs), comparé à du pointage classique ? Et si oui, dans quel contexte ?** ».

Pour ce faire, nous menons une expérience sur plusieurs utilisateurs. Le protocole expérimental est détaillé ici. Ce rapport est ainsi divisé en trois parties :

- **Description générale** : description générale de la tâche à réaliser et de l'organisation de l'expérience.

- **Hypothèses** : présentation des hypothèses émises auxquelles notre expérience doit répondre.
- **Protocole expérimental** : description des valeurs des paramètres, ordonnancement des trials et du setup de l'expérience.

# Description générale

---

Pendant l'expérience, un participant va tester une série de trials à réaliser où on lui demande simplement de cliquer sur un cube « cible » alors que celui-ci est à une distance définie du curseur et est entouré d'autres cubes « distracteurs » (organisés en grille de 10cm autour du cube cible). Le participant devra effectuer ce genre de trial avec le Bubble Cursor et avec le curseur classique afin que l'on puisse comparer les performances des deux techniques.

## Autre technique d'interaction

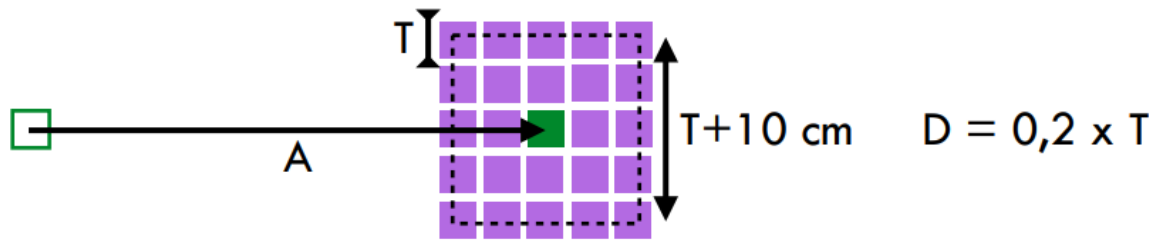
En plus du BC, on introduit le « **Line Cursor** » (LC) qui est une variante du BC. Contrairement au BC qui trace un cercle centré sur le curseur et dont le rayon va jusqu'à l'objet le plus proche, le LC trace une ligne qui va du curseur jusqu'à l'objet le plus proche (sur lequel on peut cliquer même à distance, comme pour le BC). Dans notre expérience, on désirerait donc aussi tester les avantages du LC par rapport au BC.

## Formalisation des paramètres

On dispose de trois paramètres à varier :

- **Distance (A)** : distance, en mm, entre la position du curseur au début du trial.
- **Taille (T)** : taille, en mm, du côté d'un carré cliquable (cible ou distracteur).
- **Densité (D)** : distance latérale, en % de T, entre les distracteurs dans le carré de cubes.

L'image ci-dessous montre un trial quelconque avec l'illustration de ces paramètres :



*Illustration des paramètres d'un trial*

## Mesures

Lors d'une expérience menée sur un participant, celui-ci devra enchaîner l'exécution de trials avec une variation des trois paramètres vus. Des données seront mesurées à chaque trial :

- **Temps de pointage** : temps (en ms) que le participant a mis pour sélectionner le carré cible.
- **Erreurs** : nombre de fois que le participant a cliqué autre part que dans le carré cible.

À cela viennent s'ajouter des mesures de « logs » dont on s'occupera une autre fois.

# Hypothèses

---

Avec les trials d'expériences, les paramètres et les mesures que l'on a définies, on peut maintenant essayer d'élaborer des hypothèses sur ce qui sera observé pendant notre expérience. Les hypothèses émises sont les suivantes :

- Quand la configuration est à faible densité, BC est plus rapide car on prend moins de temps pour viser, étant donné que la distance à parcourir pour sélectionner en BC sera plus petite.
- Quand la configuration est dense, BC est plus lent et fait surtout plus d'erreurs car ça devient très compliqué de viser la bonne cible, on aura besoin d'être plus précis qu'en classique.
- Quand la taille d'un carré est faible, BC fait moins d'erreurs et est plus rapide que le pointage classique car on doit moins se concentrer pour viser (et donc prendre moins de temps, moins de risques).
- Quand la taille d'un carré est grande, le pointage classique sera plus rapide car on peut se permettre de faire de grands gestes rapides et approximatifs pour toucher la cible.
- La distance de l'objet cible ne change pas le résultat final d'une observation. C'est-à-dire que, dans une configuration où BC était meilleur que le pointage classique, BC ne deviendra pas plus mauvais que le pointage dans une configuration parce que la distance a changé. Ceci peut s'expliquer par le fait que la distance entre le curseur et la zone où se trouvent les distracteurs se parcourt très rapidement et le travail de visée entre les distracteurs est le moment où on perd le plus de temps. Ce qui importe est donc juste la densité et la taille des cubes.
- Line Cursor est toujours plus performant que BC car LC montre le chemin le plus court entre le point sélectionné et le curseur. Ce chemin montre d'une manière plus directe ce point sélectionné, ce qui avertit plus rapidement et plus explicitement l'utilisateur sur ce qui est sélectionné. On devient alors plus précis et plus rapide.

On définit maintenant le protocole qui va permettre de vérifier nos hypothèses.

# Protocole expérimental

---

## Setup

Le support sur lequel est mené les expériences est le suivant :

- Ordinateur portable Dell G3 17
- Écran 17.3 pouces (34.5 x 25.9 cm)
- OS Windows 10
- Souris APM sans fil

## Valeurs des paramètres

Les trois paramètres modifiables pourront prendre les valeurs suivantes :

- **Distance (A)** : 200, 100, 50 (en mm)
  - Les distances longues et les distances courtes vont servir à vérifier notre hypothèse que la distance ne modifie pas l'ordre de performance entre les trois techniques étudiées
- **Taille (T)** : 20, 10, 5, 2.5 (en mm)
  - Les tailles très petites permettent de vérifier notre hypothèse qui stipule que le pointage classique sera généralement moins performant que le BC avec des carrés tout petits.
  - Les tailles grandes permettent de vérifier l'hypothèse qui stipule que le pointage classique sera plus rapide en cas de grandes tailles.
- **Densité (D)** : 200, 100, 20 (en %)



- La densité variable permet de vérifier notre hypothèse qui stipule que le BC est moins bon quand on a une configuration très dense, mais meilleur quand la configuration est à faible densité.

## Ordonnancement

Pour chaque participant à l'épreuve, on fait tester toutes les configurations possibles suivant les valeurs des paramètres définies au-dessus. Ces configurations de paramètres seront testées avec les trois techniques (pointage classique, BC et LC).

Afin d'éviter le bruit qui peut apparaître lors d'un trial (personne qui « trébuche » sur son clavier, qui est dérangée par quelque chose, etc), **on fait répéter 5 fois la même action au cours d'un trial**. Ces 5 répétitions s'appellent des « blocs ».

L'ordonnancement des trials (suivant la notation vue en cours) est le suivant :

- $P <$ 
  - 5, technique, A, T, D ?

En bref, pour chaque participant, tous les blocs de tous les trials possibles sont répartis aléatoirement. On obtient donc un ordonnancement purement aléatoire où c'est très peu probable qu'on puisse faire deux fois la même chose.

Le but de cet ordonnancement est d'éviter que le participant s'habitue rapidement à réaliser une même tâche. Dans cet expérience, on limite au maximum un effet de « réflexe » pour bien mesurer le processus de visée en lui-même.