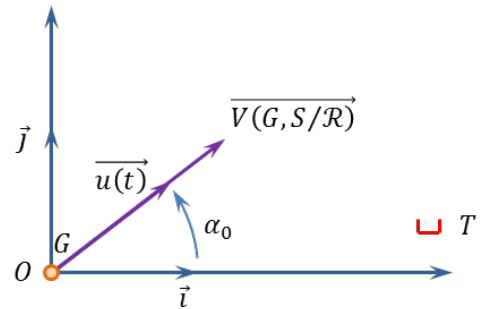


# CI 3 : INGÉNIERIE NUMÉRIQUE & SIMULATION

## CHAPITRE 2 – PROBLÈMES STATIONNAIRES RÉSOLUTION NUMÉRIQUE DE L'ÉQUATION $f(x) = 0$



En première approximation, sans prendre en compte le mouvement de rotation de la balle et les divers effets aérodynamiques, quel doit être l'angle à l'impact du club avec la balle et la vitesse initiale de la balle pour que la balle aille directement dans le trou (sans rebond) ?

**Savoir** Problème stationnaire à une dimension, linéaire ou non conduisant à la résolution approchée d'une équation algébrique. Méthode de dichotomie, méthode de Newton.

1	Introduction .....	2
1.1	Mise en situation .....	2
1.2	Définitions .....	2
1.3	Représentation graphique .....	3
1.4	Critères de convergence .....	3
2	Méthode de dichotomie .....	4
2.1	Principe .....	4
2.2	Algorithme de dichotomie .....	4
2.3	Étude théorique de l'algorithme .....	5
2.4	Application sur l'exemple .....	5
2.5	Méthode de Lagrange – Méthode des cordes .....	6
3	Méthode de Newton .....	7
3.1	Principe .....	7
3.2	Algorithme de dichotomie .....	7
3.3	Étude théorique de l'algorithme .....	7
3.4	Application sur l'exemple .....	7

# 1 Introduction

## 1.1 Mise en situation

### Recherche l'équation paramétrique de la position de la balle de golf.

En modélisant la balle de golf  $S$  comme un solide dont la masse  $m$  est considérée concentrée en son centre d'inertie  $G$ . On considère qu'en première approximation la balle est soumise à son propre poids. En l'isolant et en lui appliquant le théorème de la résultante dynamique, on a :

$$\sum \overrightarrow{F_{\text{ext} \rightarrow \text{balle}}} = m \overrightarrow{\Gamma(G \in S/\mathcal{R})}$$

L'action de pesanteur de la balle est donnée par  $\overrightarrow{F_{\text{pesanteur} \rightarrow \text{balle}}} = -mg \vec{j}$ .

On a donc :

$$\overrightarrow{\Gamma(G \in S/\mathcal{R})} = \left[ \frac{d^2 \overrightarrow{OG}}{dt^2} \right]_{\mathcal{R}} = \begin{bmatrix} x''(t) \\ y''(t) \\ z''(t) \end{bmatrix}_{\mathcal{R}} = \begin{bmatrix} 0 \\ -g \\ 0 \end{bmatrix}_{\mathcal{R}}$$

En intégrant successivement  $x''(t)$  et  $y''(t)$ , on a :

$$\begin{cases} x'(t) = V_0 \cos \alpha_0 \\ y'(t) = -gt + V_0 \sin \alpha_0 \end{cases} \quad \begin{cases} x(t) = V_0 \cos \alpha_0 t \\ y(t) = -\frac{1}{2}gt^2 + V_0 \sin \alpha_0 t \end{cases}$$

### Mise en équation du problème.

On note  $\overrightarrow{OT} = x_T \vec{i} + y_T \vec{j}$  la position du trou  $T$ .

A l'instant  $t_f$  où la balle atteint sa position, on a :

$$\begin{cases} x_T = V_0 \cos \alpha_0 t_f \\ y_T = -\frac{1}{2}gt_f^2 + V_0 \sin \alpha_0 t_f \end{cases} \Rightarrow y_T = -\frac{1}{2}g \left( \frac{x_T}{V_0 \cos \alpha_0} \right)^2 + V_0 \sin \alpha_0 \frac{x_T}{V_0 \cos \alpha_0}$$

$$\Leftrightarrow y_T + \frac{1}{2}g \frac{x_T^2}{V_0^2 \cos^2 \alpha_0} - x_T \tan \alpha_0 = 0$$

Considérant que le golfeur a un swing régulier et que sa vitesse d'impact est constante. On cherche l'angle  $\alpha_0$  qui permettra de choisir le club le mieux adapté.

## 1.2 Définitions

Définition

### Problème stationnaire

On appelle problème stationnaire un problème dont l'énoncé reste invariant au cours du temps.

Définition

### Application linéaire

Soit  $f$  une application de  $E$  dans  $F$ ,  $E$  et  $F$  étant deux espaces vectoriels. Soit  $K$  un corps commutatif.  $f$  est une application linéaire si :

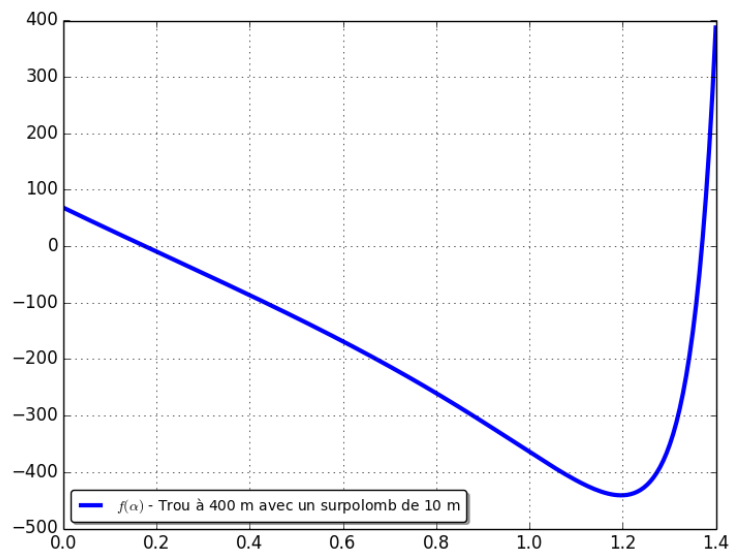
- $\forall x \in E, \forall y \in E, f(x + y) = f(x) + f(y)$ ;
- $\forall \lambda \in K, \forall x \in E, f(\lambda x) = \lambda \cdot f(x)$ .

Exemple

Soient  $(a, b)$  deux réels et  $f$  une application de  $\mathbb{R}$  dans  $\mathbb{R}$  telle que  $f : x \mapsto ax + b$ .  $f$  est une application linéaire.  
 $g : x \mapsto \cos x$  n'est pas une application linéaire.

### 1.3 Représentation graphique

Afin de résoudre un problème, il est possible d'avoir une représentation de la courbe. Ainsi, en traçant  $f(\alpha_0)$  dans le cas du swing de golf, il est possible de répondre au problème en identifiant les points où la courbe sectionne l'axe des abscisses.



Remarque

A priori il n'est pas possible de connaître le nombre de solutions que comporte notre problème. Dans notre cas, deux solutions sont possibles. Dans le cas général, il faudra faire attention

Attention

Il est important de faire attention aux représentations graphiques : selon la discrétisation de la courbe affichée, il est possible que des intersections entre la courbe et l'axe des abscisses n'apparaissent pas alors que mathématiquement ces intersections existent.

### 1.4 Critères de convergence

Numériquement, il n'est jamais possible de trouver la solution exacte à l'équation. Ainsi, sur un intervalle  $[a, b]$ , il est impossible de trouver  $c$  tel que  $f(c) = 0$ .

Il sera donc nécessaire de définir un critère de convergence, c'est à dire une valeur  $\varepsilon$  telle que  $|f(c)| < \varepsilon$ .

On pourra par exemple prendre  $\varepsilon$  de l'ordre de  $10^{-9}$ .

## 2 Méthode de dichotomie

### 2.1 Principe

Théorème

#### Théorème des valeurs intermédiaires

Soit  $f$  une fonction définie et continue sur l'intervalle  $[a, b]$  à valeur dans  $\mathbb{R}$ . Pour tout  $u \in [f(a), f(b)]$ , il existe au moins un réel  $c \in [a, b]$  tel que  $f(c) = u$ .

En particulier (Théorème de Bolzano), si  $f(a)$  et  $f(b)$  sont de signes différents, il existe au moins un réel  $c$  tel que  $f(c) = 0$ .

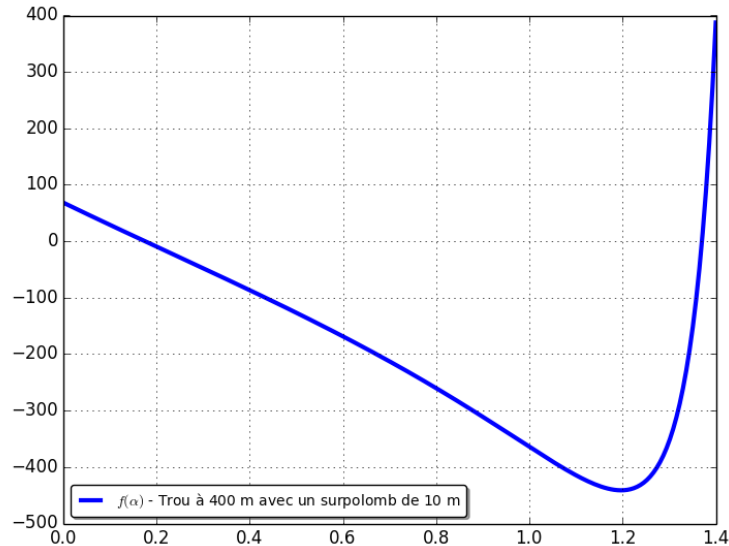
Ainsi, pour une fonction donnée définie sur un intervalle donné, le but de l'algorithme de dichotomie va être de découper en 2 l'intervalle  $[a, b]$  en deux, afin d'y trouver la solution. Par divisions successives de l'intervalle, on convergera vers la solution.

Remarque

#### Tester le signe de $f(a)$ et $f(b)$ .

Il existe plusieurs méthodes pour tester si  $f(a)$  et  $f(b)$  sont de signes différents. Si on ne se préoccupe pas de savoir la relation d'ordre entre  $f(a)$  et  $f(b)$ , un test efficace consiste en un test du signe de  $f(a) \cdot f(b)$ .

### Interprétation graphique



### 2.2 Algorithme de dichotomie

L'algorithme de dichotomie est le suivant :

Pseudo Code

```

Début Fonction
Données :  $f, a, b, \varepsilon$ 
 $g \leftarrow a$ 
 $d \leftarrow b$ 
 $f_g \leftarrow f(g)$ 
 $f_d \leftarrow f(d)$ 
tant que  $(d - g) > 2\varepsilon$  faire
     $m \leftarrow (g + d)/2$ 
     $f_m \leftarrow f(m)$ 
    si  $f_g \cdot f_m \leq 0$  alors
         $d \leftarrow m$ 
         $f_d \leftarrow f_m$ 
    sinon
         $g \leftarrow m$ 
         $f_g \leftarrow f_m$ 
    fin
fin
retourner  $(g + d)/2$ 
Fin
    
```

Exemple

*Implémenter cet algorithme en Python*

On veillera à vérifier que l'équation comporte initialement au moins une solution. On étudiera, a posteriori, la possibilité d'existence de deux solutions.

## 2.3 Étude théorique de l'algorithme

### 2.3.1 Variant de boucle

Montrons que  $d - g$  est un variant de boucle.

Tout d'abord, la quantité  $d - g$  reste positive tout au long de l'algorithme. En effet, suivant le cas, après une itération, si  $d_i - g_i > 0$  on a :

- dans un cas,  $d_{i+1} \leftarrow (g_i + d_i)/2$  et  $g_{i+1} = g_i$  en conséquence,  $d_{i+1} - g_{i+1} = (d_i - g_i)/2 > 0$  ;
- dans l'autre cas,  $g_{i+1} \leftarrow (g_i + d_i)/2$  et  $d_{i+1} = d_i$  en conséquence,  $d_{i+1} - g_{i+1} = (d_i - g_i)/2 > 0$ .

En conséquence, à chaque itération,  $d - g$  est toujours positif.

Par ailleurs, la quantité  $d - g$  décroît tout au long de l'algorithme. En effet, à chaque itération  $i$ ,  $d - g = \frac{b - a}{2^{i-1}}$ . Ainsi, il existera un entier  $i$  tel que  $d - g > 2\varepsilon$ .

$d - g$  est donc un variant de boucle.

### 2.3.2 Invariant de boucle

Montrons que  $f(d) \cdot f(g) \leq 0$  est un invariant de boucle.

On rappelle qu'il faut :

1. définir les préconditions (état des variables avant d'entrer dans la boucle) ;
2. définir un invariant de boucle ;
3. prouver que l'invariant de boucle est vrai ;
4. montrer la terminaison du programme ;
5. montrer qu'en sortie de boucle, la condition reste vraie.

### 2.3.3 Complexité algorithmique

La boucle while s'exécute jusqu'à ce que  $\frac{b-a}{2^n} < 2\varepsilon$ . En conséquence, la boucle s'exécutera suivant la condition suivante :

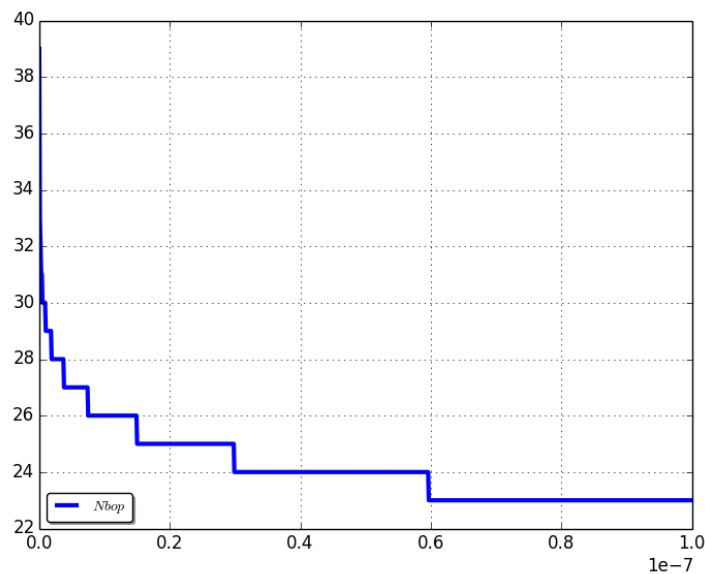
$$\frac{b-a}{2^n} < 2\varepsilon \iff \frac{b-a}{2\varepsilon} < 2^n \iff n > \frac{\ln\left(\frac{b-a}{2\varepsilon}\right)}{\ln 2}$$

La complexité de l'algorithme de recherche dichotomique est donc en  $\mathcal{O}(\log(n))$ .

### 2.4 Application sur l'exemple

Le trou étant d'un diamètre de 108 mm et la balle ayant un diamètre de 42,67 mm. L'erreur admissible sur l'impact de la balle est donc de  $((108 - 42,67)/2)/1000 \simeq 0,032$  m. Dans ce cas, obtenir une erreur inférieure pourrait accroître les calculs sans pour autant apporter une réelle plus value du point de vue du golfeur.

Cependant, on peut tout de même observer le nombre d'opérations en fonction de l'erreur demandée :



### 2.5 Méthode de Lagrange – Méthode des cordes

La méthode de Lagrange diffère de la méthode de dichotomie par le fait que l'intervalle n'est pas divisé en 2 parts égales. Dans cette méthode, on cherche  $c$ , intersection de l'axe des abscisses et de la droite passant par les points  $(a, f(a))$  et  $(b, f(b))$ .

1. Donner une interprétation graphique de cette méthode.
2. Donner l'algorithme permettant de résoudre l'équation  $f(x) = 0$  en utilisant la cette méthode.

## 3 Méthode de Newton

### 3.1 Principe

### 3.2 Algorithme de dichotomie

### 3.3 Étude théorique de l'algorithme

#### 3.3.1 Variant de boucle

#### 3.3.2 Invariant de boucle

#### 3.3.3 Complexité algorithmique

### 3.4 Application sur l'exemple

## Références

[1] Germain Gondor, Problèmes stationnaires à une dimension du type  $f(x)=0$ , Lycée Carnot, Dijon. UPSTI.