

Métaheuristiques pour l'optimisation d'un clavier

Projet d'Intelligence Artificielle 2A



Arthur Cappellina
Erwan Vaillant

Table of contents

01

Modélisation

02

Recuit Simulé

03

Algorithme génétique

04

Recherche Tabou

1. Modélisation

Définition d'un score à minimiser

Score = 0

Pour chaque lettre x:

 Pour chaque lettre y:

 Score += freq[x][y]*dist(x,y)

Paramètres: liste des 40 touches du clavier au format 1D

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

Exemple de configuration

2. Recuit simulé

Algorithme inspiré du phénomène de diffusion thermique:

- État initial: les lettres dans l'ordre alphabétique, puis des touches vides
- On définit l'énergie => fonction objectif
- Température donnée
- Itérations:
 - Déplacement élémentaire
 - Vérification de l'évolution d'énergie engendrée
=> acceptation avec une probabilité :

$$e^{-\frac{\Delta E}{T}}$$

3. Algorithme génétique

Algorithme inspiré du phénomène d'évolution naturelle:

Génotype d'un individu = une configuration de clavier (liste 1D de 40 éléments)

Fitness = fonction objectif

Croisement choisi:

On choisit un point dans la liste à partir duquel on échange la suite des touches

Mutation (avec une probabilité p):

On inverse deux touches aléatoires dans le génotype d'un individu

| | | |
|-----|---|-----|
| A | | B |
| Z | | J |
| T | ↔ | P |
| Y | ↔ | D |
| ... | | ... |
| ... | | ... |
| M | ↔ | I |

4. Recherche Tabou

Algorithme basé sur la recherche itérative de voisins minimisant la fonction objectifs:

Voisins obtenus successivement en inversant deux touches

Itération sur les voisins pour trouver une meilleure configuration de clavier

Interface de notre projet

Keyboard Layout Optimizer

Recuit Simulé

| | | | | | | | | | |
|--|---|---|---|---|---|---|---|---|---|
| | | d | | | w | r | | l | f |
| | g | q | p | | i | h | e | j | z |
| | a | | | b | v | o | x | | n |
| | k | t | y | m | | | s | c | u |

Valeur finale de la fonction objectif: 350852.7989907235

Nombre d'itérations:

Température de départ:

Algorithme Génétique

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| b | n | f | | l | a | d | h | k | w |
| j | x | p | v | t | s | e | r | y | c |
| z | q | g | o | m | i | u | | | |

Valeur finale de la fonction objectif: 246551.1596799356

Nombre de générations:

Taille de la population:

Probabilité de mutation:

Recherche Tabou

| | | | | | | | | | |
|--|---|---|---|---|---|---|---|--|---|
| | g | v | s | a | d | b | c | | k |
| | y | m | i | e | n | r | h | | |
| | x | p | t | i | u | o | w | | |
| | j | | f | q | | z | | | |

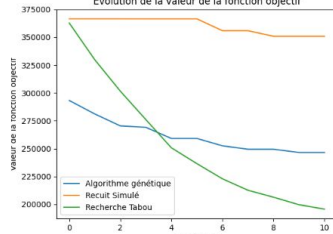
Valeur finale de la fonction objectif: 195681.24812313757

Nombre d'itérations:

Taille de la liste tabou:

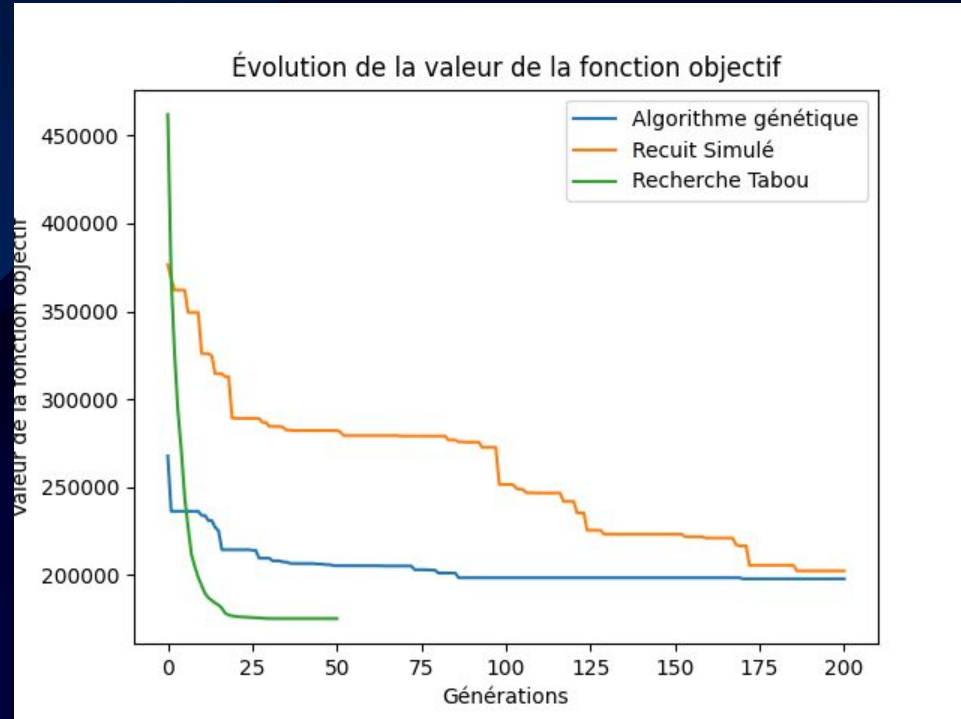
Start

Evolution de la valeur de la fonction objectif



Site Web Python
(Flask)

Interface de notre projet



Exemple de résultat