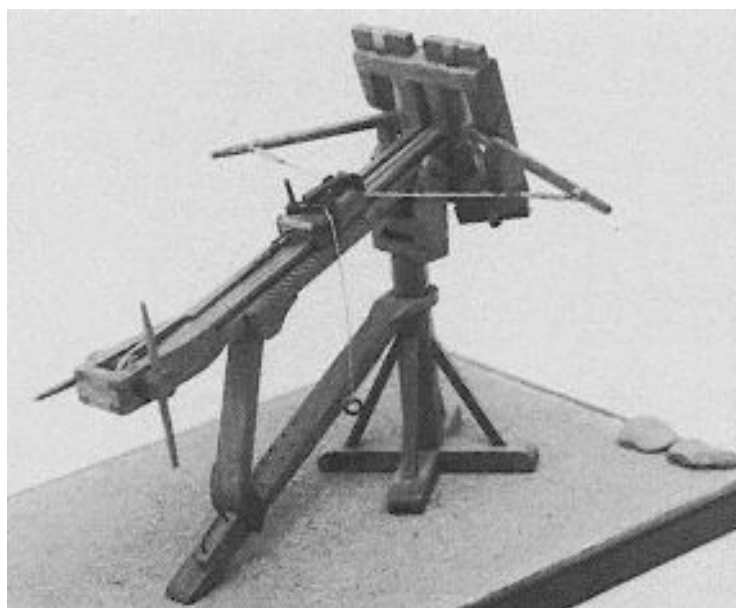


Algorithme génétique

TP - Scorpion



BOUCHAIN Loïc

Bachelor 3 Informatique

Ynov Nantes 2019/2020



Introduction

1. Contexte
2. Fonctionnement du programme
3. Graphique
4. Autres

Le contexte

Le scorpion ou scorpio (lanceur de flèche) était une pièce d'artillerie romaine inventée en 50 av. J.-C., (également connue sous le nom de baliste quand il lançait des boulets mais nous ne lancerons pas de boulet) -- Wikipedia. On a reconstruit des scorpions, en se basant, entre autre, sur les écrits de Vitruve.



Le fonctionnement

- L'objectif

L'objectif du programme est de créer un scorpion faisant le plus de dégâts possible à sa cible sur de longues portées.

- Comment le scorpion est construit ?

Les éléments qui constituent le scorpion sont générés aléatoirement suivant une plage de valeurs données.

Ainsi chaque scorpion créé est différent des autres permettant d'explorer un maximum de possibilités.

- Comment le score de fitness est calculé ?

L'objectif étant d'avoir un scorpion faisant un maximum de dégâts, l'énergie d'impact joue un rôle important sur ce score.

Ce score est calculer en fonction de plusieurs facteurs :

- L'énergie d'impact.
- La portée du tir.
- Si le scorpion peut tirer.
- Et si le scorpion ne rompt pas.

Ainsi, le scorpion comment avec un score de 20.

Chaque scorpion est évalué et son score varie en fonction de ces facteurs.

- Si le scorpion ne peut pas tirer son score diminue, en effet en cas de rupture du scorpion ou de l'impossibilité d'assembler le bras de l'arc (si la longueur de la corde est trop courte), le score de diminue de 5000 points, sinon il augmente du même montant.
- Si la portée du tire est inférieur à 300 m son score diminue de 5000 points, sinon son score augmente de 5000.
- Le résultat d'énergie d'impact est ajouté au score permettant de lui accorder beaucoup d'importance.

- Comment le programme fonctionne ?

Le programme commence par générer une première génération de scorpion, son nombre dépend de valeur initiée.

Ensuite une phase d'évaluation des scorpions est faite, un score de fitness pour chaque scorpion est alors donné.

Vient ensuite la phase de choix, j'ai choisi la méthode du "Tournement" permettant de choisir, en fonction de leur score de fitness, les 2 meilleurs, les 2 seconds ... afin d'obtenir les meilleurs couple de scorpions ensemble.

Une phase de croisement vient alors, pour chaque couple on définit le nombre de propriétés qu'ils vont s'échanger (entre 50 et 80% de 12 propriétés), chaque propriété à changer est définie aléatoirement (longueur de bras, longueur de flèche...). Ainsi un croisement permet d'interchanger un nombre aléatoire de propriétés, avec chaque propriété définie aléatoirement.

Par exemple, si un couple doit d'interchanger 50% de ses propriétés, 6 (50% de 12) propriétés sont choisies aléatoirement.

Par la suite, il y a une phase de mutation, chaque scorpion a 1% de chance de se faire modifier une propriété définie aléatoirement.

Pour finir, on supprime la moitié des scorpions, cette moitié est celle qui a le score de fitness le plus bas pour recréer le même nombre de scorpions, ceci permet d'avoir de nouveaux scorpions avec de nouvelles propriétés.

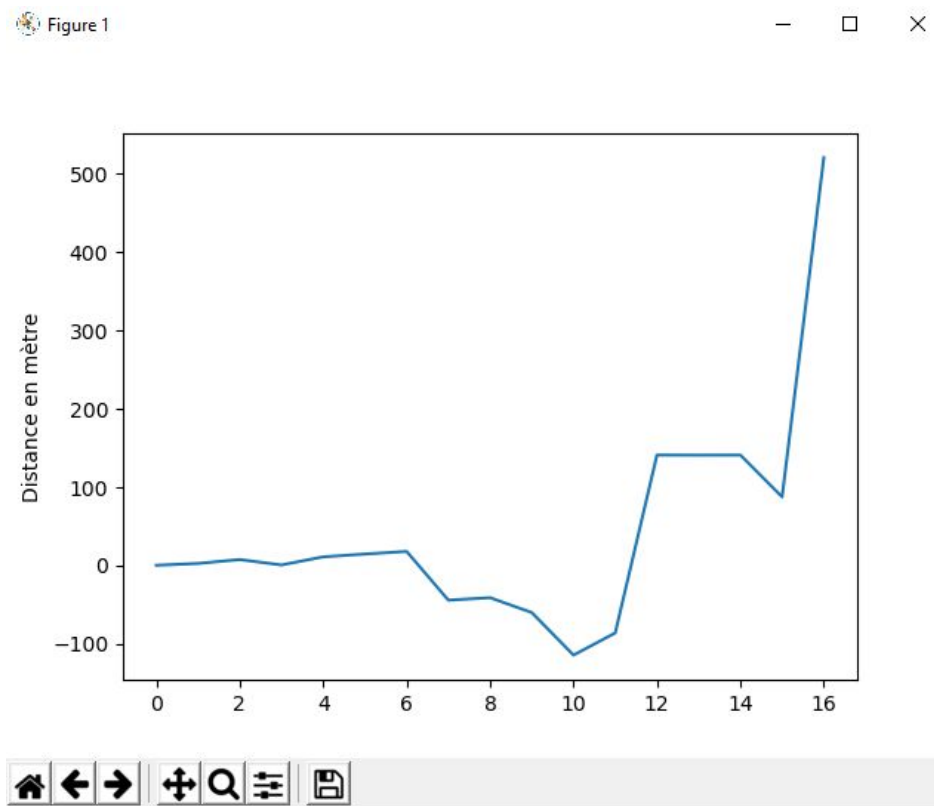
Ces instructions se répètent tant qu'un scorpion ne puisse pas :

- Pouvoir tirer (pas de rupture ou problème d'assemblage).
- Tirer à plus de 300 m.
- Avoir un score de fitness supérieur à 16 000.

Les graphiques

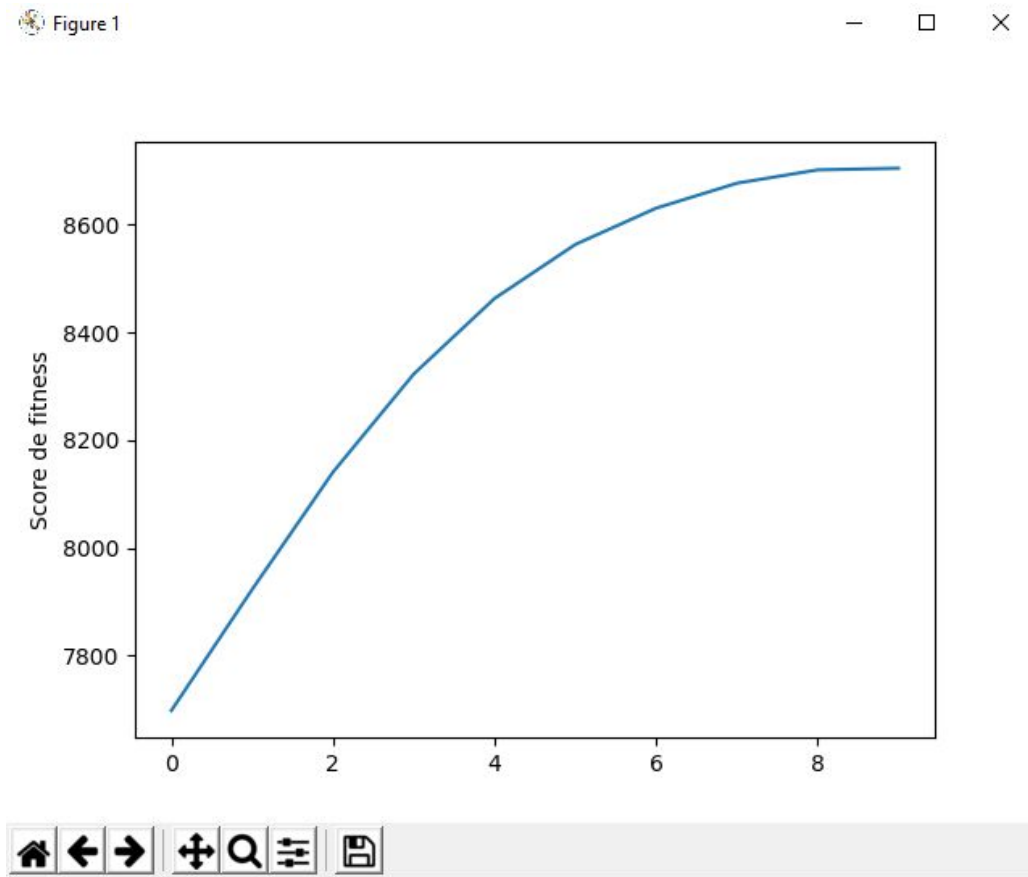
Graphique de la moyenne de la distance de chaque génération, jusqu'à remplir les conditions.

À noter : chaque lancement du programme le résultat sera différents car les données sont générées aléatoirement.



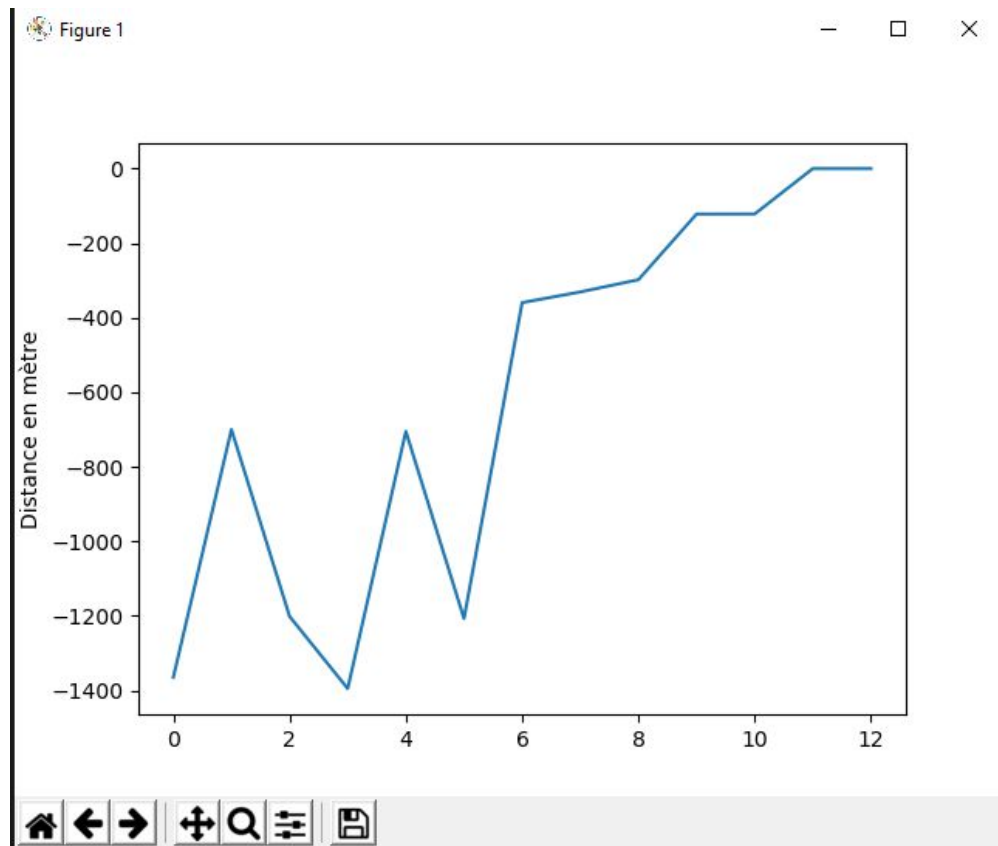
Ici on peut voir que c'est la 16ème et dernière génération qui a la meilleur moyenne, de plus la moyenne des distances des scorpions est quasiment en constante progression depuis la 1er génération.

Graphique de la moyenne du score de fitness pour chaque génération, jusqu'à remplir les conditions.



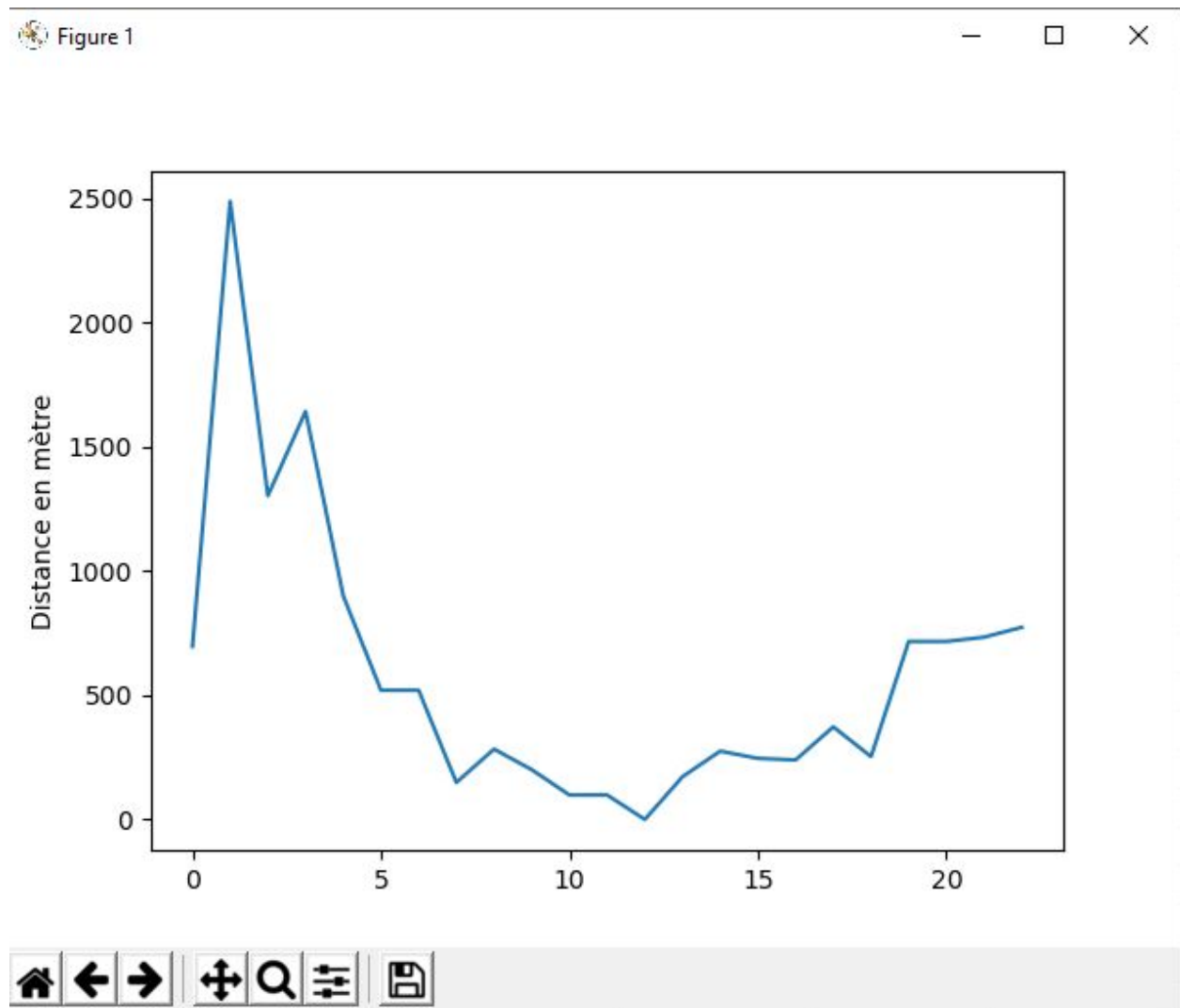
Le graphique montre une progression constante de la moyenne de fitness pour chaque génération.

Graphique du scorpion ayant la plus petite distance pour chaque génération, jusqu'à remplir les conditions.



Le graphique montre globalement une augmentation de la distance minimum pour chaque génération. De plus les valeurs sont négatives dans les premières générations et progresse vers des valeurs positives vers la fin.

Graphique du scorpion ayant la plus grande distance pour chaque génération, jusqu'à remplir les conditions.



On peut observer que les plus grandes distances sont dans les premières générations, certainement dû au fait que ces scorpions ne pouvaient pas tirer ou autres, la distance maximum a diminuée en fonction des générations.



Divers

Lien gitlab : <https://gitlab.com/loicbouchain/tpalgo.git>

Tutoriel : Changer les valeurs dans le constructeurs de l'objet scorpion pour changer les valeurs d'un scorpion.

Changer la valeur NUMBER_OF_SCORPION pour changer le nombre de scorpion

python scorpion.py pour lancer le programme.