

TP Algorithmes génétiques

Le scorpion

Le principe de ce TP va être d'utiliser la programmation d'algorithmes génétiques pour améliorer les dégâts d'un scorpion (l'arme pas l'animal) sur une cible située à la même altitude. Le présent document a pour but de vous aider dans les étapes permettant la réalisation de ce TP.

La présentation graphique de la résolution (la trajectoires) et des résultats est l'objectif.

Les éléments à prendre en compte sont les suivants :

- Le contexte choisi
- La structure de données utilisée
- La fonction de fitness choisie
- Le code (à commenter)
- Un ou des jeux de test

Pour chaque jeu de test, des données de performance seront les bienvenues.

TP Algorithmes génétiques

1. Contexte

Vous allez travailler à trouver les meilleurs paramètres pour la création d'un scorpion. 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 scorpion, en se basant, entre autres, sur les écrits de Vitruve. On se propose ici d'utiliser les technologies les plus modernes pour battre ces reconstitutions.

Un peu plus d'information sur le scorpion Vitruvien sur ce site :

<http://www.leg8.com/histoire-vivante/armee-romaine/artillerie/scorpion-vitruvien>

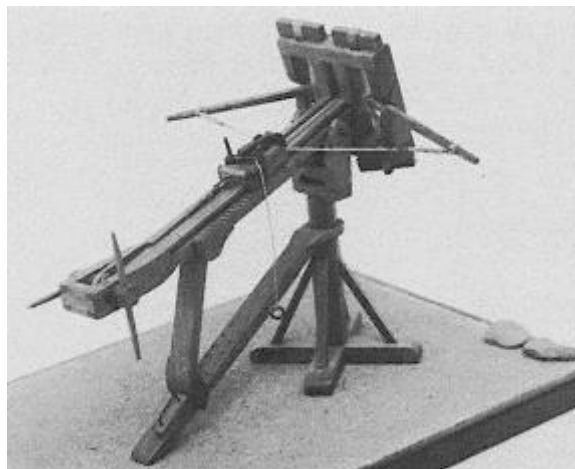


Figure 1: Un scorpion

Résumons la situation par un schéma.

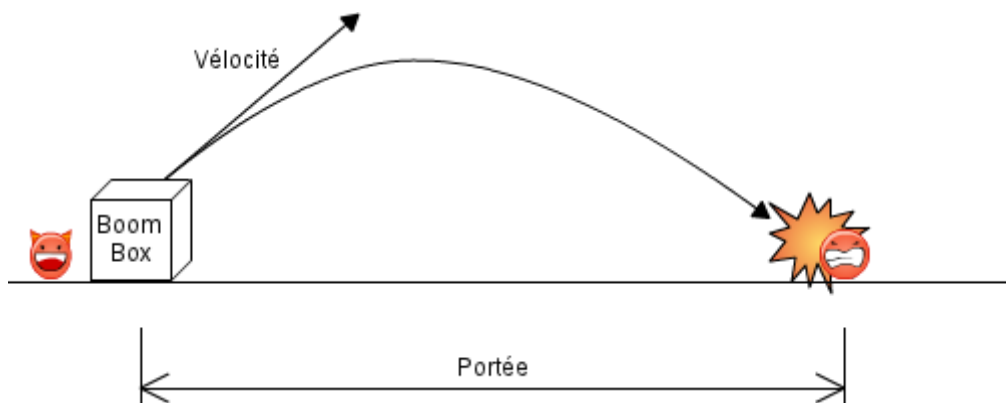


Figure 2: Le principe

Comme la dimension métier du problème est assez liée à la physique et plus particulièrement à la balistique, les équations et leurs significations seront fournies. Grâce sera faite de toutes les résistances et autres frictions. Seul le cas idéal sera considéré.

L'objectif final est de **maximiser les dégâts possibles à une cible située à une certaine Portée** du scorpion, soit de maximiser l'énergie cinétique que nous allons lui transmettre.

Regardons donc comment est construite un scorpion de façon sommaire.

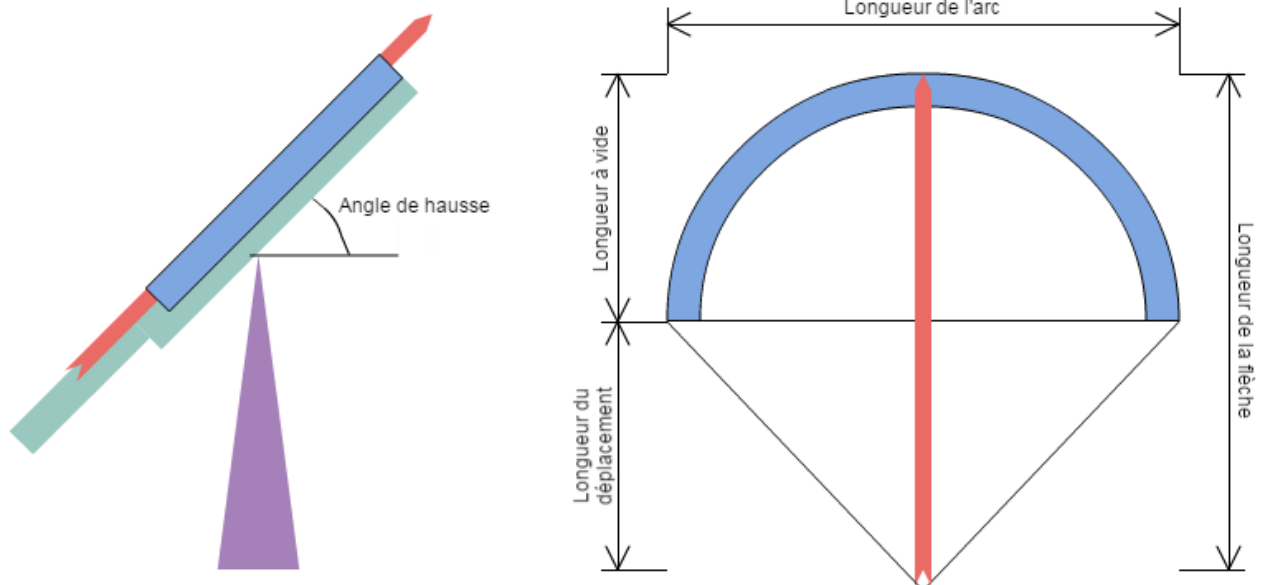


Illustration 1: Principe d'un scorpion

Si on liste les paramètres simplifiés rentrant en compte dans notre TP, nous obtenons :

- L'angle de hausse, qui peut être déterminé par l'angle α (en degrés mais **attention Python fait ses calculs d'angle en Radians**)
- la longueur du bras de l'arc L_b (en mètres)
- la base de la section du bras b (en mètres)
- la hauteur de la section du bras h (en mètres)
- la longueur de la corde L_c (en mètres)
- la longueur de la flèche L_f (en mètres)

Certains éléments dépendent eux des matériaux utilisés. Et voici un lien pour vous aider à borner ces valeurs et éventuellement à déterminer de quelles matières votre scorpion aura besoin.

http://www.simulationmatériaux.com/ComportementMecanique/comportement_mecanique_Liste_modules_de_Young.php

- La masse volumique de la flèche ρ (en kg/m^3)
- Le module de Young du matériau de l'arc E (en GPa)
- Le Coefficient de Poisson du matériau de l'arc ν

Et pour rappel :

- sur Terre
- sur la Lune
- sur Jupiter

$$g = 9,81 \text{ m.s}^{-2}$$

$$g = 1,62 \text{ m.s}^{-2}$$

$$g = 24,80 \text{ m.s}^{-2}$$

2. Règles métier

Nous en arrivons maintenant à la liste des équations qu'il faut posséder pour résoudre ce problème. Bref, la composante métier.

1. Ressort K (en N/m)

$$K = \frac{1}{3} \frac{E}{1 - 2\nu}$$

2. Longueur à vide (en m)

$$L_v = \frac{1}{2} \sqrt{L_b^2 - L_c^2}$$

3. Longueur du déplacement (en m)

$$L_d = L_f - L_v$$

4. Masse du projectile (en kg)

$$m_p = \rho \cdot b_f \cdot h_f \cdot L_f$$

Ou

$$m_p = \rho \cdot \pi \left(\frac{D_f}{2} \right)^2 \cdot L_f$$

5. Vitesse V (en m/s)

$$V = \sqrt{\frac{K \cdot L_d^2}{m_p}}$$

6. Portée P (en m)

$$d = \frac{V^2}{g} \cdot \sin(2\alpha)$$

7. Energie d'impact (en joules), assimilée à la force cinétique transformée à l'impact

$$E_c = \frac{1}{2} m_p V^2$$

8. Equivalence Joule et gramme de TNT

$$Energie_{TNT} = \frac{Energie_{Joule}}{4184}$$

3. Limites

Maintenant, quelques points peuvent poser problèmes. Par exemple, la résistance des matériaux nous apprend que le bras peut casser. Il y a rupture si $L_d > f$ (et il ne vaut mieux pas être à côté). Calculons f .

9. Moment quadratique du bras I (en m^4)

$$I = \frac{b \cdot h^3}{12}$$

10. Force de traction F (en N)

$$F = K \cdot L_d$$

11. Flèche du bras f max

$$f = \frac{F \cdot L_b^3}{48 \cdot E \cdot I}$$

D'autres cas existent. Réfléchissez-y. Mais à minima, on pourra trouver les cas suivants :

- Si la longueur à vide est supérieure à la longueur d'une flèche, pas de tir.
- Si la longueur de corde est supérieure à la longueur de l'arc, pas de tir.

4. Implémentation

Et nos algorithmes génétiques dans tout cela ? Voici des étapes pour vous aider/guider.

1. Déterminez quelle est votre population. Sa taille sera n .
2. Qui seront les individus de cette population ?
3. Dans quelle proportion peuvent varier les gènes de ces individus

Attention, les valeurs peuvent être interchangeables lors du croisement. Ensuite, les gènes étant déterminés, essayez de croiser un ou deux individus à la main, à des hauteurs diverses, pour tester votre idée.

4. Réalisez la fonction de génération aléatoire de votre population

Ici se pose la question de la forme de vos chromosomes. En effet, sous quelle forme allez-vous stocker l'ensemble des gènes : chaîne de caractère (avec un caractère pour un chromosome), graphe...

5. À l'aide des équations précédentes, construisez une fonction d'évaluation qui donnera une note à chaque individu, représentant son adaptation au problème.

N'oubliez de mal noter les individus qui ont un défaut mais sans les éliminer.

6. En vous appuyant sur cette note, il s'agira de tirer au sort $n/2$ couples.

Chaque chaîne a une probabilité d'être sélectionnée proportionnelle à son adaptation. Comme nous l'avons vu, plusieurs méthodes existent : Roulette Wheel Selection, Stochastic Universal Sampling, Tournament... À vous d'en implémenter au moins une.

7. Ensuite il s'agit de croiser les couples pour donner 2 chaînes filles. Pour cela deux étapes, l'enjambement et la mutation.

- a. L'enjambement consiste à couper les deux chaînes parentes en un point fixe et à les échanger au-delà de ce point. On peut déplacer le point de coupe aléatoirement assez régulièrement (entre 50 et 80% des cas, à tester).
- b. La mutation consiste à modifier de temps à autre (probabilité à fixer mais 1% est un bon départ) un des chromosomes d'une chaîne fille.

Vous utiliserez pour votre TP le Python et le framework DEAP (<https://github.com/DEAP/deap>).

L'affichage graphique montrera la trajectoire mais aussi les informations les plus utiles pour déterminer si votre population a évolué dans le bon sens (variance, moyenne...). Un exemple où vous pouvez voir l'évolution de la variance (rouge) mais aussi le score fitness moyen, minimum et maximum :

Alpha: 48.48; Lb: 99.69; b: 17.01; h: 74.19; Lc: 14.35; Lf: 1.23; diaFleche: 1.23
Rho: 1205.41; E: 448.59; nu: 0.48; V: 64.96; D: 322.04; NRJ: 0.89

