

I.A. TD 3 Recherche stochastique : les algorithmes génétiques

I.A. master 1 Informatique

4 décembre 2008

I.A. TD 3 Recherche stochastique : les algorithmes génétiques

Table des matières

1	Algorithmes évolutionnaires	1
2	Algorithmes génétiques	1
2.1	Principe général	1
2.2	Opérateurs génétiques	1
2.3	Sélection des individus	2
2.4	Codage des individus	2
2.5	Exemple	2
3	Application au problème des N reines	4
4	Quelques liens	4

1 Algorithmes évolutionnaires

Les « algorithmes évolutionnaires » sont des « **métaheuristiques** » dont le but est de trouver *approximativement* la solution d'un problème difficile plus rapidement qu'avec des approches analytiques classiques.

Les algorithmes évolutionnaires contiennent plusieurs classes, en particulier les « algorithmes génétiques » et la « programmation génétique ». Les algorithmes génétiques s'inspirent de la théorie génétique de la sélection naturelle en utilisant des opérateurs génétiques tels que la *mutation* et les *croisements* sur un ensemble de solutions pour trouver une solution quasi-optimale. La programmation génétique quant à elle applique ces opérateurs directement sur le programme menant à la solution.

2 Algorithmes génétiques

On doit l'invention des algorithmes génétiques à John H. Holland dès 1960.

Les algorithmes génétiques, comme la plupart des métaheuristiques, sont des méthodes *stochastiques*. Contrairement aux méthodes exhaustives (que nous avons appliquées jusqu'à présent avant l'emploi d'heuristiques) ou analytiques, ils cherchent une solution en utilisant le hasard, mais un hasard *guidé*.

2.1 Principe général

Un algorithme génétique travaille sur une *population* d'individus représentant des *solutions potentielles*. Ces solutions sont codées sous forme de chaînes de symboles : les nucléotides (les algorithmes génétiques de base utilisaient des chaînes de bits), un gène étant formé de plusieurs nucléotides. Par exemple si on cherche la valeur minimale d'une fonction $f(x)$, la chaîne de symboles sera représentée par les bits formant le paramètre x .

Chaque individu peut se voir attribuer un *score* par une fonction de *fitness*. Plus le score est élevé, plus l'individu est approprié (c-à-d. proche de la meilleure solution). C'est grâce à cette fonction de fitness que l'on va guider l'algorithme génétique. Dans l'exemple précédent, la fonction de fitness est directement f et plus sa valeur est basse en fonction de x meilleur sera le score de l'individu x .

On génère la population initiale aléatoirement. Afin d'améliorer les solutions, on crée ensuite d'autres générations en appliquant des *opérateurs génétiques* sur les individus de la génération précédente, jusqu'à ce qu'une solution satisfaisante soit trouvée. Les individus sont sélectionnés pour la reproduction en fonction de leur scores.

Un algorithme génétique suit donc le schéma suivant :

1. Codage du problème sous forme d'une chaîne de nucléotides.
2. Génération d'une population initiale d'individus.
3. Calcul du score de chaque individu.
4. Si l'objectif est atteint, sortie.
5. Sélection des *reproducteurs* en fonction des scores.
6. Construction des descendants par application de différents opérateurs génétiques (étape de reproduction).
7. Remplacement de la population précédente par les descendants (ou mélange avec les descendants). Goto 3.

2.2 Opérateurs génétiques

L'opérateur génétique principal est le *croisement*. Ce dernier génère, à partir de deux individus parents, un troisième dont la moitié du génome provient du premier parent et l'autre moitié du second parent. L'endroit où les deux génomes parents sont coupés, le *locus*, est en général choisi aléatoirement.

L'autre opérateur génétique important est la *mutation*. Il s'agit d'introduire dans le génome d'un individu un nucléotide aléatoire. Cette étape est appliquée après le croisement. Il ne doit en général pas y avoir trop de mutation (moins de 10%) pour éviter la création d'individus à trois têtes et donc non viables. En effet le « bruit » introduit par les mutations risque de détruire des séquences génétiques intéressantes sélectionnées par l'algorithme. Cependant, l'opérateur de mutation est essentiel pour dépasser les limites imposées par le matériel génétique de départ.

Il existe bien sûr d'autres types d'opérateurs génétiques. Le croisement peut être décliné en fonction du nombre de loci, d'une probabilité d'échange de nucléotides, *etc.* De même, la mutation peut porter sur plusieurs gènes, *etc.*

2.3 Sélection des individus

La sélection des individus pour la reproduction peut se faire de différentes façons. La méthode la plus utilisée, dite *roue de la fortune biaisée*, consiste à ce qu'un individu ait une probabilité d'être choisi directement correspondante à son score. Une autre méthode, dite *par tournoi* sélectionne au hasard une paire d'individus et ne garde que le meilleur. Une variante, le *tournoi stochastique*, tire un nombre aléatoire entre 0 et 1 et si ce chiffre est inférieur à une valeur prédéterminée (ex. 0.8), choisit le meilleur individu de la paire, sinon le moins bon.

Bien sûr un individu peut être sélectionné plusieurs fois pour la reproduction, et il est possible de jouer sur le fait que les meilleurs individus soient sélectionnés plus souvent que d'autres. La population précédente peut être complètement remplacée à chaque étape ou remplacée en partie. La partie restante peut alors être par exemple formée des de ses meilleurs individus.

On peut aussi éviter la convergence vers un minimum local (une population globalement bonne ayant des caractéristiques très proches, mais ne représentant pas un optimum général au problème) par l'introduction à chaque étape de reproduction d'individus aléatoires.

2.4 Codage des individus

Historiquement le codage binaire des solutions a souvent été préféré au codage par chaîne de symboles. Néanmoins, ce dernier est plus pratique et laisse une plus grande latitude. John Holland avait préféré le codage binaire car les chaînes générées sont souvent plus longues et fournissent donc plus de loci pour le croisement, opérateur essentiel de l'algorithme.

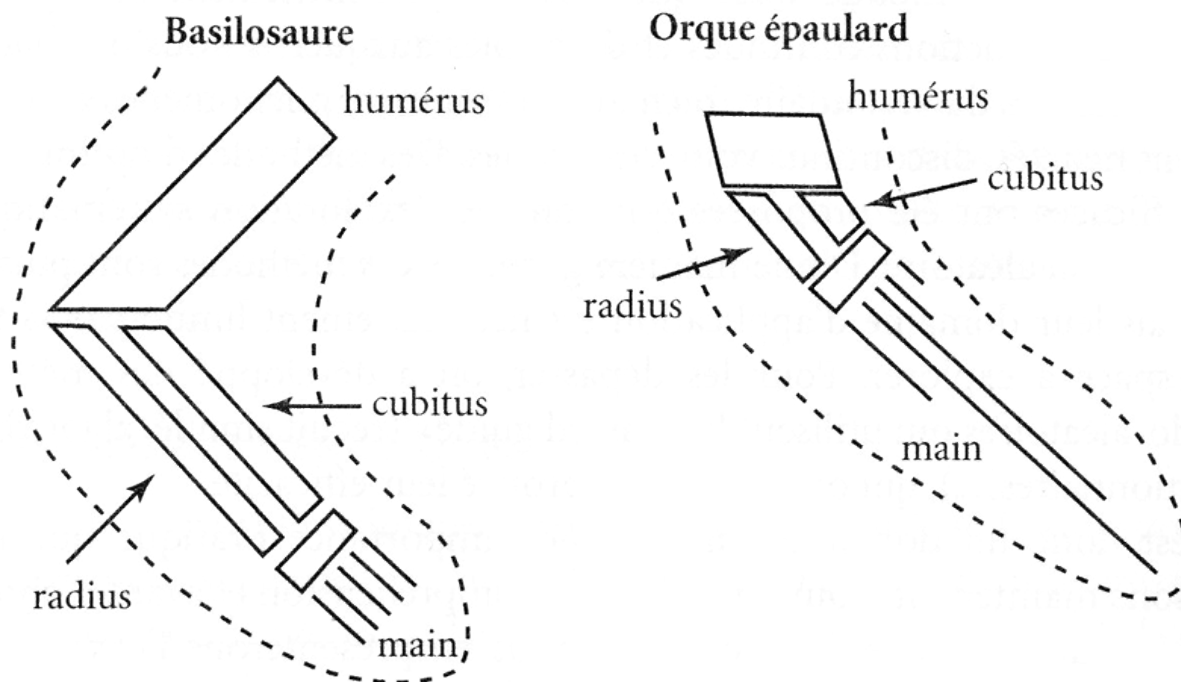
2.5 Exemple

Cet exemple est tiré du livre de Jean-Philippe Rennard, *Vie artificielle* (Vuibert informatique).

Remontons 40 millions d'années plus tôt, et considérons le basilosaure, un prototype de la baleine moderne. Ce mammifère était retourné à la vie aquatique après une phase terrestre, et ses nageoires étaient encore peu adaptées à la nage. Hors, le basilosaure était un prédateur, et sa survie dépendait de son agilité à attraper les poissons rapides de la côte.

L'amélioration de ses nageoires vers une forme efficace était donc vitale. Pour cela, un double phénomène :

- Raccourcissement du « bras » ;
- allongement des « doigts » qui vont former l'ossature principale de la palette natatoire.



Nous allons étudier uniquement la partie du chromosome dédiée au codage de la nageoire, de manière simplifiée. Ce chromosome est fait de deux gènes formés chacun de deux nucléotides. Le premier gène code la longueur du bras, le second la longueur des doigts. De plus le basilosaure est haploïde (un seul chromosome) et il n'existe que deux sortes de nucléotides: L et C pour la forme long et courte respectivement de chaque partie de la nageoire, et la position du nucléotide dans le gène est indifférente.

Au départ la population de basilosaures est en majorité formée d'individus ayant des bras longs et des doigts courts, héritage d'individus adaptés à la marche :

LLCC

Le but est d'arriver à des individus adaptés à la nage avec des bras courts et des doigts longs :

CCLL

On considère maintenant la population suivante:

A	LCCC	1
B	CLCC	1
C	CLCL	2
D	CCLC	3

Les scores sont indiqués en face des chromosomes, ils correspondent au nombre de nucléotides à la bonne place par rapport à la forme "efficace" de la nageoire.

Voici à quoi pourrait correspondre le premier cycle reproductif :

Accouplements	Croisements	Descendants	Score
A : LCCC et D : CCLC	LC LL *	A' : LCLL	3
B : CLCC et D : CCLC	CL LC	B' : CLLC	2
C : CLCL et B : CLCC	CL CL	C' : CLCL	2
C : CLCL et D : CCLC	C CLC	D' : CCLC	3

* une mutation s'est produite ici, transformant LC en LL.

L'individu D qui avait un score de 3 avait de « plus belles nageoires » et a donc été choisi plus souvent pour l'accouplement. Cette probabilité de choix induit une tendance à l'élévation du score moyen au cours des cycles reproductifs. Ici, le score total est de 10 contre 7 à la génération précédente. Si au cours du cycle suivant C' et D' s'accouplent il y a des chances que l'individu soit doté du chromosome recherché (en coupant à deux loci cependant).

Le lien 1 donne un très bon exemple, avec une applet, de ce mécanisme.

3 Application au problème des N reines

Utilisez un algorithme génétique pour trouver une solution au problème des N reines :

- Comment coder les individus ?
- Comment générer la population initiale ?
- Quelle sera la fonction de fitness ?
- Comment choisir la taille de la population initiale ?

Comparez la vitesse d'exécution avec la méthode récursive avec heuristiques utilisée précédemment. Critiquez.

Si vous avez fini en avance : nous avons utilisé des individus haploïdes, considérez des individus diploïdes. Quels pourraient être les avantages ?

4 Quelques liens

1. [Une applet de démonstration](#) (on peut en particulier sur cette applet comparer les approches "codage par bits" et "codage par gène")
2. [Appliqués au problèmes de trafic aérien](#)