

Les algorithmes génétiques

Bernard Vuilleumier

Centre informatique
pédagogique (CIP)
Rue Théodore-de-Bèze 2
Case Postale 3144
1211 GENÈVE 3
Tél: (022) 318.05.30
Fax: (022) 781.03.50
Responsable:
Raymond Morel

Lettre n° 31

*En 70 à 100 millisecondes,
nous regardons, portons
attention et reconnaissons.
Notre système visuel s'adapte
pour produire une réponse*

*Un algorithme génétique est
aussi un système adaptable.
Il s'inspire de la génétique et
de la théorie de l'évolution et
en utilise le vocabulaire*

*Mais vous pouvez, si vous le
souhaitez, vous passer de la
terminologie biologique et
parler de chaîne de variables,
de position dans la chaîne et
de valeur au lieu de chromo-
some, de gène et d'allèle*

Les algorithmes génétiques, comme les réseaux d'automates, sont des systèmes adaptables. Mais qu'est-ce au juste qu'un système adaptable? Le système visuel par exemple est un système adaptable. Considérez la façon dont il répond au stimulus provoqué par le motif suivant:

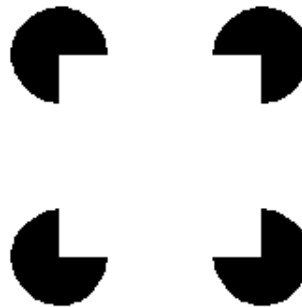


Fig. 1 : Lorsque nous regardons cette figure, nous voyons se détacher un carré «plus blanc que blanc» sur un fond blanc. La frontière entre le carré et le fond blanc n'existe pas: c'est le résultat d'une adaptation.

Nous voyons un carré blanc se détacher d'un fond blanc. Nous percevons une frontière imaginaire entre le fond et le carré. Mais voyons-nous vraiment un carré? Au sens strict, nous ne pouvons pas le voir car il n'existe pas. Nous reconnaissons un carré. Le carré existe dans notre cerveau mais pas dans la réalité physique: il n'y a sur la page que quatre motifs symétriques! Notre système visuel s'est «adapté» au stimulus pour produire une réponse. Un algorithme génétique est également capable de s'adapter. C'est une procédure qui s'inspire de la génétique et de la théorie de l'évolution.

En s'accouplant, les membres d'une population produisent des descendants dont l'adaptation au milieu est d'autant meilleure qu'ils retiennent les «bonnes» caractéristiques de leurs parents. De cette manière, l'adaptation de la population peut augmenter de génération en génération. Dans un organisme biologique, la structure qui code les prescriptions spécifiant comment l'organisme doit être construit est le *chromosome*. Chaque chromosome comprend des *gènes*. Chaque gène code une caractéristique particulière de l'organisme, et la position du gène dans le chromosome détermine quelle caractéristique le gène représente. Pour une position donnée, un gène peut coder une des différentes valeurs de la caractéristique qu'il représente: la couleur des yeux par exemple peut être brune, verte, bleue, etc. Les différentes valeurs d'un gène sont les *allèles*. Dans un algorithme génétique, les chromosomes sont typiquement représentés par une *chaîne de variables*. Chaque variable occupe une *position* dans la chaîne et correspond à un gène. La variable peut prendre différentes *valeurs* qui sont le pendant des allèles.

La plupart des problèmes auxquels les algorithmes génétiques s'appliquent sont des problèmes d'optimisation. On identifie la quantité qui doit être optimisée à l'adaptation ou à la chance de survie d'un individu et on applique les règles suivantes:

- commencer avec une population d'individus dont les allèles sont générés aléatoirement;
- déterminer la chance de survie de chaque individu dans la population courante;
- sélectionner les parents pour la génération suivante en fonction de leur chance de survie;
- accoupler les parents sélectionnés pour produire la descendance;
- répéter les étapes 2 à 4 jusqu'à ce qu'une solution satisfaisante soit trouvée.

Travaux pratiques

Pour les exercices, je me suis inspiré du chapitre «Genetic Algorithms» de l'ouvrage de James A. Freeman, *Simulating Neural Networks*, Addison Wesley, New-York (1994)

Pour illustrer la notion de «chaîne de variables»

Exercice 1

Considérons dix singes assis devant dix machines à écrire. Laissons chaque singe frapper huit lettres (nous supposons que le clavier ne comporte que des minuscules non accentuées). Appelons «génération» l'ensemble des chaînes produites par les dix singes. Si aucun singe n'a frappé une séquence choisie d'avance, nous remplaçons les feuilles dans chaque machine à écrire et recommençons l'expérience afin de produire une nouvelle génération.

- Simuler l'expérience décrite ci-dessus;
- calculer la probabilité qu'un singe obtienne la chaîne souhaitée.

Pour apprendre à convertir des lettres en nombres et à provoquer des mutations

Exercice 2

La probabilité calculée dans l'exercice 1 sous le point b) montre à l'évidence que la seule chance ne suffit pas à obtenir une chaîne choisie d'avance, même courte. Mais la notion de «sélection» n'intervient pas dans l'exercice 1. Pour introduire cette notion, nous allons modifier le processus de production des chaînes. La première génération de chaînes sera toujours tirée au hasard, mais les suivantes seront générées à partir de la chaîne la plus proche de la séquence souhaitée. Nous allons faire en sorte que chaque lettre puisse changer d'une génération à l'autre. Nous évaluerons l'adaptation des chaînes de chaque génération à la séquence désirée. Nous formerons la nouvelle génération à partir de la chaîne la mieux adaptée. **N.B.** Dans cet exercice, nous travaillerons avec les codes ASCII des lettres plutôt qu'avec les lettres elles-mêmes. Lorsque nous parlerons de lettre, il s'agira en fait du code ASCII de la lettre.

- Convertir la chaîne de votre choix en une liste de codes ASCII et affecter cette liste à la variable «keyPhrase»;
- tirer au hasard les lettres de la première génération et former 10 chaînes dont le nombre de lettres sera égal au nombre de lettres de la chaîne choisie;
- définir une fonction donnant «True» pour une probabilité supérieure ou égale à p et «False» autrement;
- définir, en utilisant le résultat de c), une fonction permettant de changer une lettre en une autre lettre avec une probabilité donnée;

Pour apprendre à mettre en œuvre la notion de «sélection»

Exercice 3

Ecrire un programme permettant de simuler un processus de production de générations avec sélection en utilisant les indications et résultats de l'exercice 2.

Pour illustrer les effets de la sélection

Exercice 4

Estimer le nombre de générations nécessaires pour obtenir, avec une adaptation voisine de 1, les chaînes suivantes:

- invinoveritas
- arslongavitabrevis
- fugitirreparabiletempus
- bonumvinumlaetificatcorhominis.

Prochaine réunion: lundi 5 septembre 1994 à 17h.

