

Algorithmes évolutionnistes

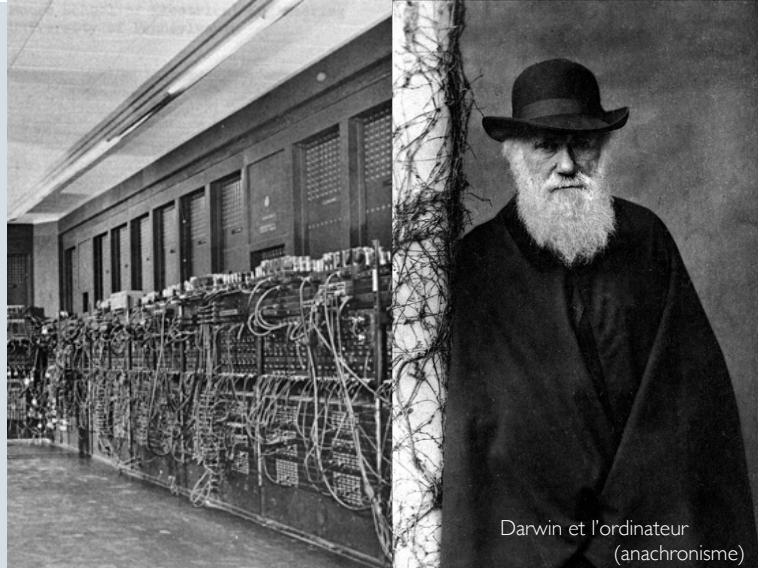
Optimisation stochastique à base de population

UE méthodes et outils pour l'IA et la RO
L3 informatique

Nicolas Bredeche

Université Pierre et Marie Curie
ISIR, UMR 7222
Paris, France
nicolas.bredeche@upmc.fr

...



Darwin et l'ordinateur
(anachronisme)



rev. 2017-03-23

Optimisation: éléments

2

Problème: $y^* = \arg\min_{y \in Y} f(y)$

Solution candidate: $a := (y, f(y))$

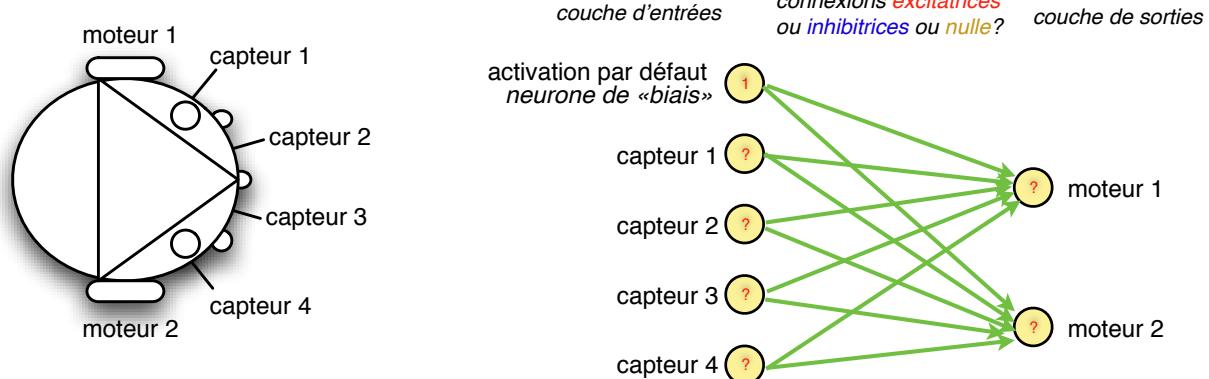
Formulation simplifiée

Formulation plus générale: $a := (y, s, f(y), f'(y), f''(y))$

- Des méthodes pour des classes de problèmes
 - ▶ Algorithme de gradient (recherche locale, suit le gradient)
 - ▶ Hill-climbing (recherche locale, change un élément à la fois)
 - ▶ Méthodes énumératives (recherche globale, espace de recherche discret)
 - ▶ Méthodes heuristiques (espace structuré)
 - ▶ Méta-heuristique et méthodes stochastiques
 - recherche aléatoire (recherche globale, sans a priori) [Monte carlo, Tabu]
 - recuit simulé (recherche globale) ["simulated annealing"]
 - méthodes bio-inspirées (recherche globale) [DE, PSO, Algo. évol., ...]

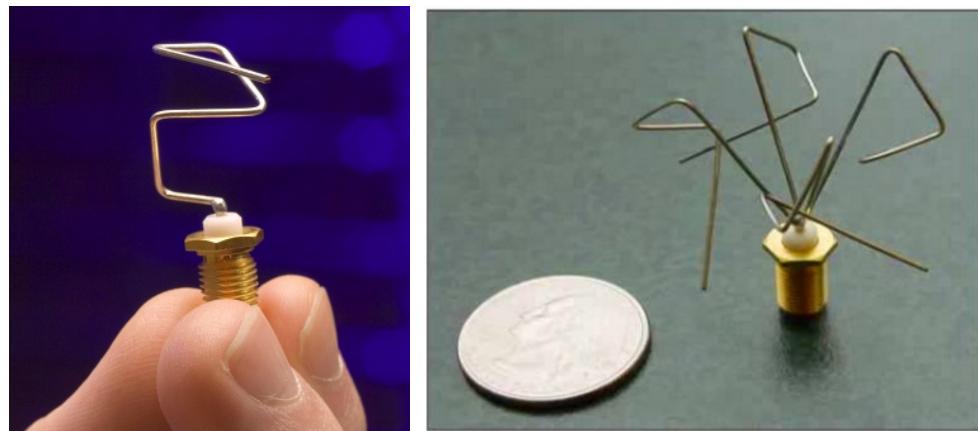
Recherche aléatoire

4



- Fonction objectif:
 - ▶ $f(y) = \text{distance parcourue pendant } n \text{ itération}$
- Description d'une solution:
 - ▶ $\{-1, 0, 1\}^{10}$, soit: $[x_1, x_2, \dots, x_{10}]$, avec $x_i \in \{-1, 0, +1\}$
- Algorithme:
 - ▶ recherche aléatoire

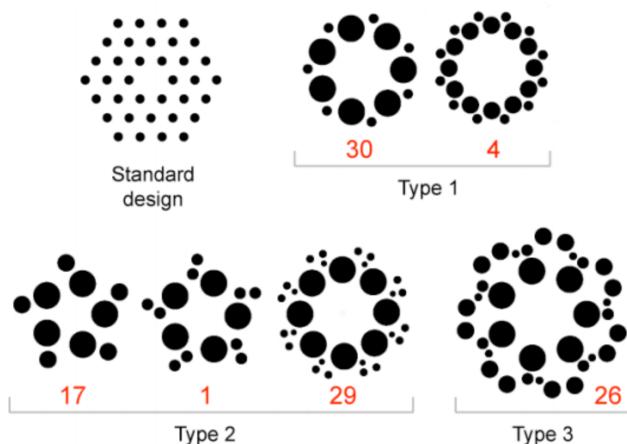
exemple pratique



- Antenne de petite taille pour satellite
 - Objectif: maximisez l'efficacité, minimiser la taille
 - Espace de recherche: structure de l'antenne (graphe)
 - Difficulté: absence de modèles d'interférences

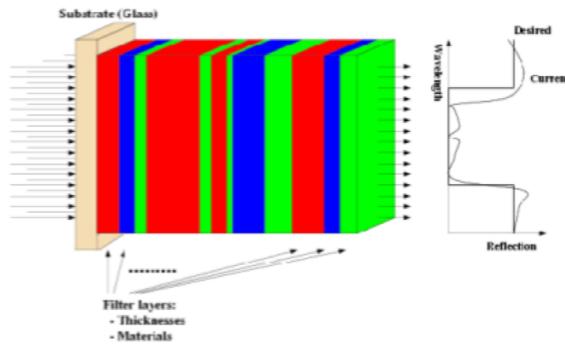
Source: Lohn et al., Genetic Programming Theory and Practice II, 2004

Fibre optique



- Microstructure d'une fibre optique
 - Objectif: max. efficacité sous contrainte de constructibilité
 - Espace de recherche: microstructure symétrique
 - Difficulté: formulation de l'espace de recherche

Source: Manos et al. GECCO 2007



- Définir les couches d'un filtre optique
 - Objectif: contraintes fixées par un cahier des charges
 - Espace de recherche: (matériau, épaisseur)ⁿ
 - Difficulté: espace de recherche discret et continu

[Image extraite de cours de Mr Schöenauer (avec permission)]

[Schutz, Bäck][Martin et al., 1995]

Recherche avec un critère subjectif, problème inverse



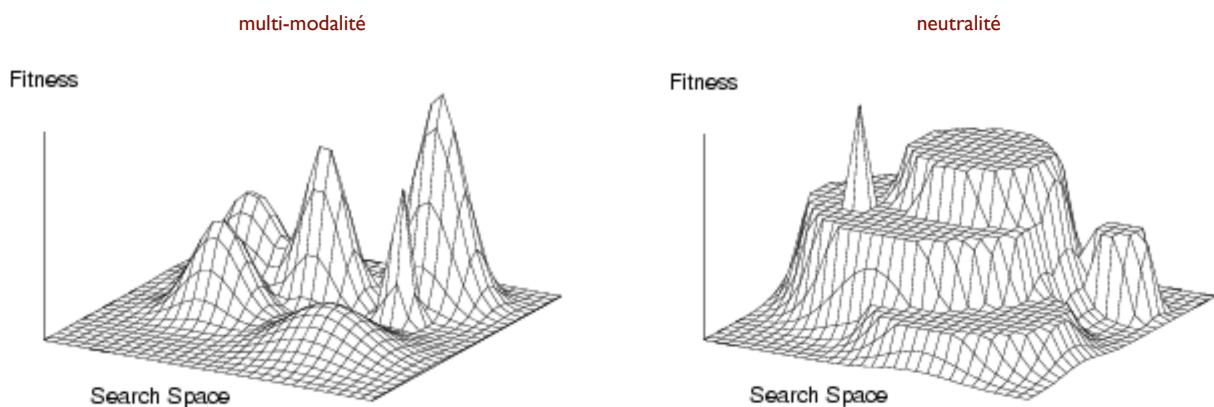
crédit photo: Chung-Leng Tran

- Retrouver la composition d'un café à l'arôme
 - Objectif: maximiser la satisfaction de l'expert-évaluateur
 - Espace de recherche: mélanges de café
 - Difficulté: évaluation empirique par l'expert

- Propriétés
 - Espace de recherche
 - ▶ binaire, symbolique, continu
 - ▶ structuré ou non
 - Fonction de performance
 - ▶ lien tenu entre représentation et performance
 - ▶ évaluation potentiellement bruitée
 - Relation faible entre espace de recherche et objectif

Caractérisation de l'espace de recherche

10



- espace de recherche “complexe”
 - ▶ multi-modalité, plateaux de neutralité
 - ▶ irrégulière, non différentiable, discontinue
- espace de recherche inconnu
 - ▶ problème type “boîte noire”
 - ▶ MAIS: notion de voisinage (sinon, il ne reste plus que Monte Carlo)

Algorithmes évolutionnistes*

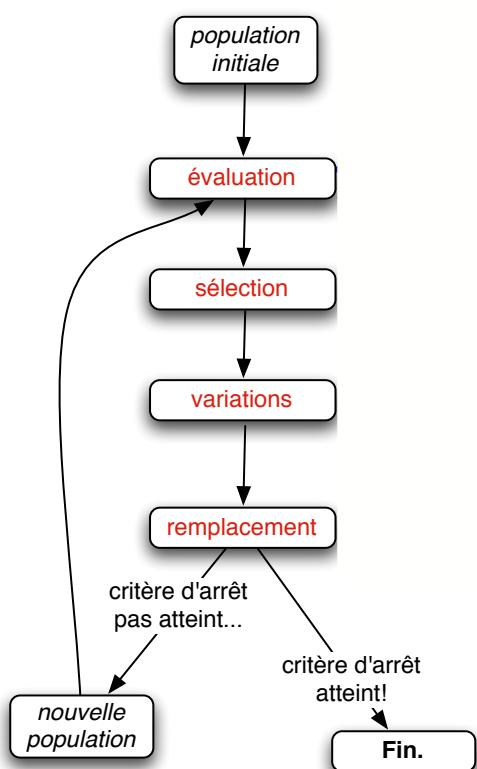
Principes généraux

*Ou: Algorithmes évolutionnaires

*Ou: Algorithmes pour l'optimisation stochastique à base de population

Optimisation par évolution artificielle

12



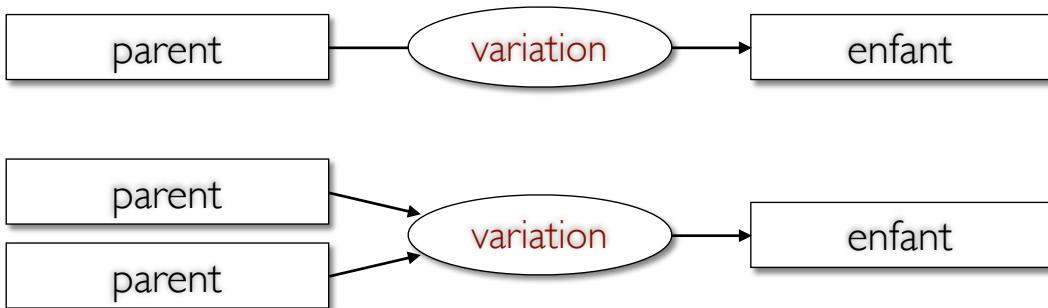
- Caractérisation
 - Algorithme d'optimisation stochastique à base de population
 - Famille des méta-heuristiques
- Deux mécanismes principaux
 - Pression à la sélection
 - Variations +/- aveugles
- Propriété souhaitée
 - Recherche globale et locale



- Définition
 - ▶ Sélectionne une sous-partie des solutions candidates
- Exemple
 - ▶ Renvoie les N meilleurs individus parmi M
- Propriétés
 - ▶ Déterministe vs. stochastique
 - ▶ Compromis exploration/exploitation
 - ▶ Elitiste ou non



- Méthodes
 - ▶ fitness proportionate
 - ▶ k-tournament
 - ▶ plus-selection ($\mu+\lambda$)
 - ▶ comma-selection (μ,λ)
 - ▶ NSGA-2 (multi-objectif)

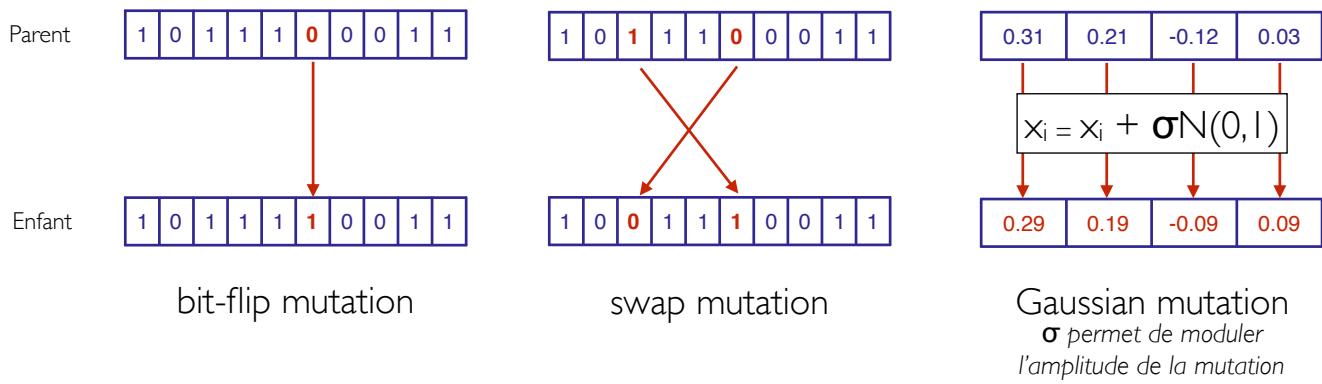


- Définition
 - ▶ Construit un nouvel individu à partir d'un (ou plusieurs) individus
- Exemple
 - ▶ Modifie aléatoirement un élément du génome
- Propriétés
 - ▶ Conservatif vs. disruptif

Opérateur de variation: mutation

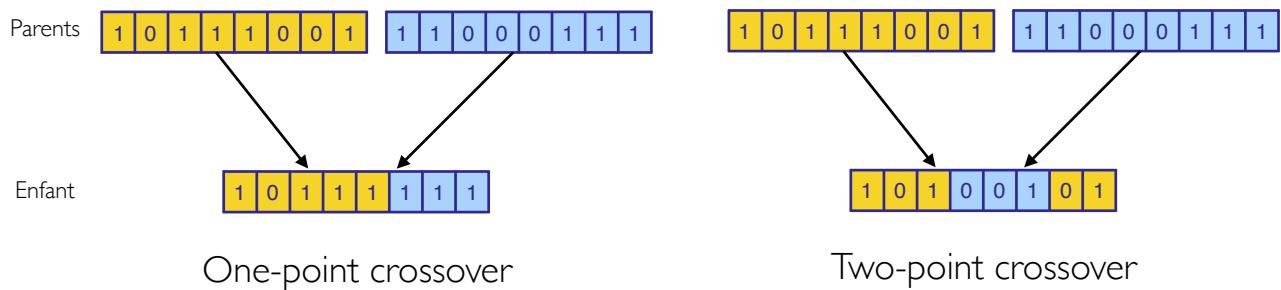


- Définition
 - ▶ Construit un nouvel individu à partir d'un seul individu parent
- Exemples



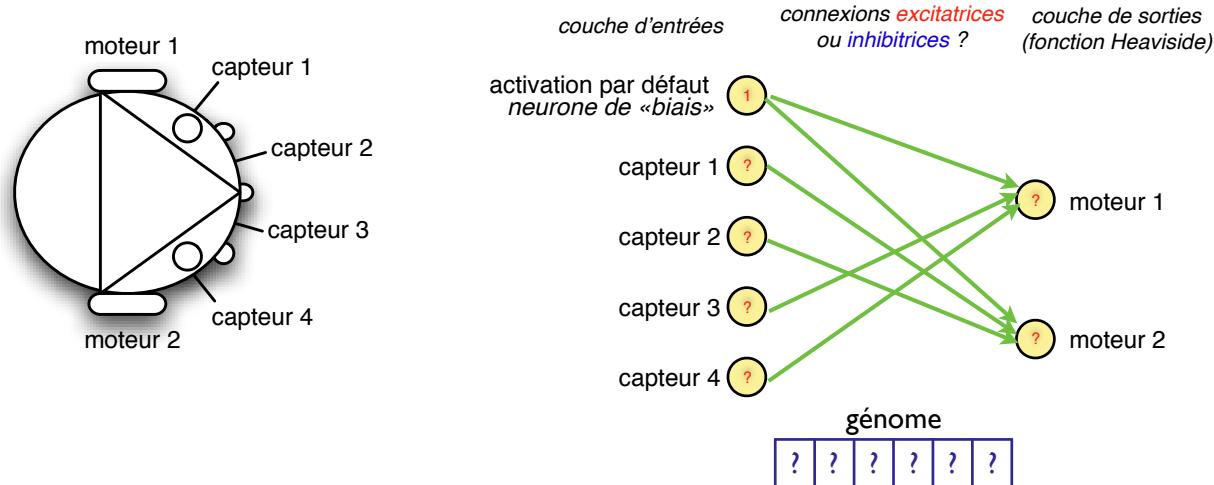


- Définition
 - ▶ Construit un nouvel individu à partir de 2 (ou +) individus parents
- Exemples:



Cas d'étude

Robotique évolutionniste



$$fitness(x) = \int_T V * (1 - \sqrt{\delta_v}) * (1 - sensor_{max})$$

- ➡ maximiser la vitesse de chaque moteur
- ➡ maximiser la vitesse de translation
- ➡ maximiser la distance au mur

nicolas.bredeche@upmc.fr

exemple

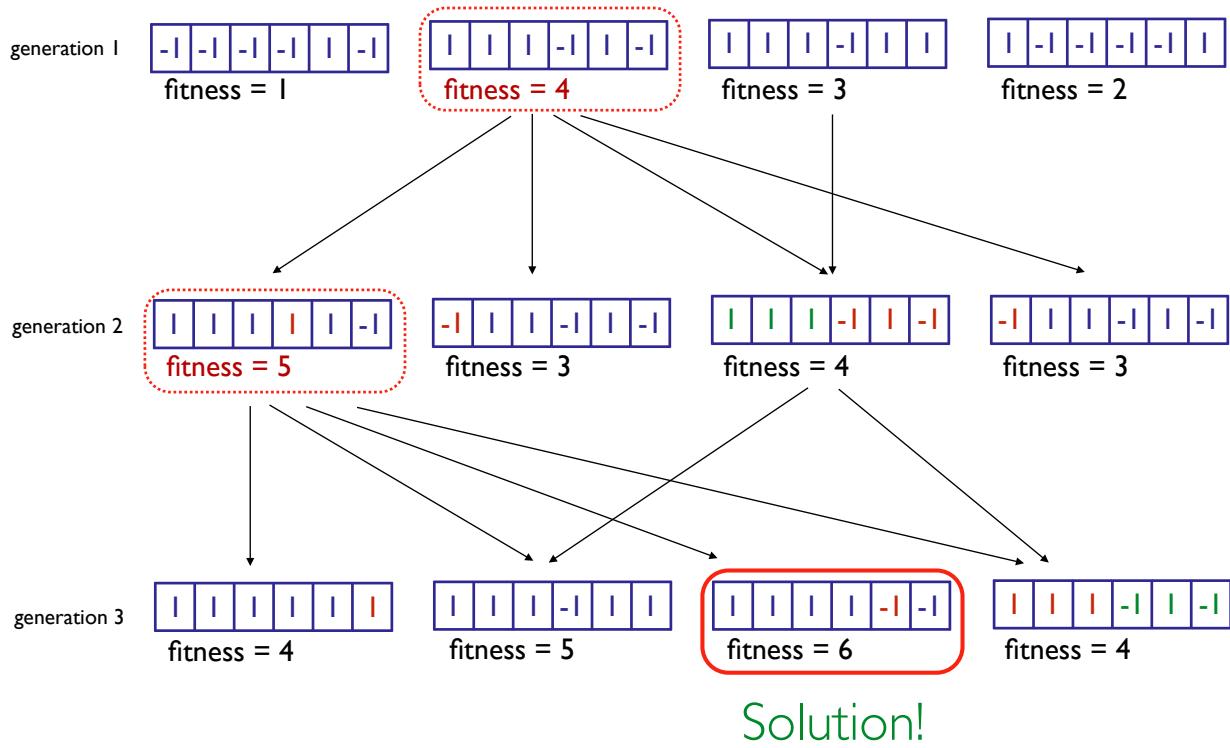
Problème : trouver le meilleur éviteur d'obstacles

il s'agit d'un problème jouet pour lequel on suppose ici que [1,1,1,1,-1,-1] est la solution optimale

- **Objectif :** maximiser la fonction fitness
- **Population initiale :** 4 individus tirés au hasard
- **Opérateur de Sélection :** prend le meilleur
- **Opérateurs de Variation :** croisement ou mutation
 - Probabilité de croisement: p ; probabilité de mutation: 1-p ; avec p=0.5
 - Croisement: on mélange le début d'un génome et la fin d'un second
 - Mutation: on change une valeur au hasard

Remarques:

1. les opérateurs sont ici choisis arbitrairement. D'autres choix sont possibles.
2. tout aussi arbitrairement, on décide ici que la performance est donnée par le nombre de valeur du génome correctes (ie. [1,1,1,1,-1,-1] vaut 6). En réalité, la performance résulte du comportement du robot.



nicolas.bredeche@upmc.fr

max-one

22

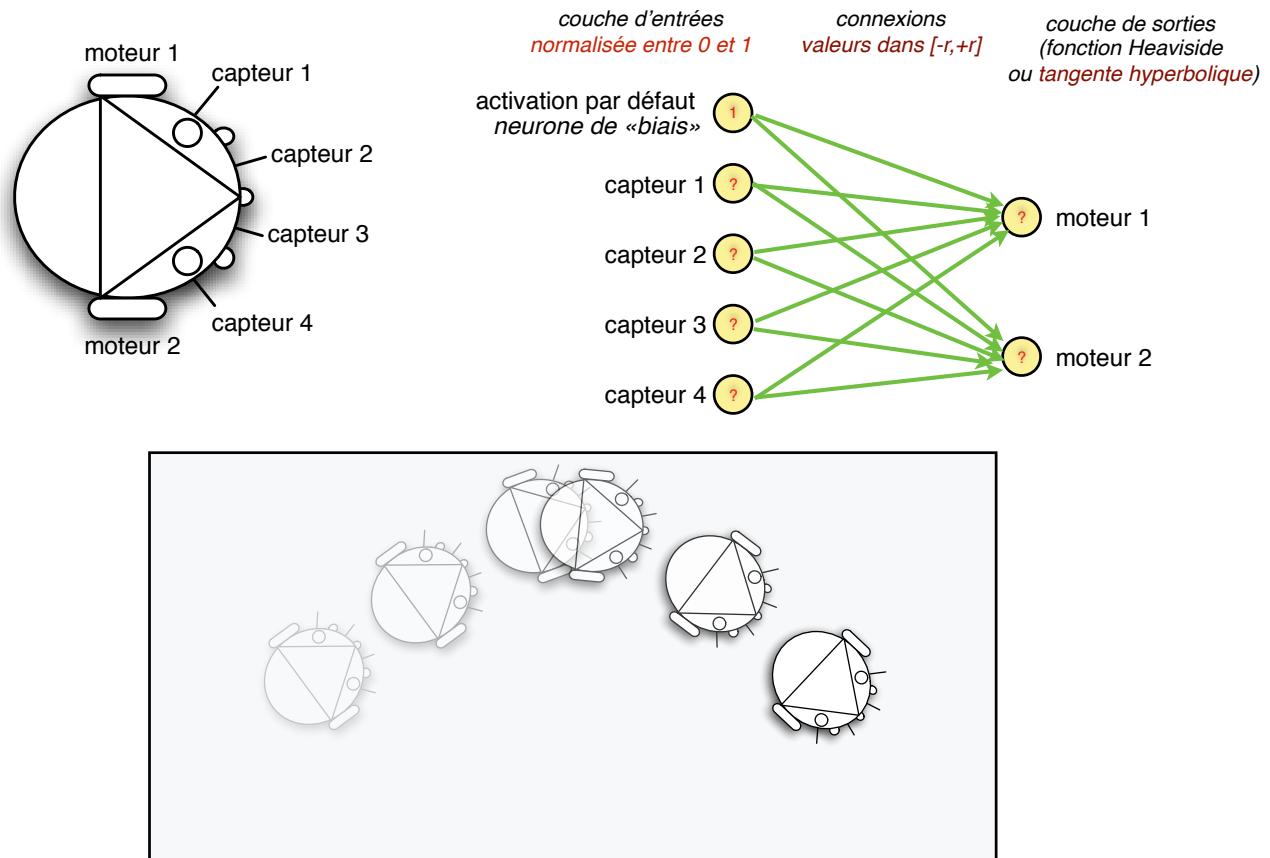
exemple pratique

Le problème du **max-one** (cas d'école)

- **Objectif** : maximiser la correspondance entre une chaîne de bits et une autre (ex.: [1,1,1,1,1,...,1])
- **Représentation d'un individu** : $\{0, 1\}^m$
- **Population initiale** : N individus
- **Opérateur de Sélection** : par tournoi de taille k
- **Opérateurs de Variation** : mutation bit-flip (probabilité/bits)

en rouge, les paramètres à régler

nicolas.bredeche@upmc.fr



cf.TME

nicolas.bredeche@upmc.fr

Etude de cas: (1+1)-ES, règle des 1/5ème

24

```

Initialize  $x \in \mathbb{R}^d$  et  $\sigma > 0$ 
while not terminate
     $x' = x + \sigma N(0, I)$ 
    if  $f(x') \leq f(x)$ 
         $x = x'$ 
         $\sigma = 2\sigma$ 
    else
         $\sigma = 2^{-1/4}\sigma$ 

```

Optimisation dans le domaine continu

nicolas.bredeche@upmc.fr

Famille d'algorithmes

26

- Points communs

- ▶ Opérateurs de sélection et remplacement
- ▶ Questionnement sur le compromis exploration-exploitation
 - trop d'exploitation: convergence prématurée
 - trop d'exploration: pas de convergence du tout

- Différences

- ▶ Représentations
 - bits, symboles, entiers, réels, arbres, graphes
- ▶ Opérateurs manipulant ces représentations
 - opérateurs unaires (ex.: *mutation*), n-aire (ex.: *recombinaison*)

algorithmes évolutionniste/évolutionnaire -- “evolutionary computation”

- **Algorithmes génétiques**

- ▶ Holland, 1975 (IA et Biologie)
- ▶ représentation: chaînes de bits

- **Stratégies d'évolution**

- ▶ Rechenberg, Schwefel, 1965 (Math Appl.)
- ▶ représentation: vecteur de réels

- **Programmation évolutionnaire**

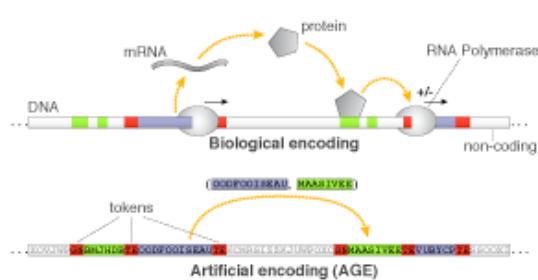
- ▶ Fogel, 1966
- ▶ représentation: Automates

- **Programmation génétique**

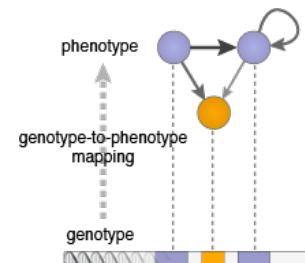
- ▶ Koza, 1992
- ▶ représentation: arbre, DAG, graphe, AdF

nicolas.bredeche@upmc.fr

Etude de cas: Analog Genetic Encoding



Génome: une séquence
de lettres (A-Z)



Réécriture sous la forme d'un
graphe (p.ex. réseau de neurones)

- Représentation
 - ▶ Vecteur de réels

- Opérateurs
 - ▶ Mutation gaussienne
 - ▶ Taux de mutation
 - Heuristique (ex.: (1+1) avec la règle des 1/5^{ème})
 - Auto-adaptatif

- Applications:
 - ▶ Optimisation paramétrique (fonctions, modèles)
 - ▶ Poids d'un réseau de neurones

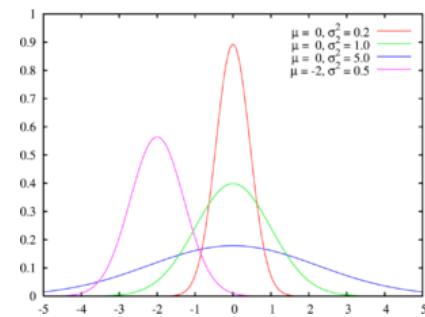
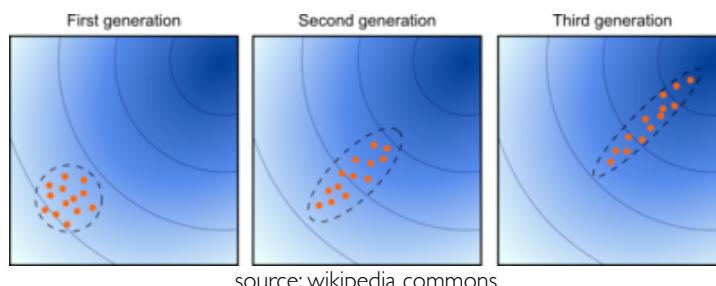


image: wikipedia commons

[Rechenberg, Schwefel, 1965]

Etude de cas: CMA-ES



source: wikipedia commons

- "Co-variance Matrix Adaptation ES"
 - ▶ path length control (cumulative step-size adaptation, CSA)
 - ▶ Matrice de covariance (estimation de la hessienne)
- Caractéristiques
 - ▶ recherche locale et globale
 - ▶ invariance p/r aux rotations/translations
 - ▶ pas/peu de paramètres à régler
 - ▶ nombre de dimensions: de 3 à 100+

- Représentation
 - ▶ Arbre, graphe, DAG
 - ▶ Fonctions et Terminaux
 - ▶ ADF
- Opérateurs
 - ▶ Croisement
 - ▶ (Mutation)
- Applications
 - ▶ Régression symbolique
 - ▶ Circuit électronique (multiplexer)
 - ▶ Construction de structures - cf. transparents suivants

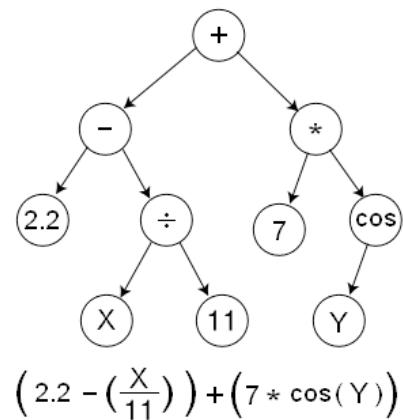


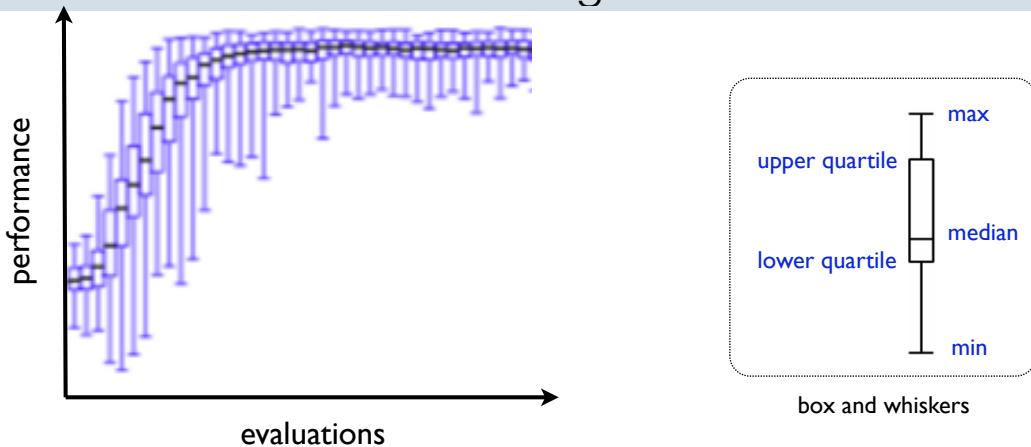
image: wikipedia commons

Conclusions

- Ce qu'il faut retenir
 - ▶ Algorithmes d'optimisation stochastique à base de population
 - ▶ Compromis exploitation-exploration
 - ▶ Espace de recherche (discret, continu, structuré)
 - ▶ Fonction objectif
 - ▶ Application à l'optimisation de la prise de décision
- Application dans le cadre du projet
 - ▶ Définir une fonction objectif
 - ▶ Définir un espace de recherche

Bonnes pratiques

- Critère d'arrêt
 - ▶ atteinte de l'optimum (si connu)
 - ▶ budget (en nombre d'évaluation)
 - ▶ heuristique/empirique (convergence/stagnation observée)
 - ▶ perte de diversité
- Validité des résultats
 - ▶ Il s'agit d'un algorithme stochastique
 - ▶ Il faut une mesure statistique de la performance
 - en pratique, sur N runs
 - median, min, max, quartile... ou moyenne, écart type



- A retenir:

- ▶ Médianes plutôt que moyennes, évaluations plutôt que générations
- ▶ Répéter les expériences, poursuivre jusqu'à convergence
- ▶ Donner les résultats pour les meilleures perf's et pour les moyennes
 - meilleures performances au mieux: le choix de la solution
 - meilleures performances en moyenne: le choix de l'algorithme

Il s'agit d'un exemple à fin d'illustration

Méthodologie 3/3

- Choix d'un algorithme

- ▶ objectif conception
 - trouver la meilleure solution
- ▶ objectif production
 - trouver l'algorithme le plus stable

Fin du cours