



# **Les Algorithmes Génétiques (AGs)**

**A. BENGHENI(2019/2020)**

*Université IBN Khaldoun - TIARET – Faculté MI –  
Département informatique*

# Plan

- Introduction
- La description des Algorithmes Génétiques (AGs)
- Mise en place des éléments de l'algorithme génétique
- La description des opérateurs génétiques
- Conclusion

# Introduction

## (Les Algorithmes Génétiques (AGs))

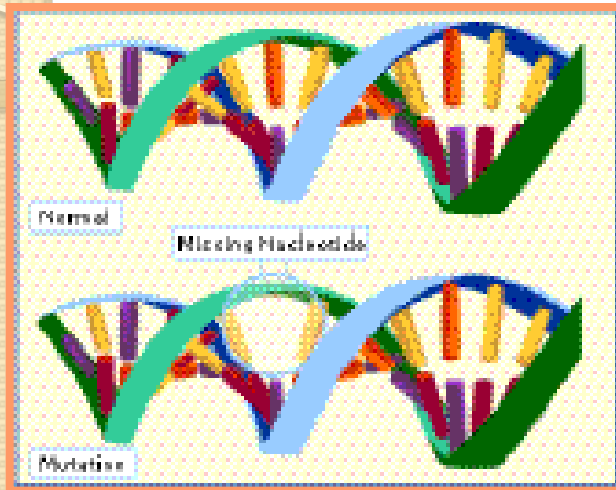
- Introduit par J.H. Holland et son équipe du Michigan
- Ses travaux ont commencé en 1960 et en 1975, il publia le premier ouvrage ayant comme titre « **Adaptation in Natural and Artificial System** »
- Objectifs : améliorer la compréhension des processus naturels d'adaptation, et concevoir des systèmes artificiels possédant des propriétés similaires aux systèmes naturels
- Application dans tous les domaines de l'optimisation

# Introduction (suite)

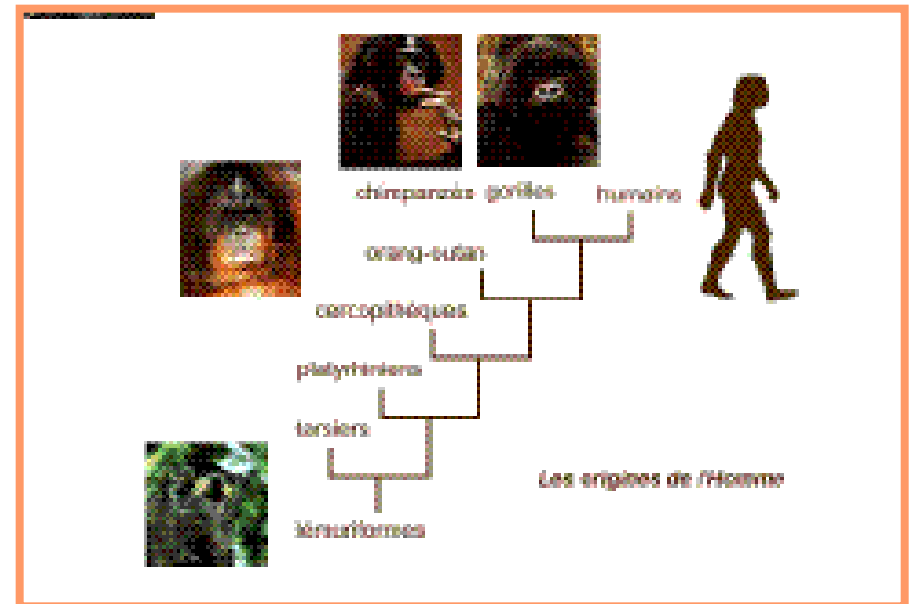
## (Les Algorithmes Génétiques (AGs))

### Principe

- Principe original : Principe du **Crossing Over** et de la **mutation**



- Charles Darwin publie en 1859 « Origin of species » qui explique le mécanisme qui peut présider à l'évolution **des espèces** dans la nature



# Introduction (suite)

## (Les Algorithmes Génétiques (AGs))

### Théorie de darwin

- **La sélection naturelle** : Sélection des individus les mieux « adaptés » à un milieu donné et qui auront une plus grande faculté de reproduction que les autres
- **La sélection naturelle** soutient donc que les êtres vivants qui s'adaptent le mieux aux conditions naturelles de leur environnement **vaincront et survivront**

# La description des Algorithmes génétiques (AGs)

## Pourquoi les AGs ?

- Les AG ne permettent pas d'obtenir assurément une solution optimale exacte, mais plutôt une solution de qualité et ça en peu d'effort.
- Espaces de recherche importants
- Pas d'algorithmes déterministes adaptés

# La description des Algorithmes génétiques (AGs)

## Lexique

- **Population** (= génération): Ensemble de chromosomes
- **Chromosome** (= un individu) : Ensemble de gènes, de caractéristiques d'un individu
- **Gène**: Caractère / caractéristique d'un individu
- **Allèle**: Forme / valeur prise par une caractéristique
- **Exemple** : algériens, 1 algérien, yeux, yeux noirs

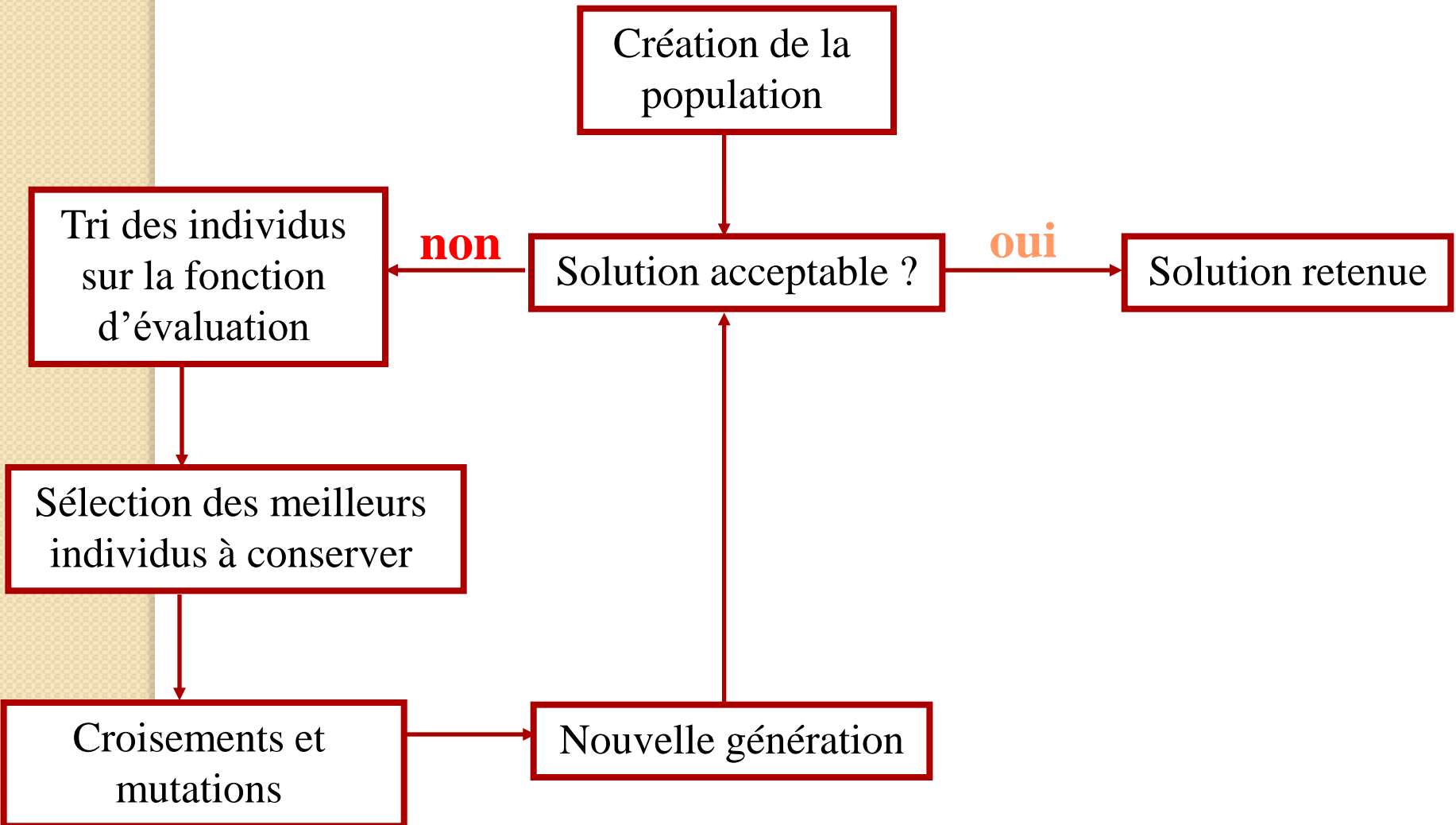
# La description des AGs

## Lexique

Vocabulaire	Mathématique
Génôme	Ensemble des points de l'espace de recherche
population	Ensemble "particulier" de points de l'espace de recherche
Individu	Point de l'espace de recherche "particulier"
Chromosome	Ensemble de variables caractérisant un point de l'espace
Gène	Variable
Allèle	Valeur possible de variable



# L'algorithmes Génétiques



# L'algorithmes Génétiques

## Préparation

### **Mise en place des éléments de l'algorithme:**

Codage des données

Paramétrage

Fonction d'évaluation

Critère d'arrêt

Population initiale

# Codage binaire

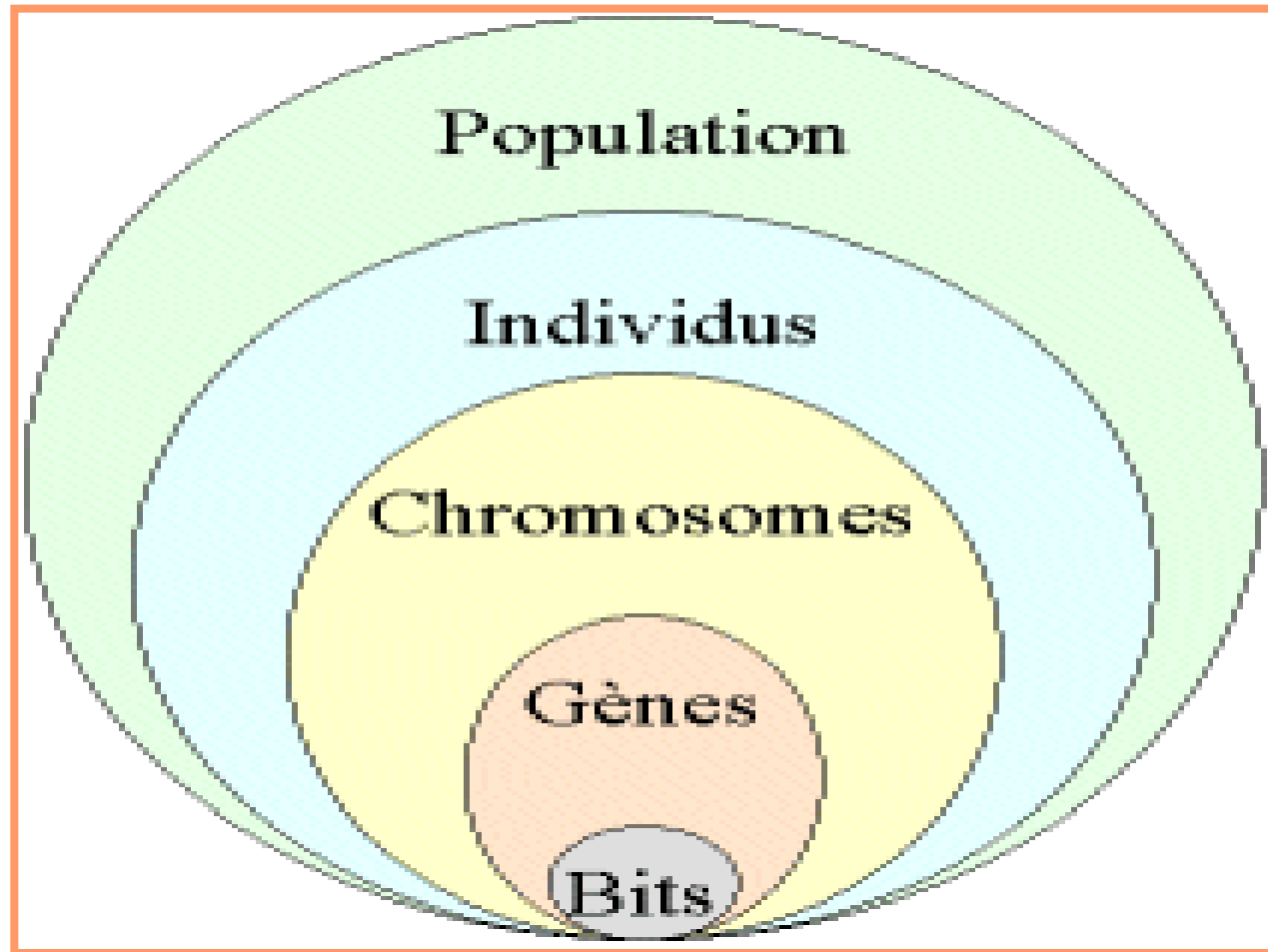
- très utilisé
- 1 gène = 1 variable  $X_i$  = chaîne de bits

Ex: Chromosome A

| 0 | | 0 0 | 0 | | 0 0 | 0 | 0 | | | 0 0 | 0 |

Chromosome B

| | | | | | | 0 0 0 0 0 | | 0 0 0 0 0 | | | | |



les cinq niveaux d'organisation des  
Algorithmes Génétiques

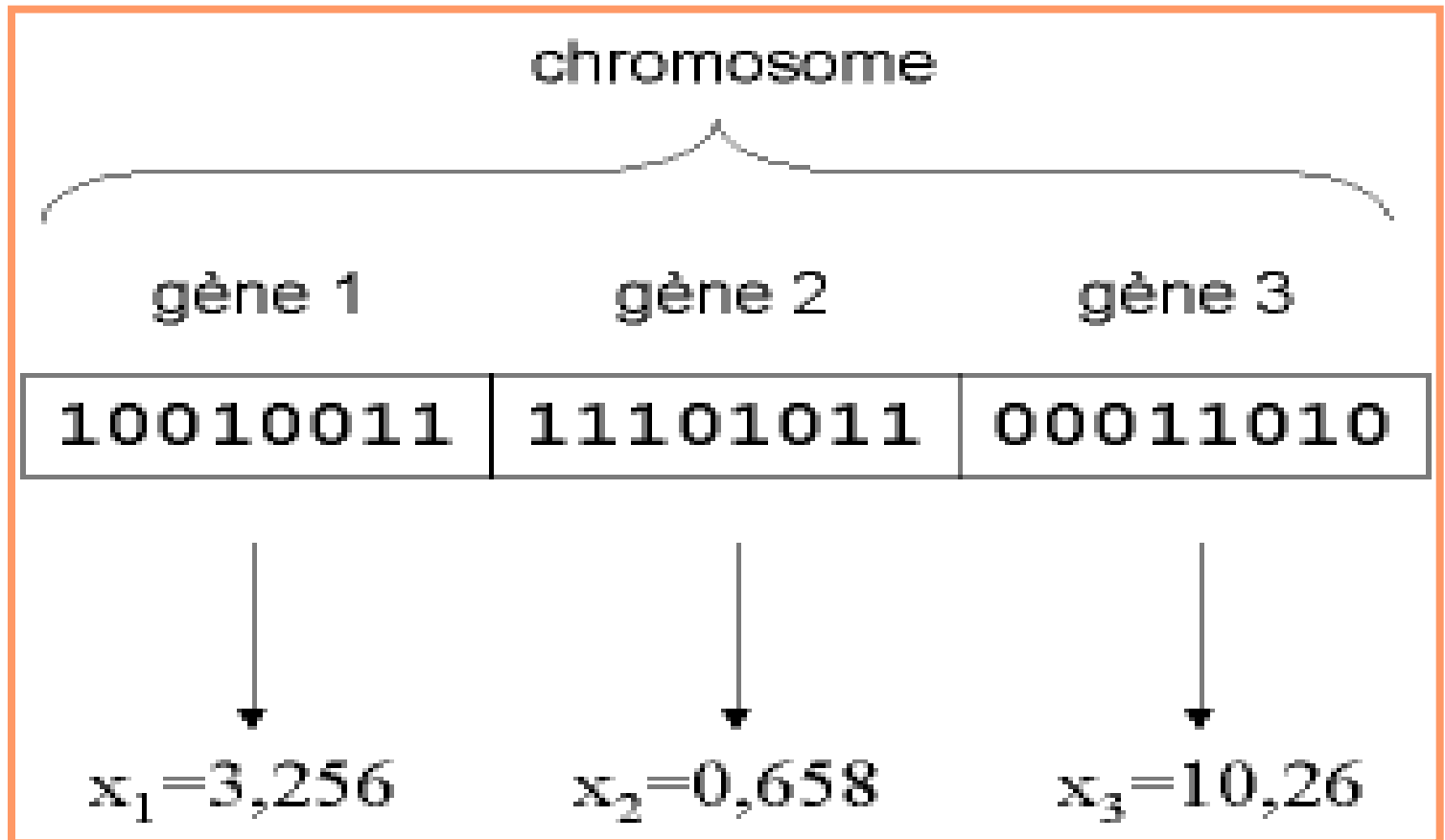


illustration schématique du codage  
des variables d'optimisation  $x_i$ .

# Codage réel

plus naturel

1 gène = 1 variable  $X_i$

allèle = 1 valeur du domaine de définition de  $X_i$

Ex: Chromosome A

1.2324 5.3243 0.4556 2.3293 2.4545

Chromosome B

ABDJEIFJDHDIERJFDLDFLFEGT

# Paramétrage

- Taille d'un groupe (nombre d'individus) ?
- Le paramétrage doit-il changer en fonction de la phase de calcul (par ex: plus de mutations à la fin...)
- Certaines fonctions sont-elles plus efficaces avec une taille de population faible ou élevée ?
- Dans le crossover, faut-il préférer la rapidité (afin de faire plus d'itérations) ou des opérateurs plus sophistiqués mais plus longs ?
- Faut-il utiliser une heuristique pour initialiser la population?

# Fonction d'évaluation

- fonction objectif = « fitness »
- Mesure l'adaptation des individus à un milieu donné
- Maximisation / minimisation en fonction du but recherché
- critère de sélection



# Critère d'arrêt

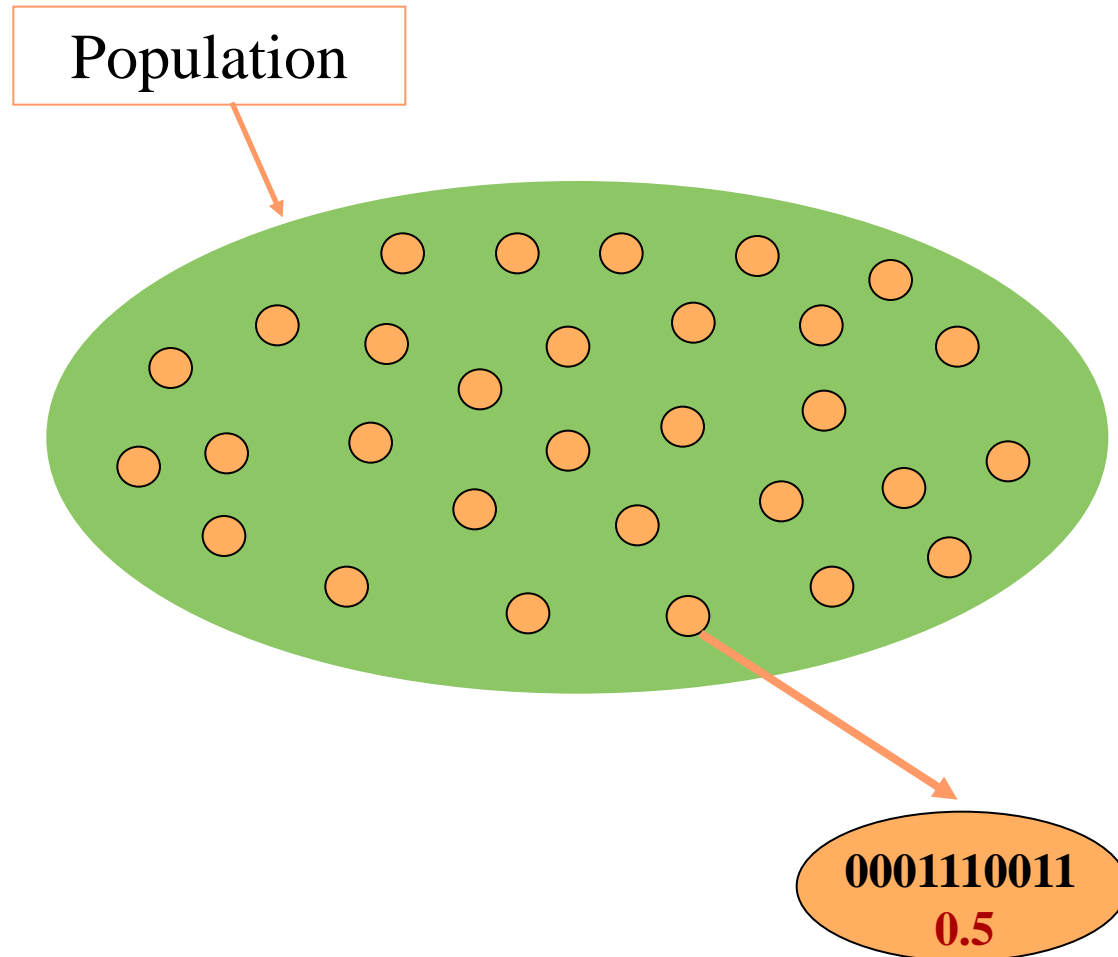
- Nombre de générations fixé
- Suivi des valeurs des meilleurs éléments de la population: s'il ne s'est pas amélioré les  $x$  dernières générations, on s'arrête.
- Suivi de la différence d'évaluation entre 2 meilleurs individus: si elle est inférieure à une certaine tolérance, on s'arrête.



# Population initiale

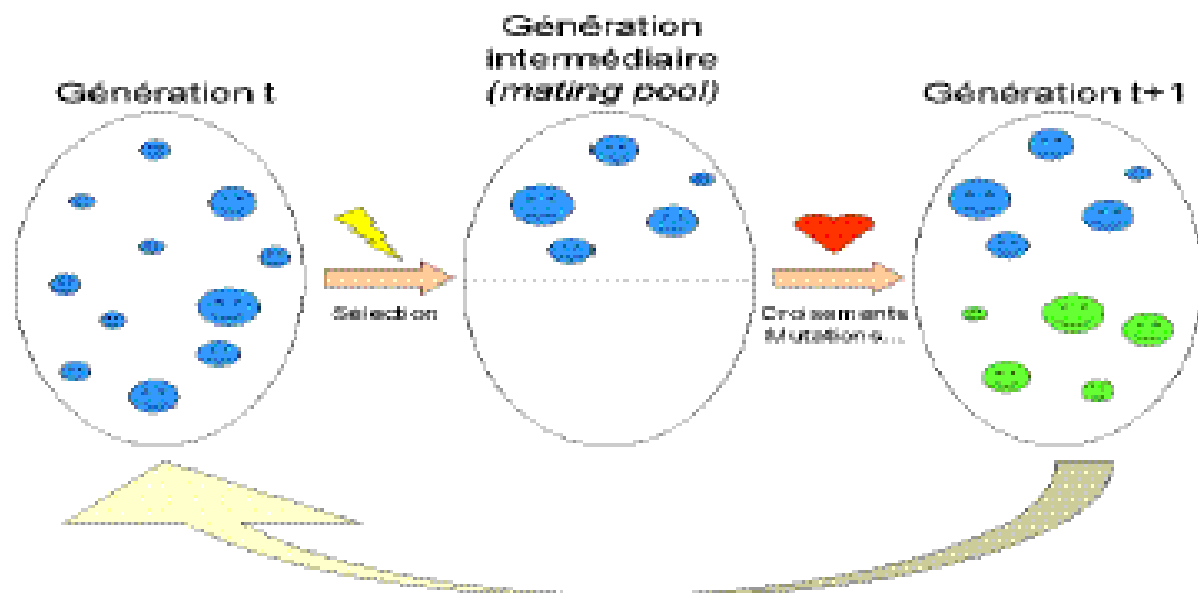
- Générée aléatoirement
- Doit respecter les contraintes du problème
- Possibilité d'utiliser un ensemble de solutions données par un autre algorithme

## Exemple



# Les opérateurs

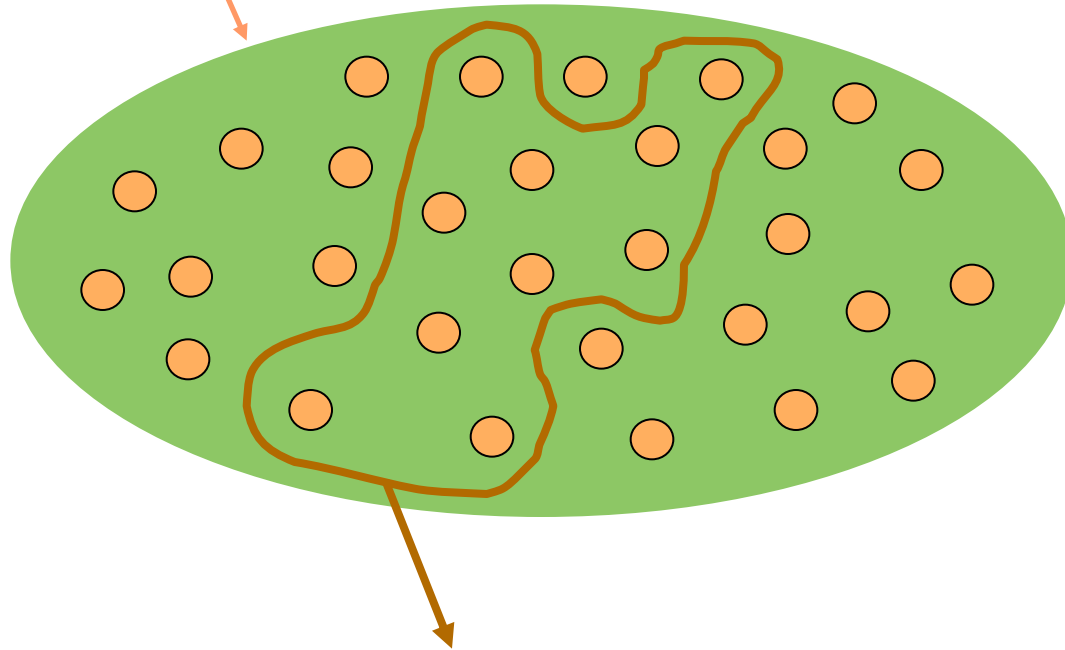
- Sélection
- Mutation
- Croisement / cross-over



# sélection

- A partir de la génération  $k$
- Fonction du critère d'adaptation afin de privilégier les individus adaptés au milieu
- Méthodes de sélection
  - Roulette de casino
  - $N/2$  élitisme
  - Par tournoi

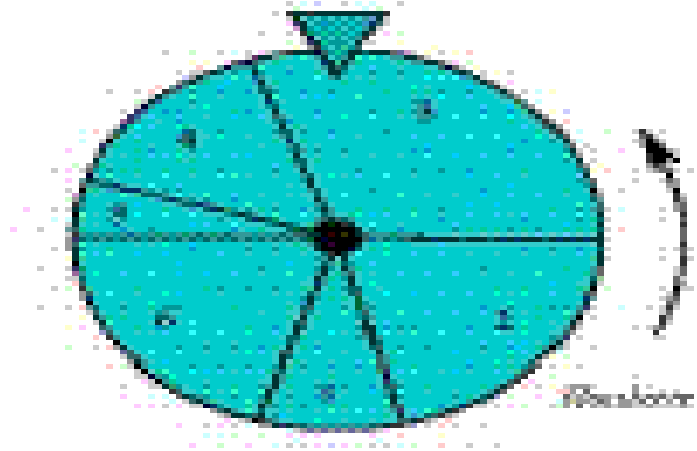
Population P



Sélection des 10 meilleurs individus

# Roulette de Casino

- La plus utilisée
- Chaque chromosome occupe un secteur de roulette dont l'angle est proportionnel à l'adaptation
- la largeur du secteur sera d'autant plus importante que l'indice de qualité le sera !
- plus le secteur est important plus les chromosomes auront de chance d'être sélectionnés

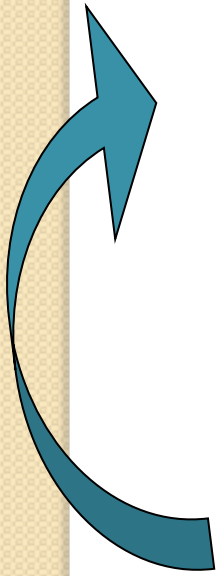


# N/2 élitisme

- Individus triés selon leur fonction d'adaptation
- Seule la  $\frac{1}{2}$  supérieur de la population = meilleurs composants seront sélectionnés
  - > risque de convergence rapide



# par tournoi

- 
- Choix aléatoire de 2 individus
  - « combat » = évaluation des 2 chromosomes
  - Sélection du meilleur

# Croisement

- **Crossover**: opération où, les individus les plus adaptés ayant plus de chances d'y participer.
- **Crossover uniforme**: chaque gène (*bit*) d'un fils est une copie du gène correspondant d'un de ses parents choisi au hasard.

Nouvelle population P

Individu X

1011001011

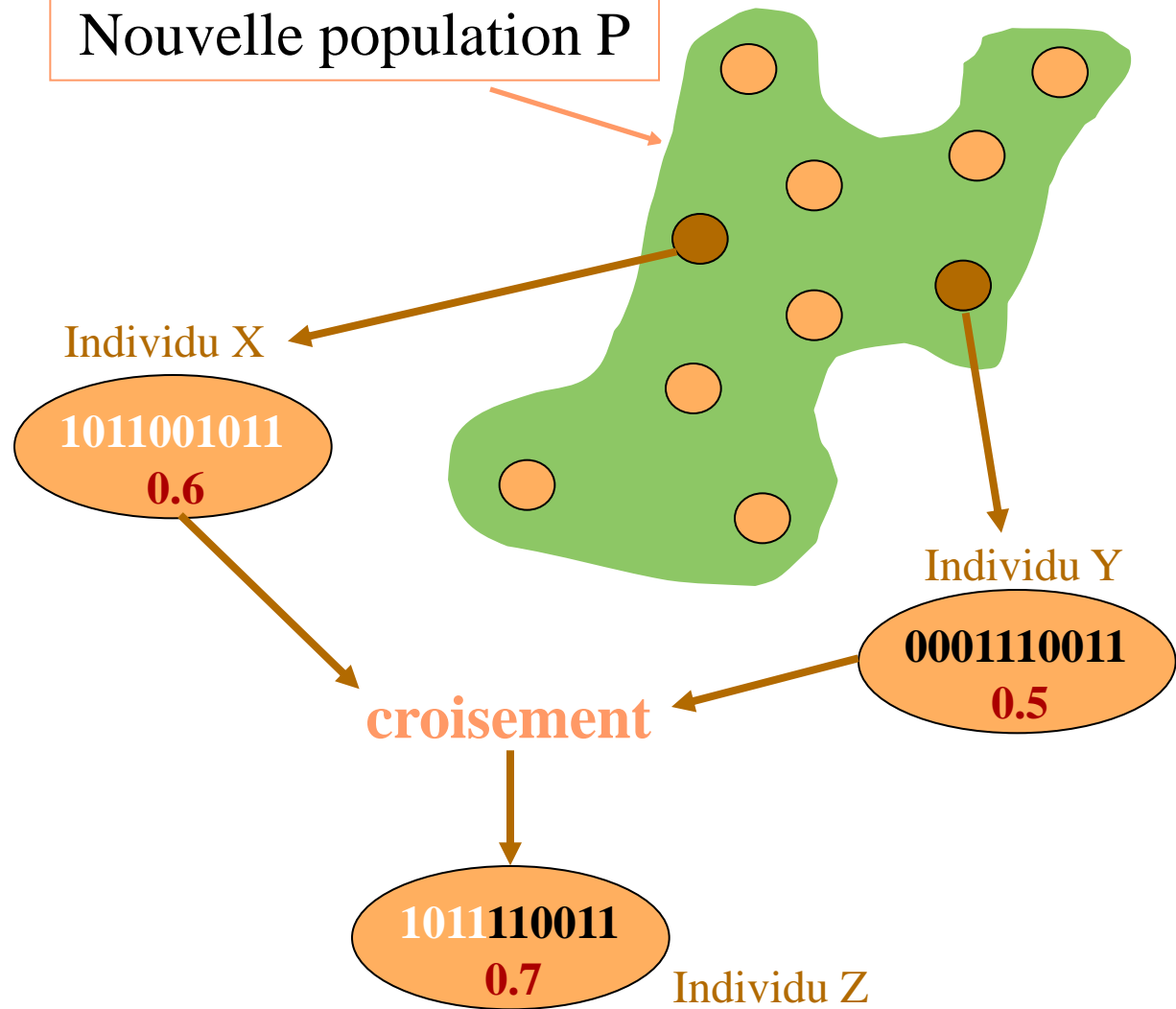
0.6

Individu Y

0001110011

0.5

Nouvelle population P



- *Crossover d'un seul point:*

parents: 2 chromosomes de n bits

$$a = (a_1, \dots, a_k, \dots, a_n) \quad b = (b_1, \dots, b_k, \dots, b_n)$$

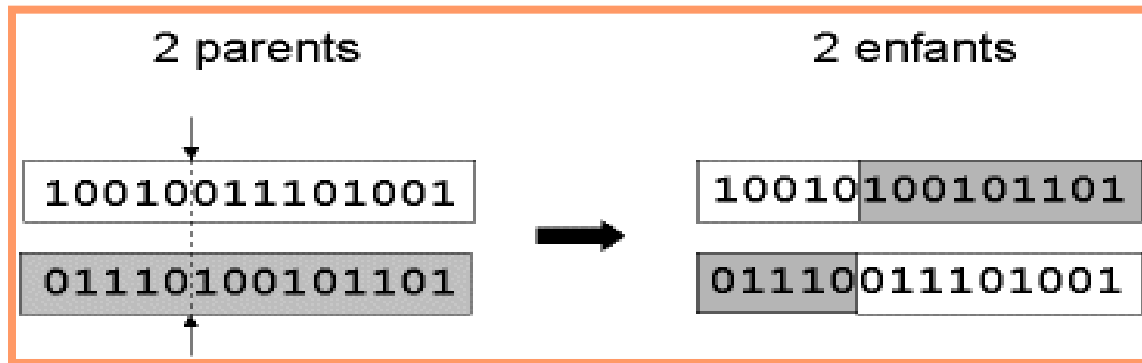
opération:  $k \in \{1, \dots, n\}$  choisi au hasard

filis:

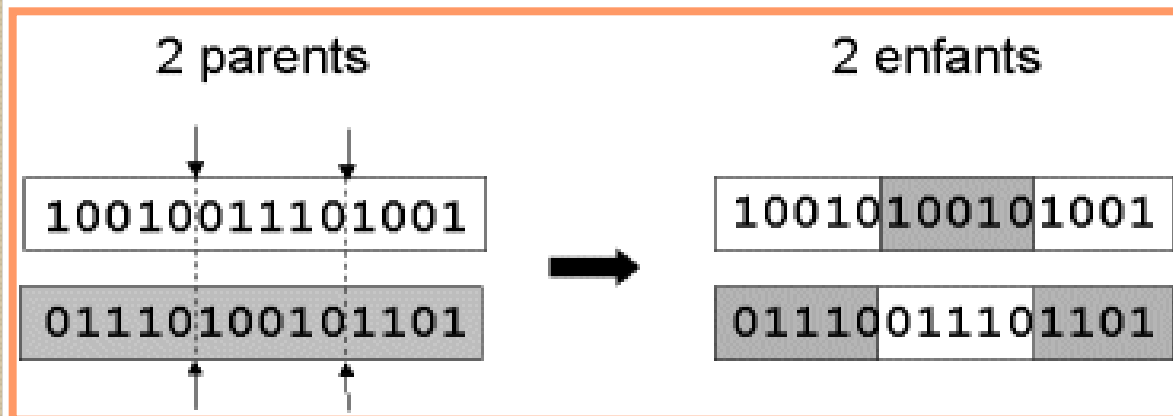
$$(a_1, \dots, a_k, b_{k+1}, \dots, b_n) \quad (b_1, \dots, b_k, a_{k+1}, \dots, a_n)$$

- *Crossover de deux (ou plus) points*

- *Crossover par fusion:* comme le premier (uniforme), mais la probabilité de choix de chaque parent est proportionnelle à la valeur de sa fonction d'adaptation



Croisement  
en un point



Croisement  
en 2 points

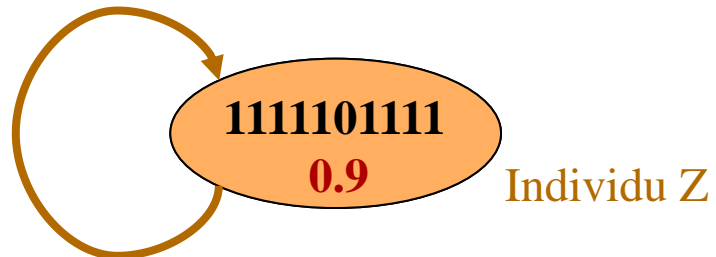
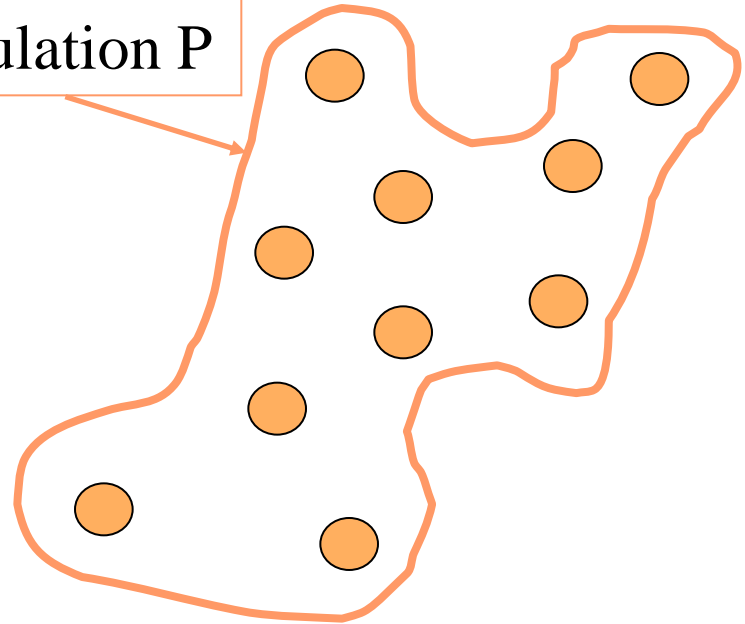
- **Difficulté:** comment faire pour traiter les contraintes et les solutions qui ne sont pas réalisables?
  - utiliser une représentation qui puisse garantir automatiquement que toutes les solutions représentables soient réalisables
  - utiliser un opérateur heuristique qui puisse transformer chaque solution non-réalisable en une autre qui soit réalisable
  - utiliser une fonction de pénalisation pour modifier la valeur de la fonction d'adaptation de toute solution non-réalisable

# Mutation

- **Mutation**: normalement implémentée comme la complémentation de bits des individus de la population.
- **Sélection au hasard des bits à modifier**: très faible fraction des individus de la population
- Les mutations ne sont pas soumises aux tests d'adaptation: la reproduction mène l'évolution à une population homogène. Les mutations permettent d'introduire un peu de diversité dans la population.
- **Mutation statique** (probabilité constante de changer un *bit*)
- **mutation adaptative** (appliquer l'opérateur de mutation sur quelques *bits* sélectionnés, pour garantir l'obtention de solutions réalisables)



Nouvelle population P



- Exemple:

Crossover avec  $k = 4$

Parent 1 = (0,1,0,1,1,0,1,1)

Parent 2 = (1,0,0,0,0,1,1,0)

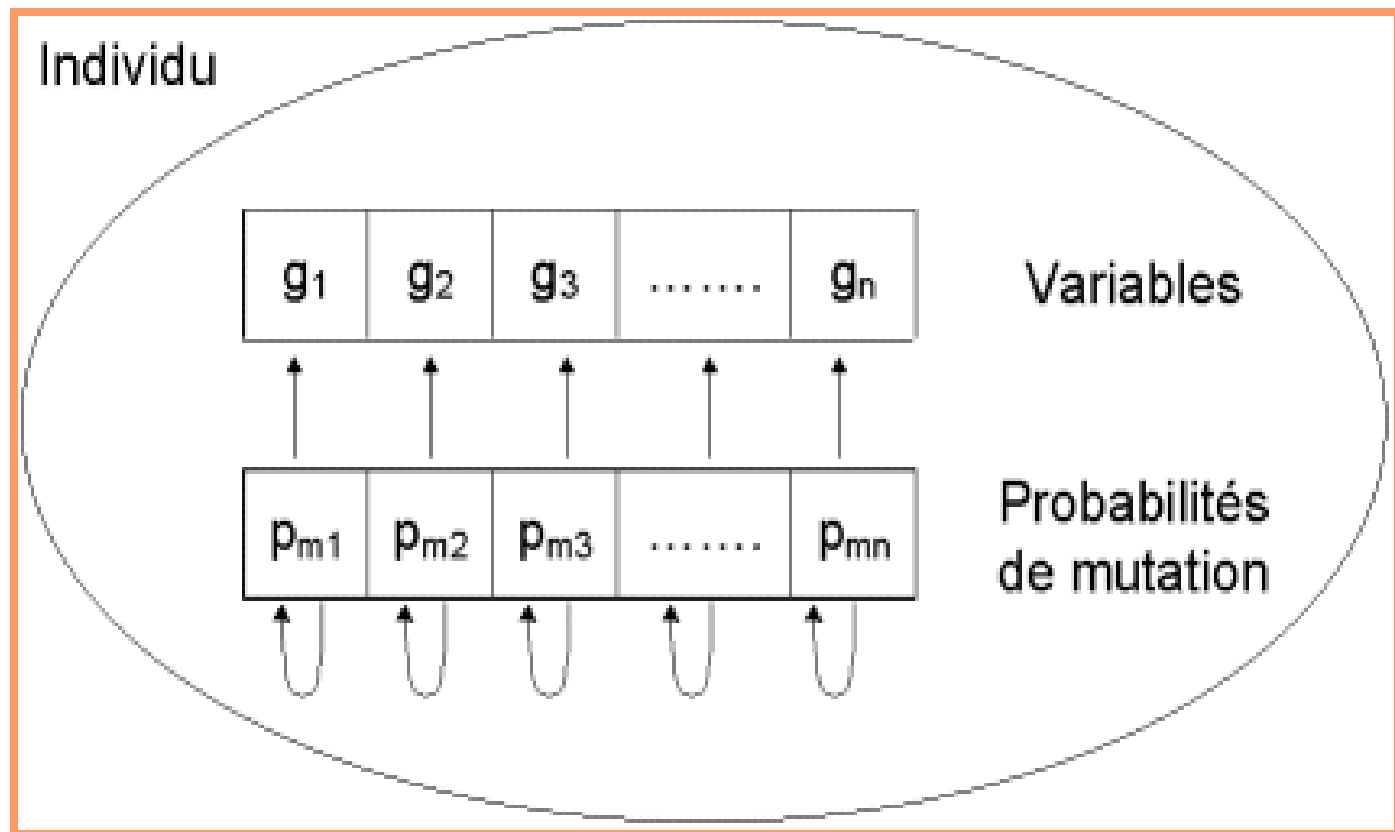
Fils 1 = (0,1,0,1,0,1,1,0)

Fils 2 = (1,0,0,0,1,0,1,1)

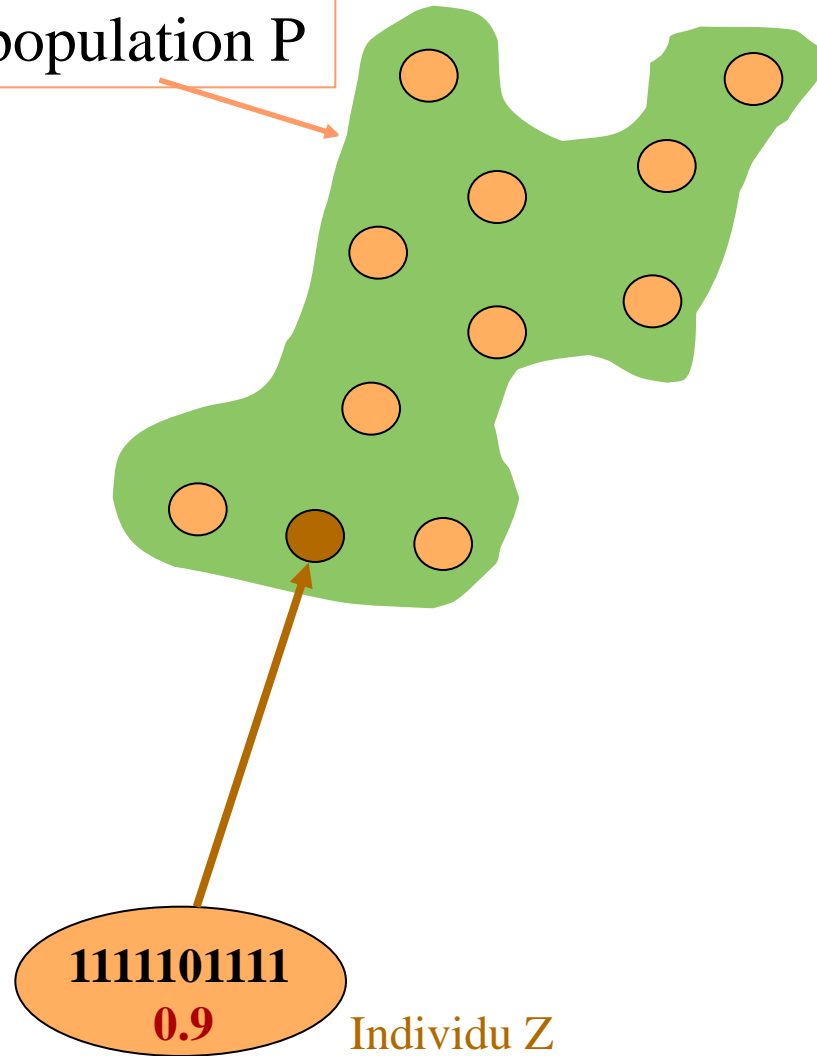
Mutation du 5ème. *bit*:

(1,0,0,0,1,0,1,1)  $\rightarrow$  (1,0,0,0,0,0,1,1)

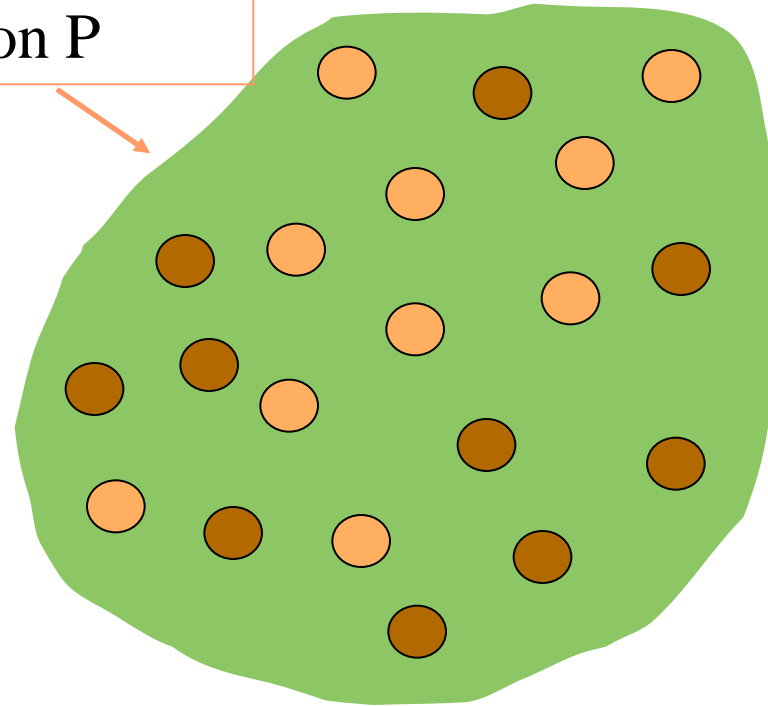
## Comment est réalisée la mutation?



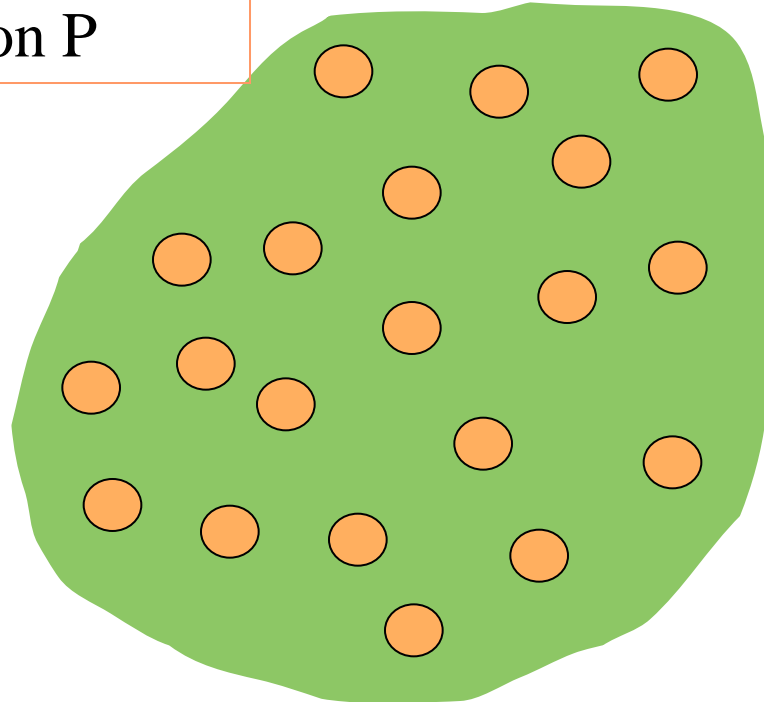
Nouvelle population P



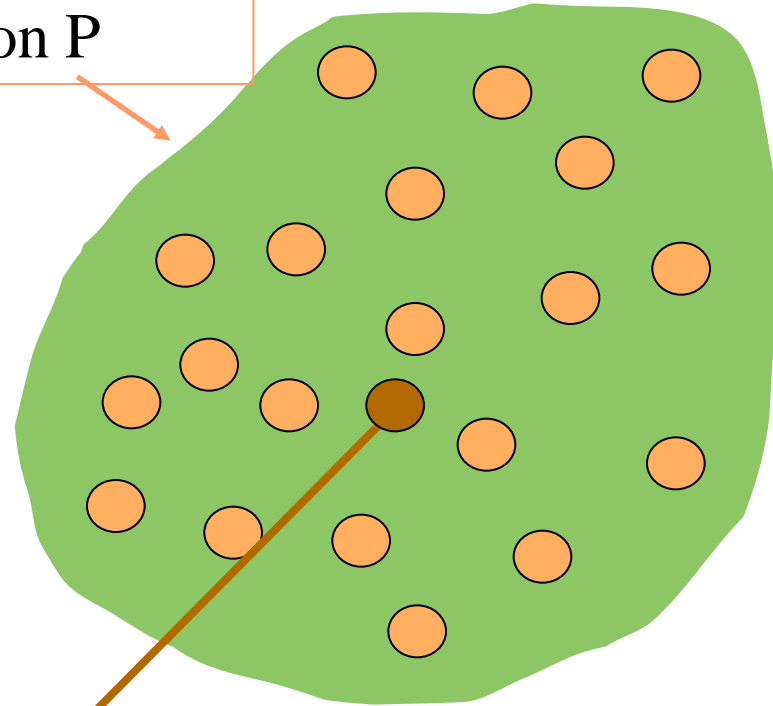
Population P



Population P



Population P



Meilleur Individu

111111111

1.0

## Cours et renseignements:

- <http://www.rennard.org/alife/french/gavintr.html>
- [http://fr.wikipedia.org/wiki/Algorithme\\_g%C3%A9n%C3%A9tique](http://fr.wikipedia.org/wiki/Algorithme_g%C3%A9n%C3%A9tique)
- <http://www.eudil.fr/%7Evmagnin/coursag/>
- <http://cs.felk.cvut.cz/~xobitko/ga/>

## Sur la programmation de l'algorithme génétique:

- <http://www.genetic-programming.org/>
- <http://www.aic.nrl.navy.mil/galist/src/#C> (sources en différents codes)
- <http://web.cps.msu.edu/~punch/> (GARAGe)

17 novembre 2004 MTH 6414 - Algorithmes Génétiques 54

Bonne solution rapidement plutôt qu'une solution optimale en une durée indéfinie !