

# Algorithmique génétique

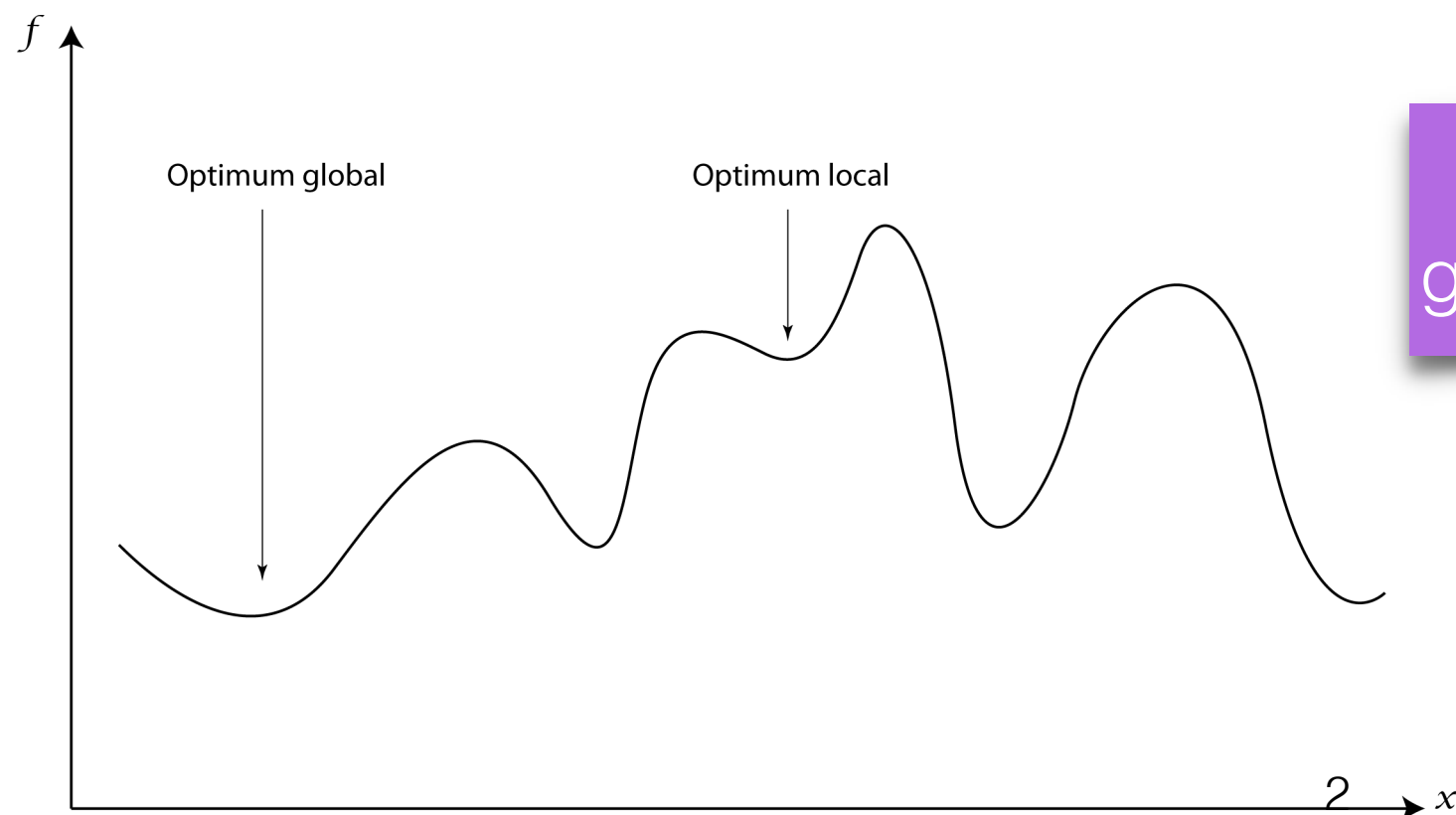
Module Algo-Prog  
Parcours Recherche

Joan Hérisson

ÉCOLE CENTRALE PARIS

# Optimisation

- Optimiser (minimiser/maximiser) une fonction
  - Parcours, emplois du temps, traitement d'image...
- Explorer un espace de recherche



petit → méthode déterministe  
grand → méthode stochastique

# Optimisation

- « Essai et erreur »
  - Solutions admissibles
  - Solutions coût optimal local

- Analyser le problème

- variables

**petit** si le système est simple  
si le concepteur sait quelles variables  
faire évoluer

**grand** si le système est complexe  
si le concepteur ne sait pas quelles variables  
faire évoluer

- espace de recherche

**infini** → **fini**

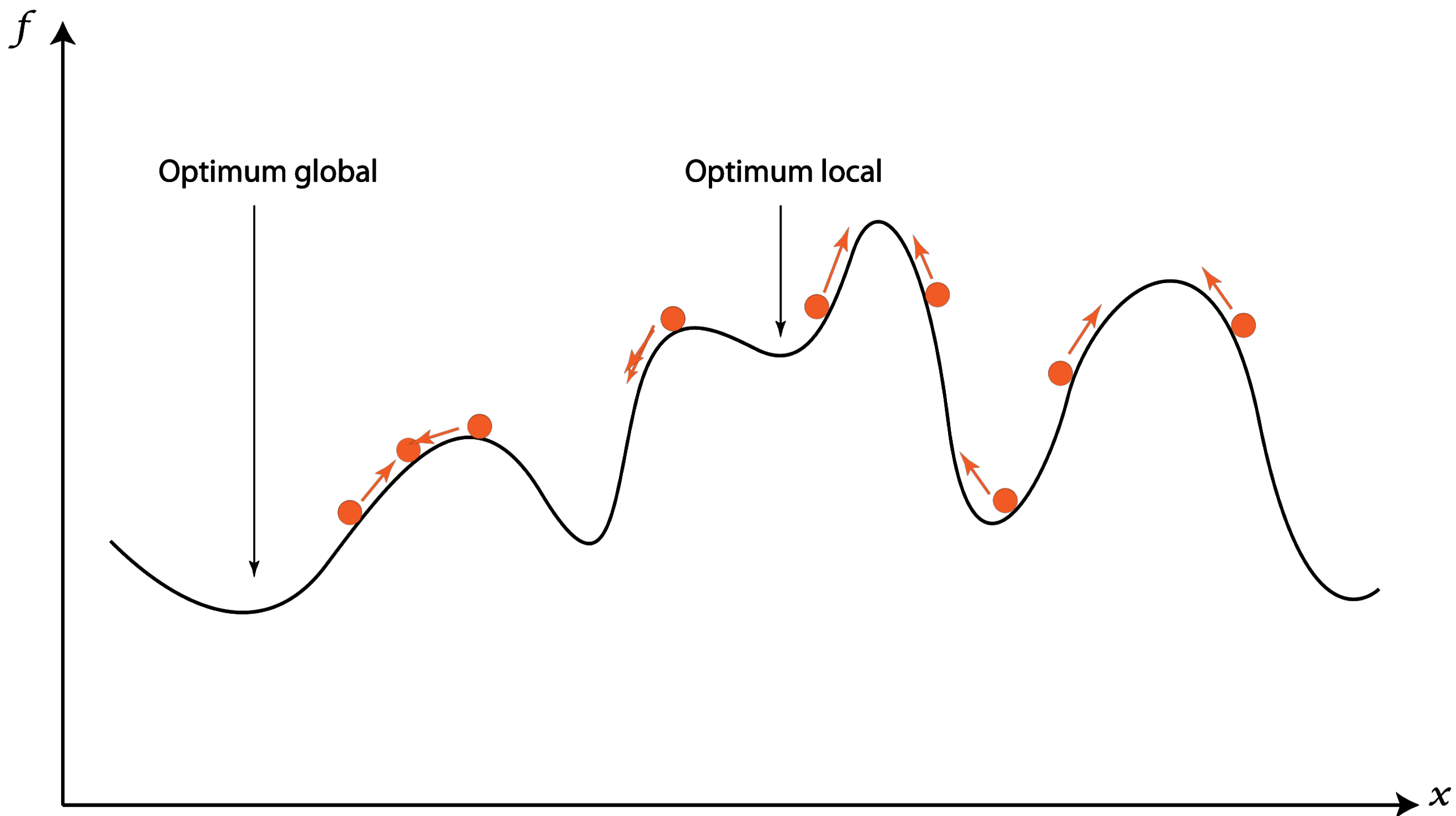
- objectif

**fitness** → **mesure**

- méthode d'optimisation

**déterministe**

**non-déterministe**



non-déterministe

stochastique

répétition

simple

# Monte-Carlo

# Monte-Carlo

## Estimation de la surface d'un lac

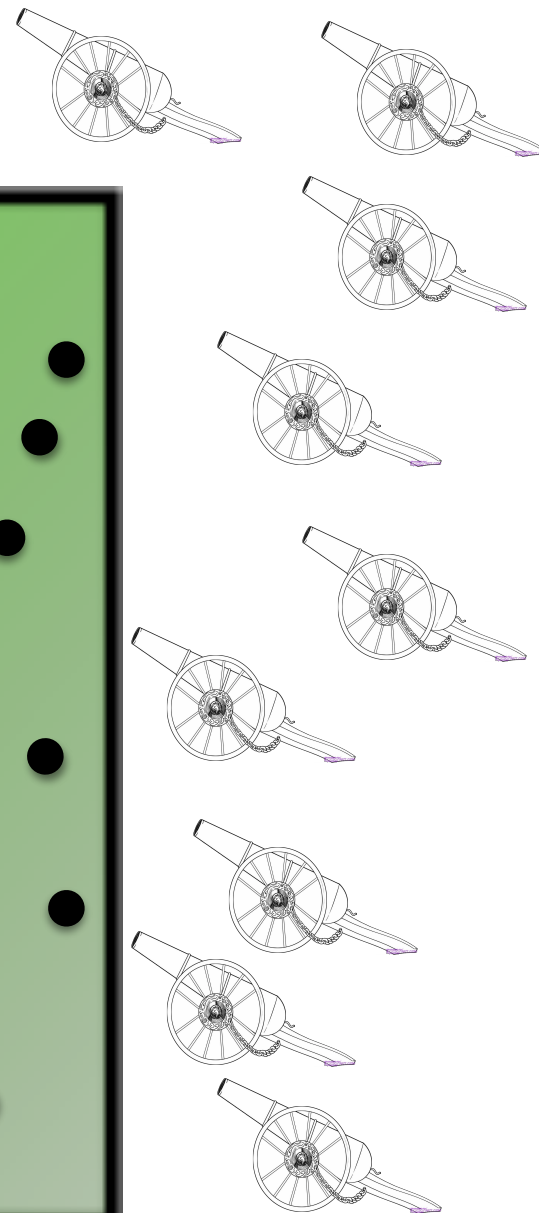
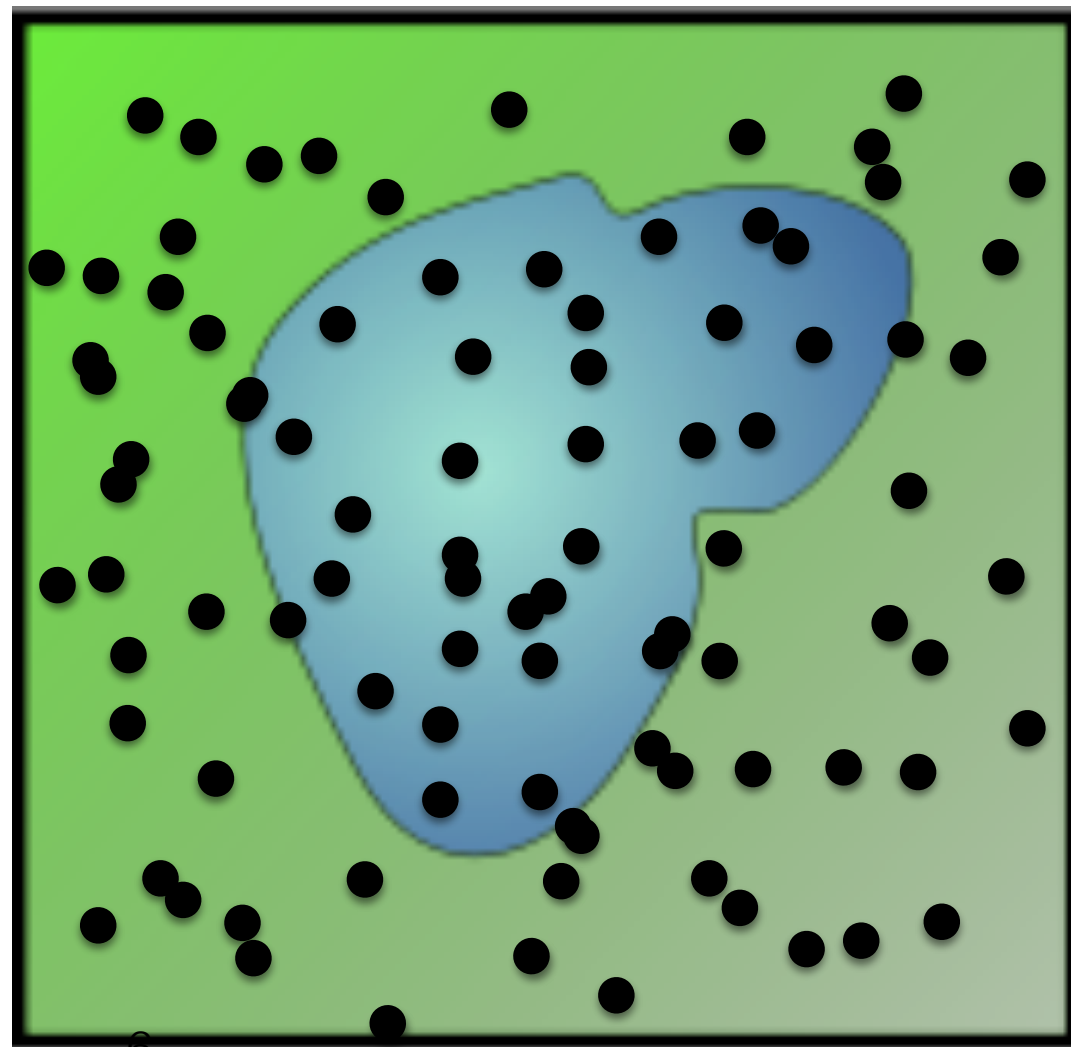
1. Tracer un terrain dont on connaît la superficie autour du lac
2.  $X$  tirs au hasard
3.  $N$  boulets sur le terrain
4.  $X \cdot N$  boulets dans le lac

$$\frac{SUPERFICIE_{terrain}}{SUPERFICIE_{lac}} = \frac{X}{X - N}$$

$$\Rightarrow SUPERFICIE_{lac} = \frac{X - N}{X} \times SUPERFICIE_{terrain}$$

**Volume des données**

**Qualité du tirage aléatoire**



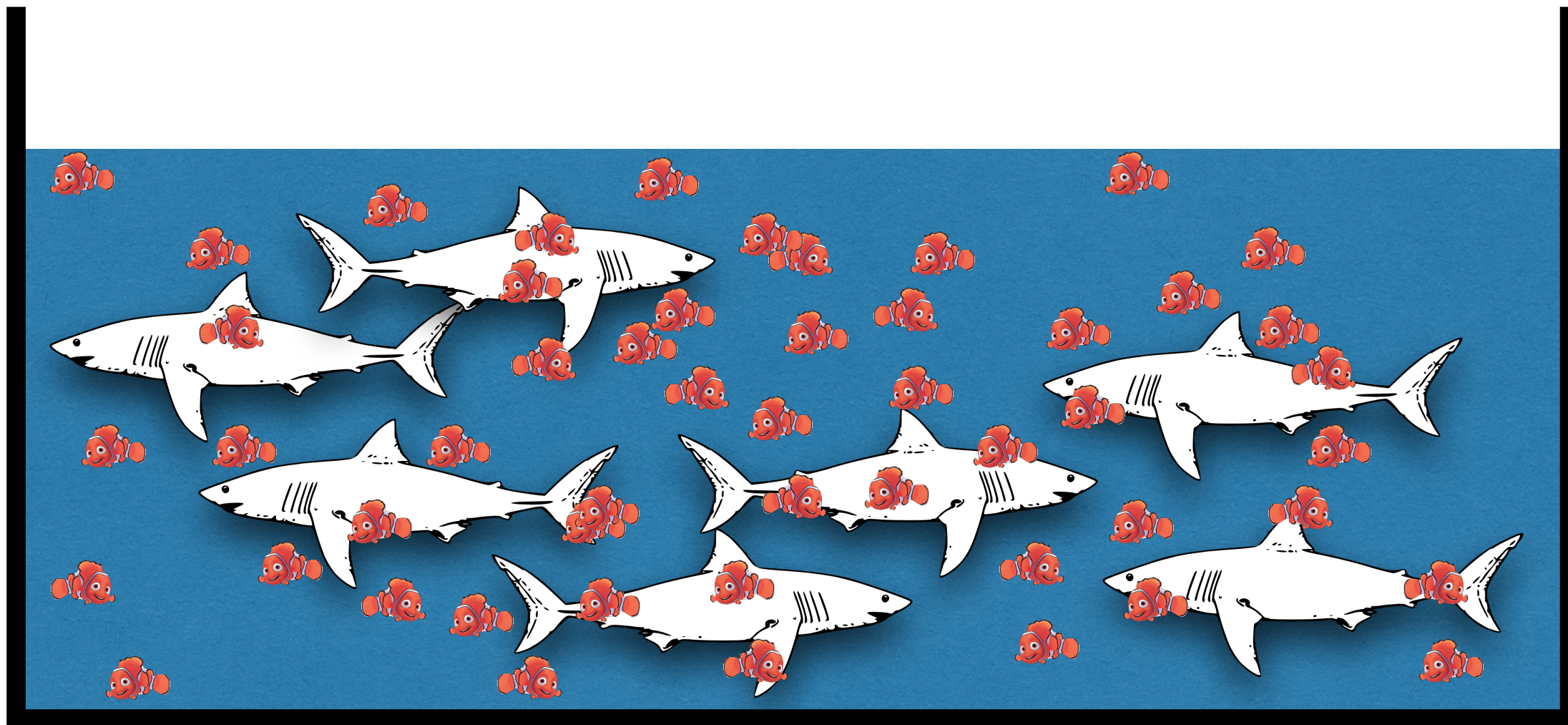
# Biologie

- Étude du vivant
- Spectre très large
  - Formes très variées, nombreuses échelles

*En quoi l'étude du vivant peut-elle nous aider à résoudre  
un problème d'optimisation ?*

- Le vivant s'adapte constamment à son environnement

*Optimisation de la fonction « survie »*





# Algorithme génétique

## **CONSTAT 1**

espace de recherche grand = temps de calcul grand

## **CONSTAT 2**

vivant optimise la fonction « survie »

## **PROPOSITION**

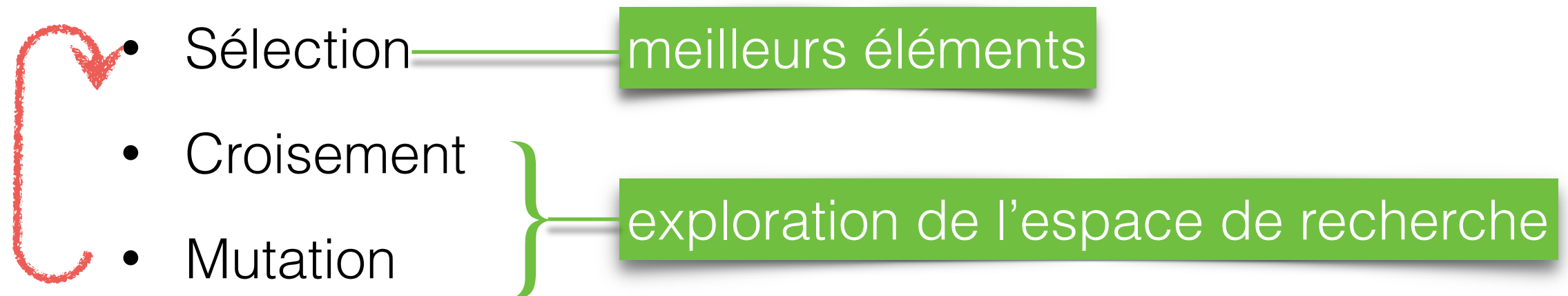
s'inspirer du vivant

pour résoudre des problèmes d'optimisation

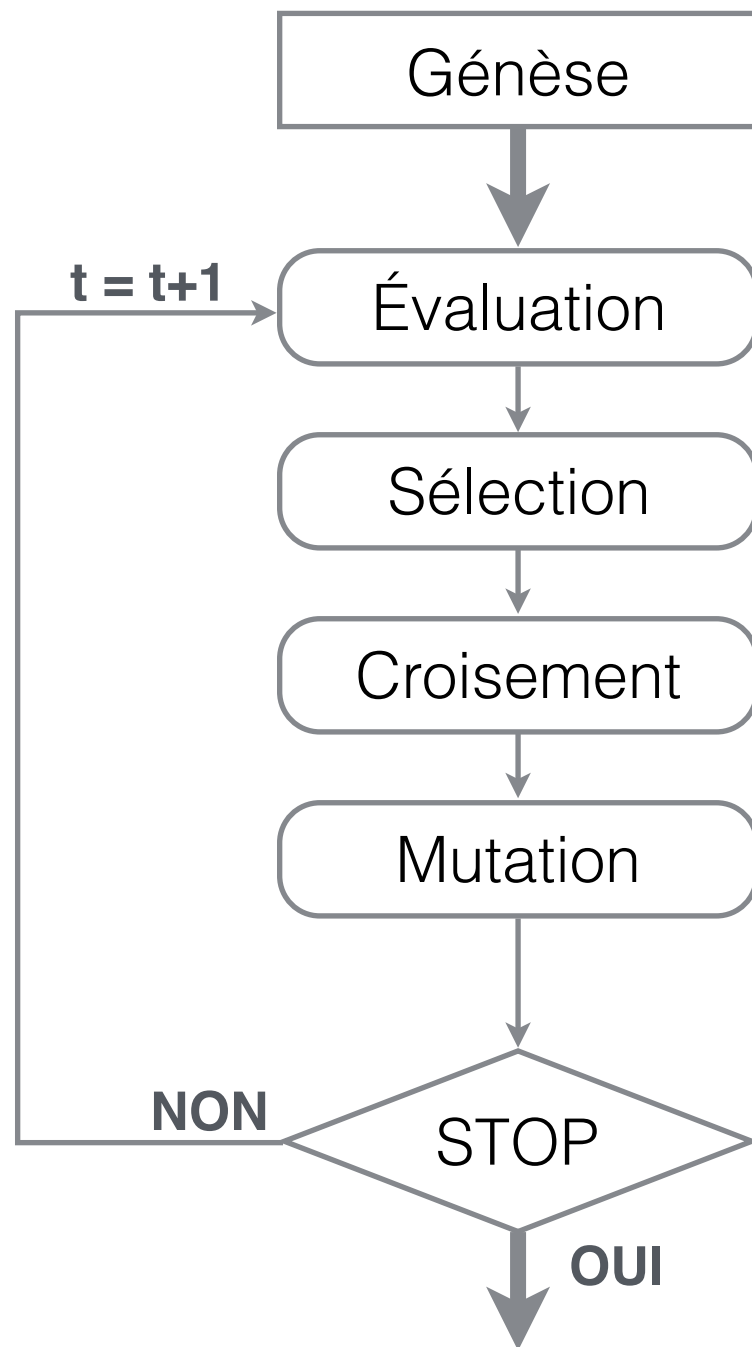
# Algorithme génétique

- Individu = point dans l'espace de recherche
- Valeur du critère à optimiser (adaptation)

- Populations



- Arrêt 
  - nombre d'itérations
  - convergence



Aucune connaissance

Solution proche de l'optimale

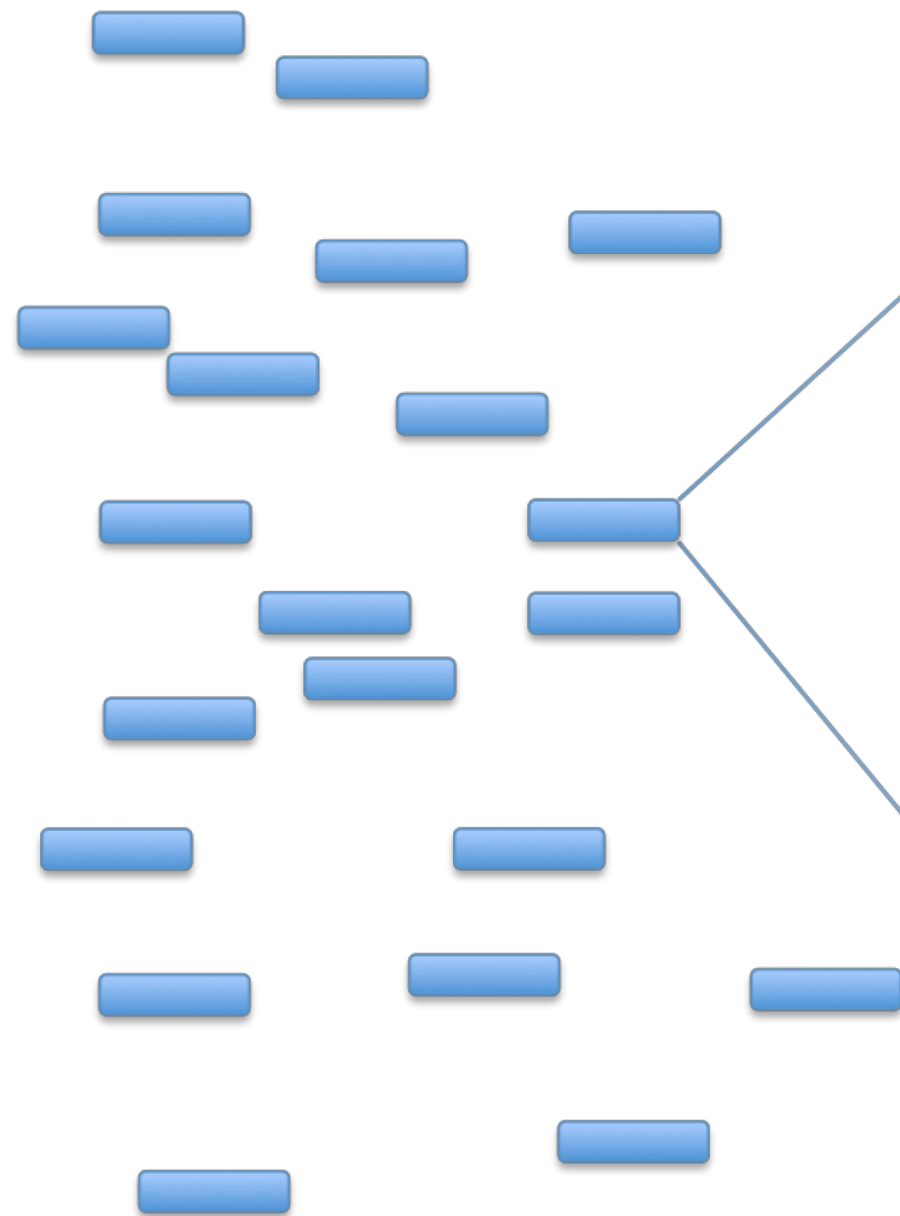
Solutions meilleures  
que par approche classique

# Le voyageur de commerce

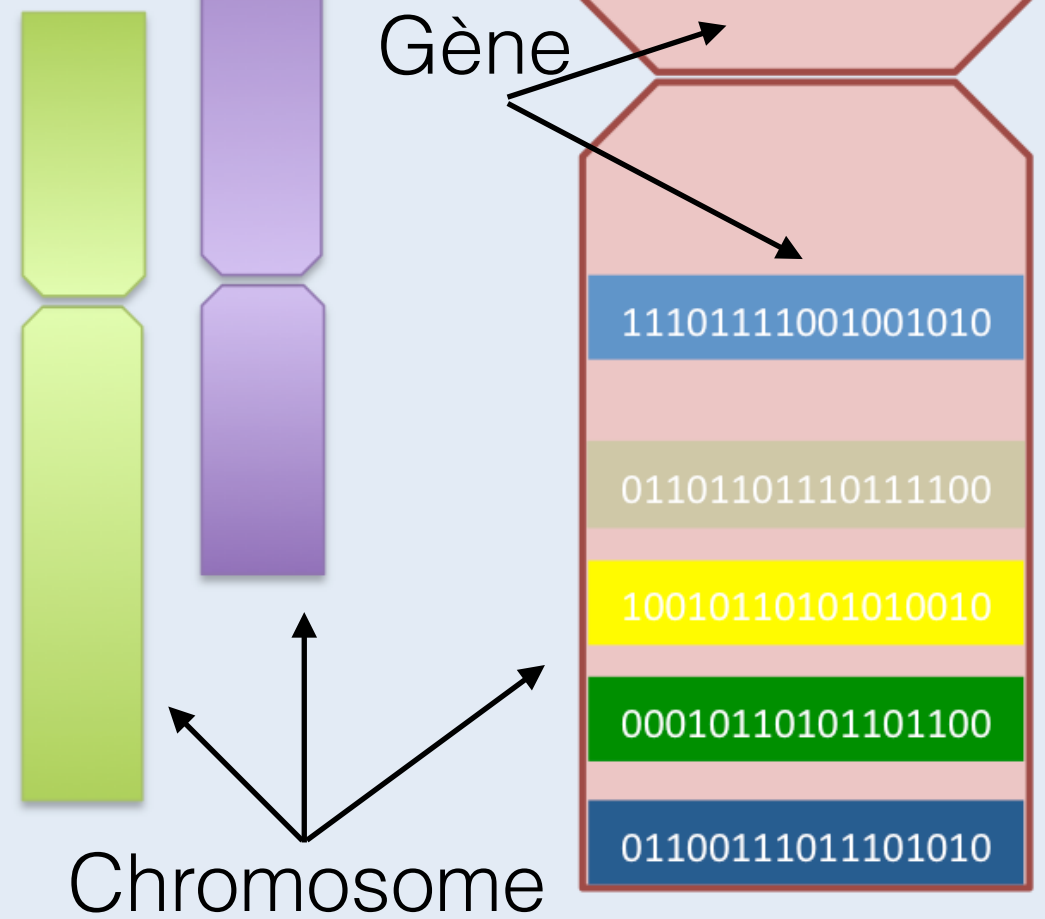
## Wikipédia

*Le problème du voyageur de commerce consiste, étant donné un ensemble de villes séparées par des distances données, à trouver le plus court chemin qui relie toutes les villes. Il s'agit d'un problème d'optimisation pour lequel on ne connaît pas d'algorithme permettant de trouver une solution exacte en un temps polynomial.*

# Population



## Individu



# Encodage des gènes

Réels, entiers, binaire, Gra

Binaire



BIT LE PLUS SIGNIFIANT

$$g_i = \sum_{j=0}^{31} b_j \cdot 2^j \quad 0 \leq g_i \leq g_{max}, \forall i \in [1, n]$$

encodage / décodage

$$g_i = \frac{x_i - x_{imin}}{x_{imax} - x_{imin}} \cdot g_{imax}$$

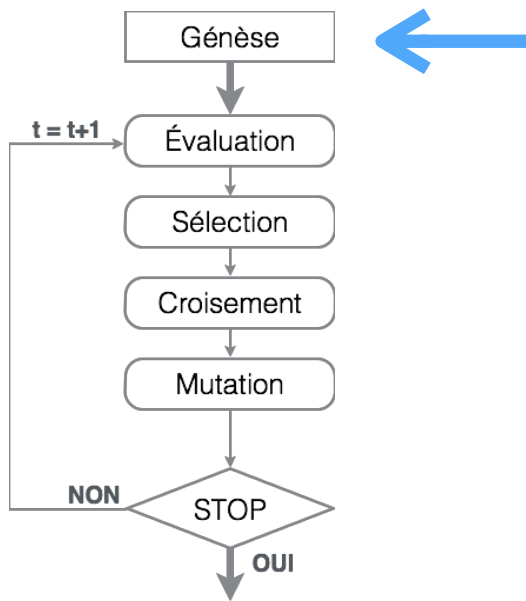
$$x_i = x_{imin} + (x_{imax} - x_{imin}) \cdot \frac{g_i}{g_{max}}$$

Application au voyageur de commerce

Un **gène** représente une ville et est codé par un entier.

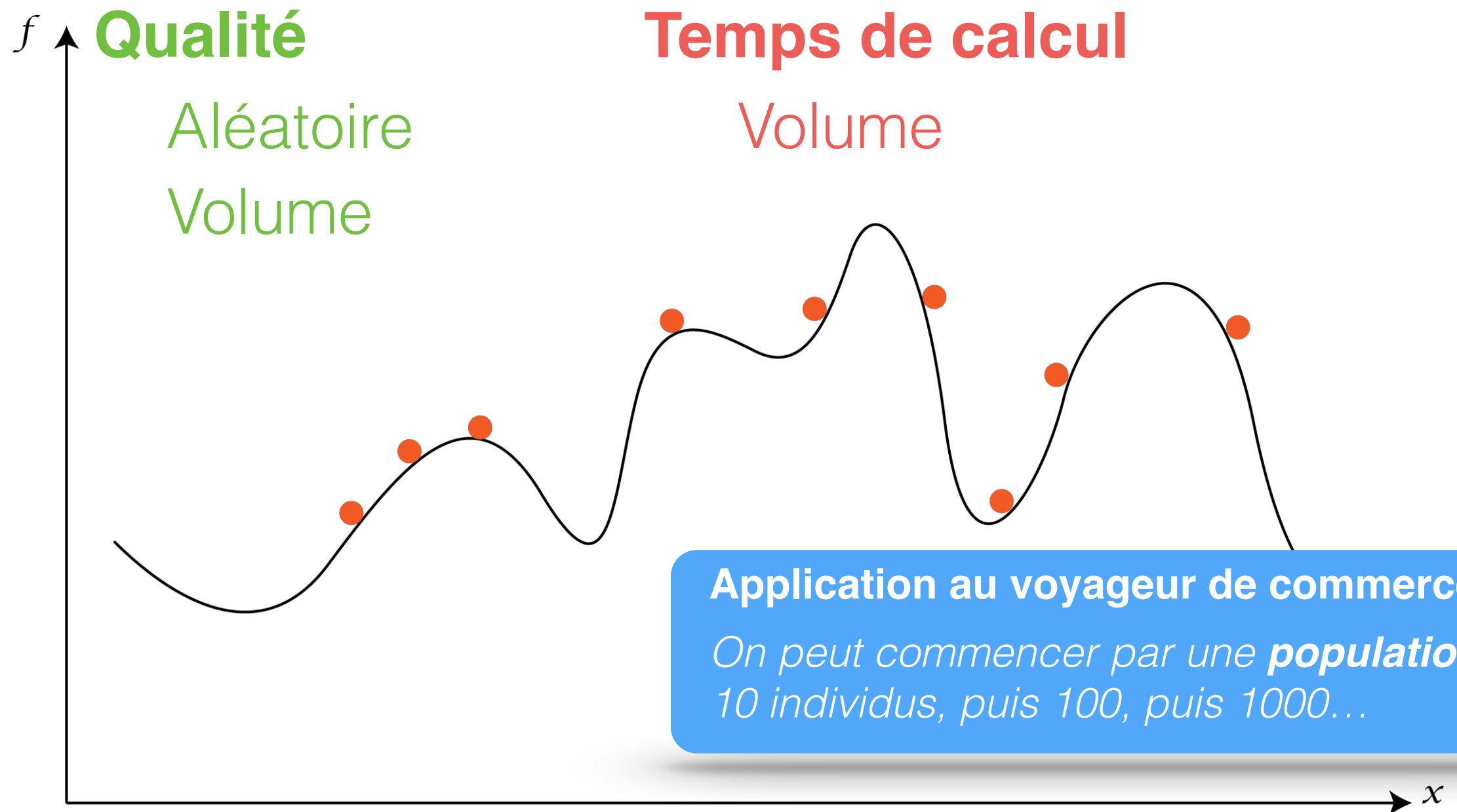
Application au voyageur de commerce

Un **chromosome** représente un chemin parmi les villes du problème.

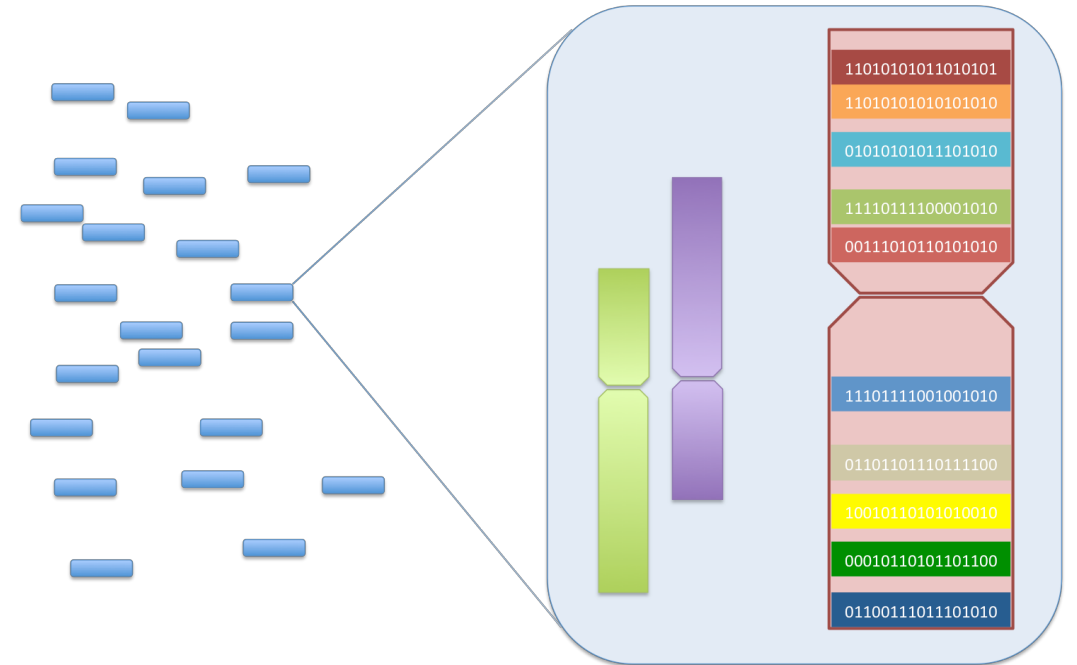
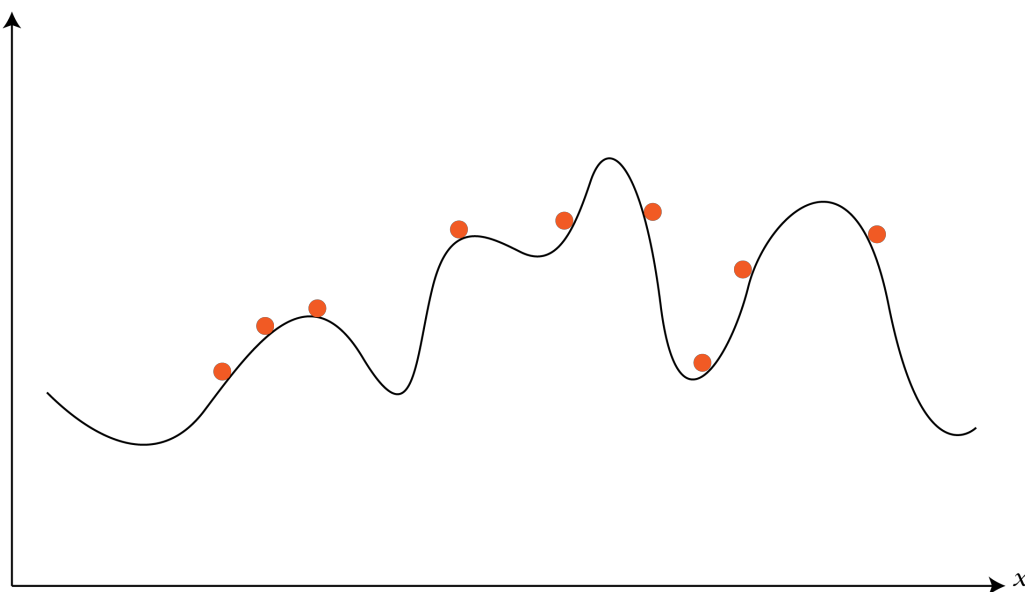
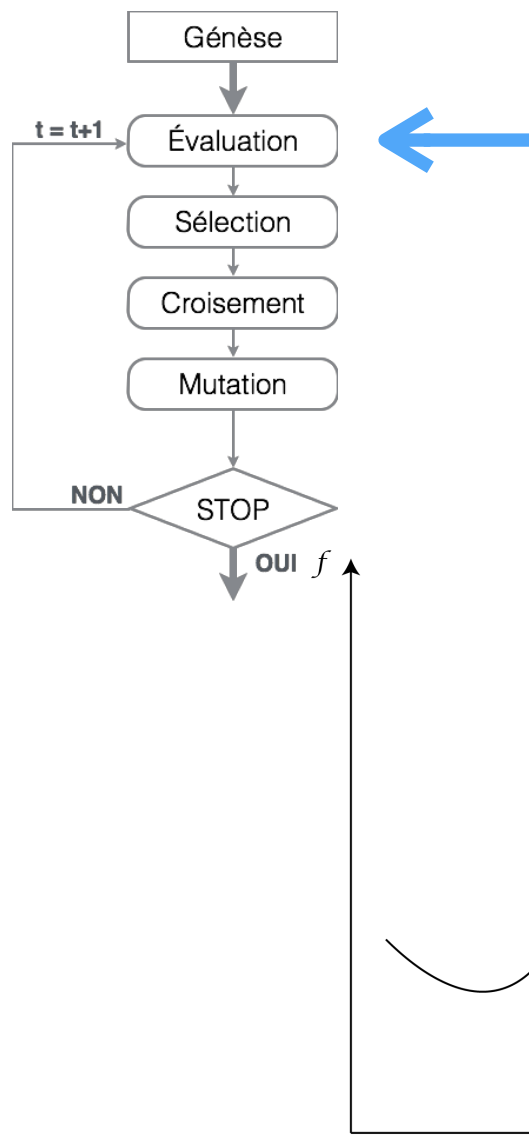


# ÉTAPE 1

## GÉNÉRATION INITIALE



# ÉTAPE 2 ÉVALUATION



## Application au voyageur de commerce

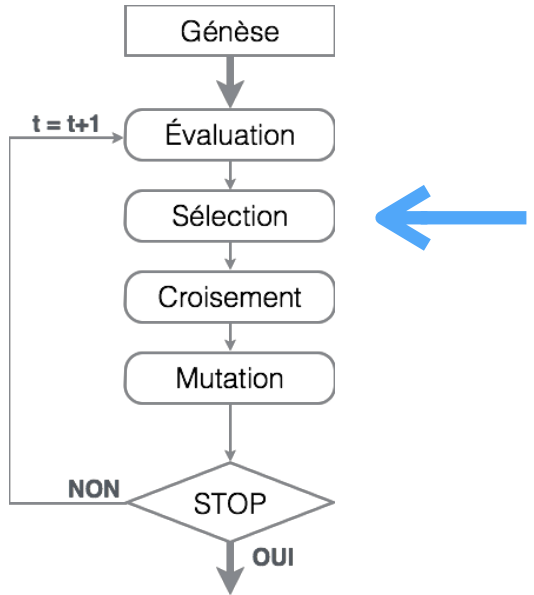
La fonction de **fitness** calculera la distance parcourue si le voyageur emprunte le chemin représenté par l'individu évalué.

INDIVIDU



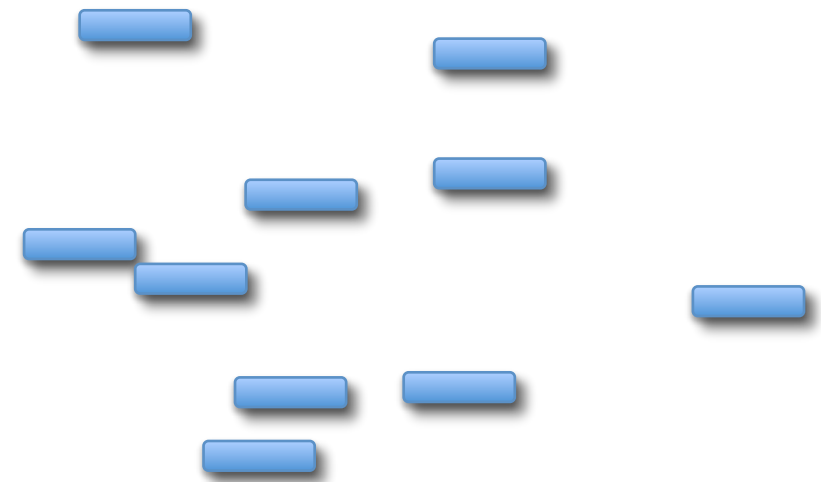
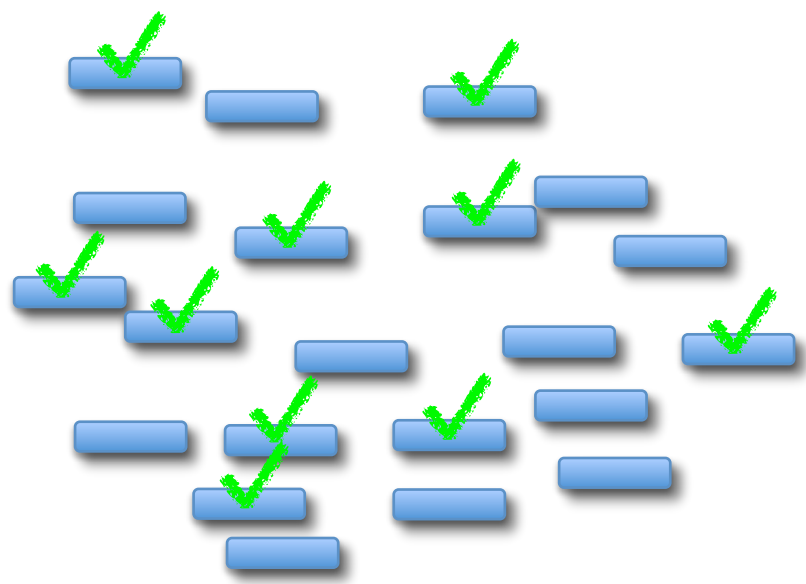
SCORE





# ÉTAPE 3

## SÉLECTION



@

INDIVIDU



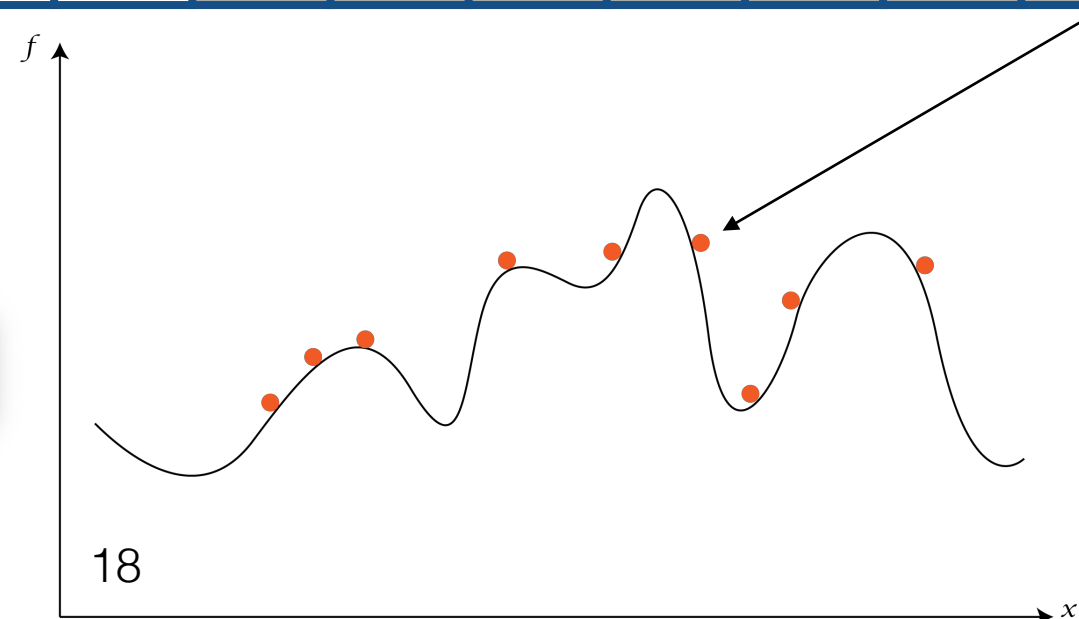
SCORE

Sélectionnés

Rejetés

469	354	352	287	244	215	201	176	154	133	87	65	34	32	14
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	----	----	----	----	----

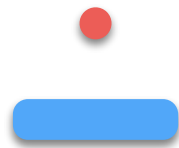
**Forte pression de sélection**



# ÉTAPE 3

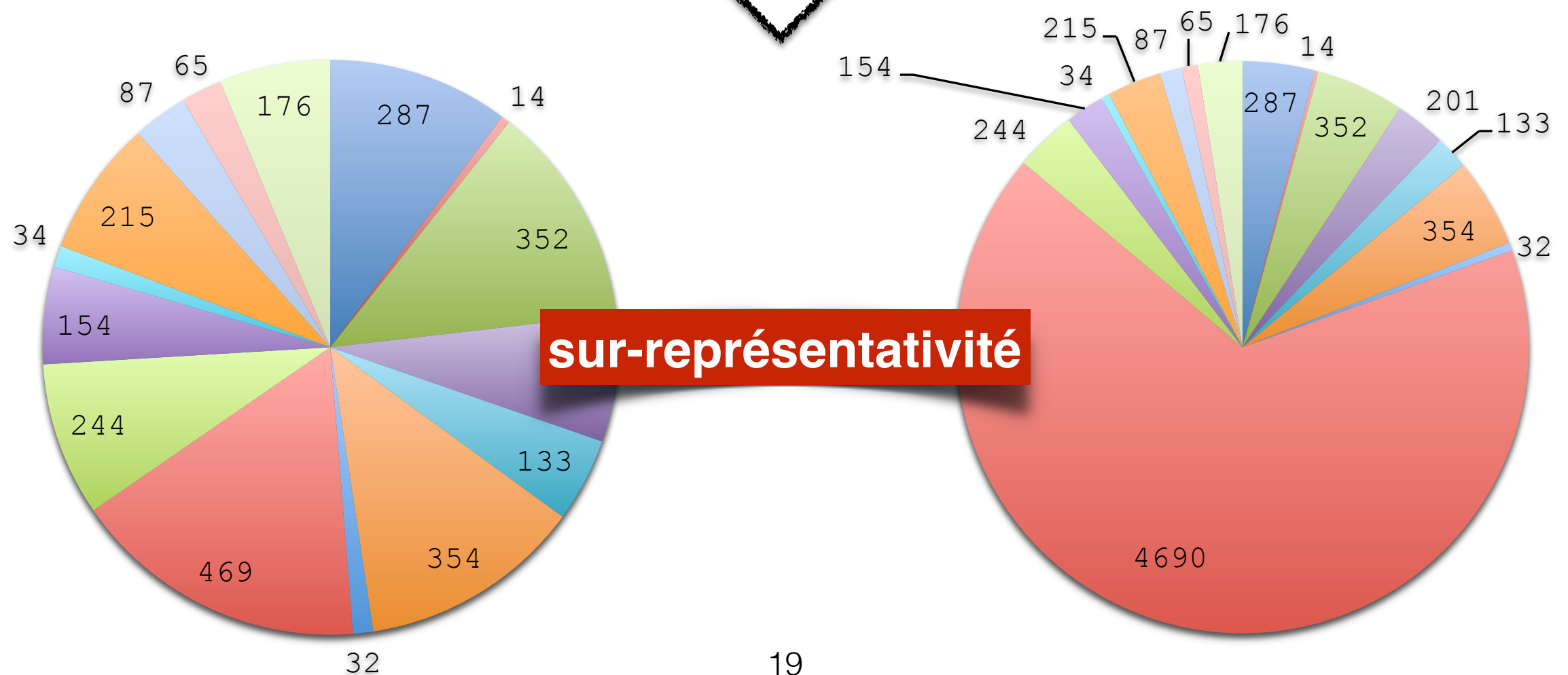
## SÉLECTION PAR ROULETTE

INDIVIDU



*fitness*

SCORE



# ÉTAPE 3

## SÉLECTION PAR RANG

INDIVIDU

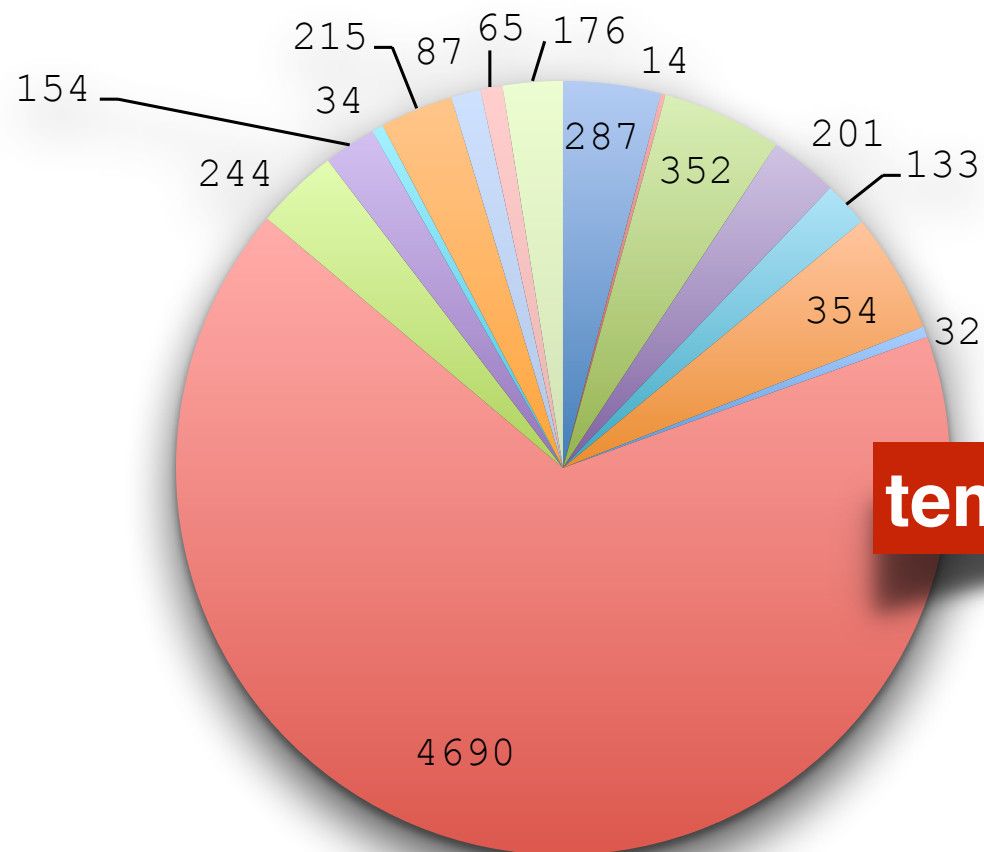


*fitness*

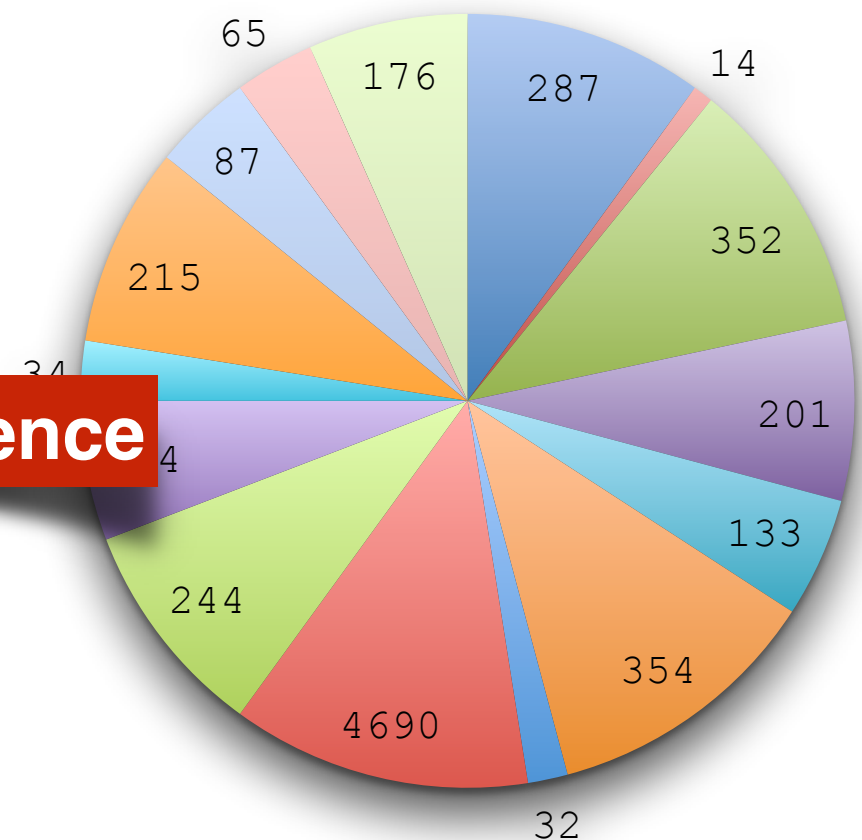
SCORE

6 10 15 3 7 12 1 8 13 9 5 14 11 2 4

133	215	4690	34	154	287	14	176	352	201	87	354	244	32	65
-----	-----	------	----	-----	-----	----	-----	-----	-----	----	-----	-----	----	----



**temps de convergence**



# ÉTAPE 3

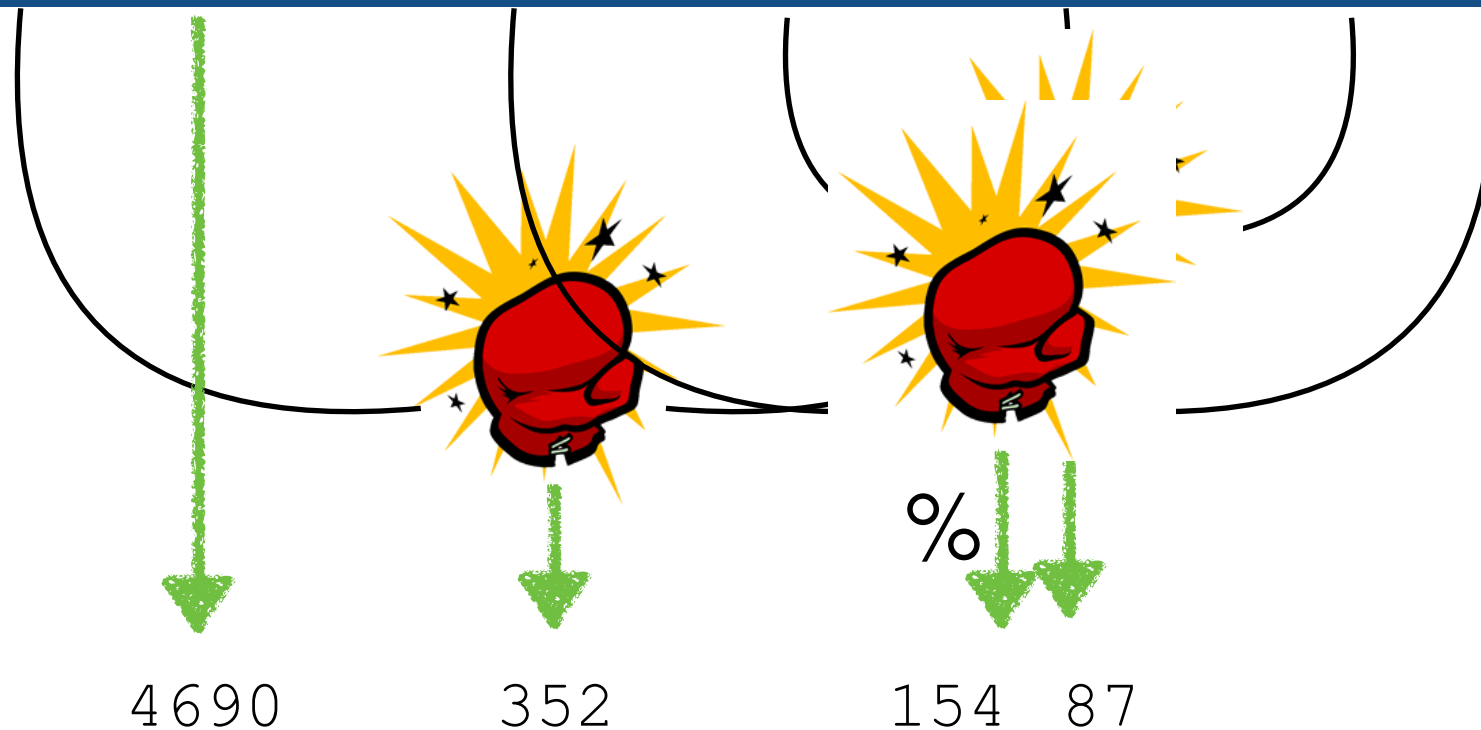
## SÉLECTION PAR TOURNOI

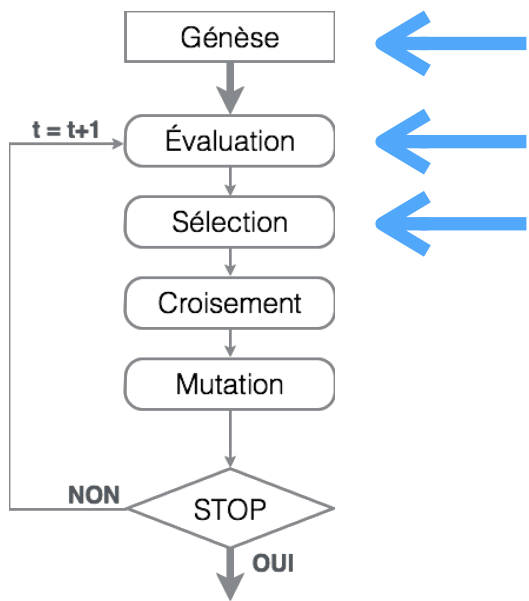
INDIVIDU



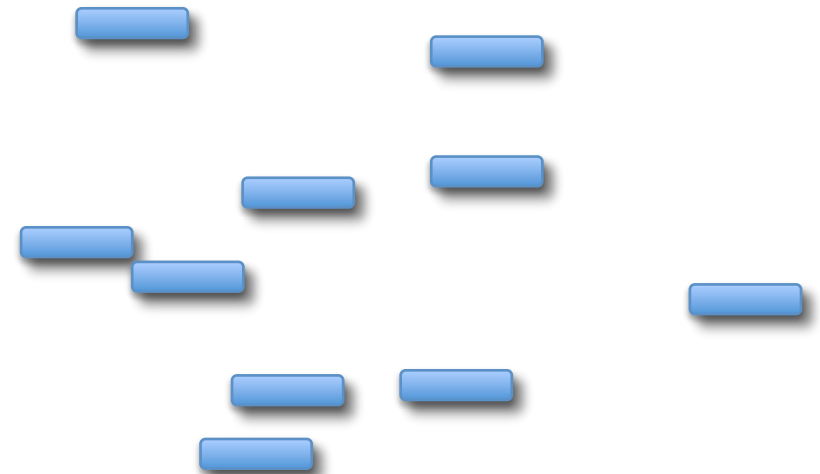
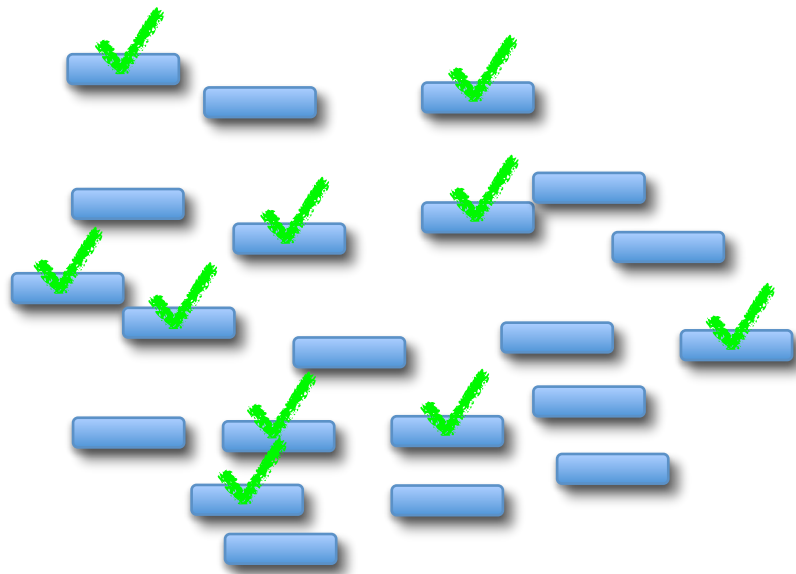
SCORE

133	215	4690	34	154	287	14	176	352	201	87	354	244	32	65
-----	-----	------	----	-----	-----	----	-----	-----	-----	----	-----	-----	----	----





# ÉTAPE 3 SÉLECTION

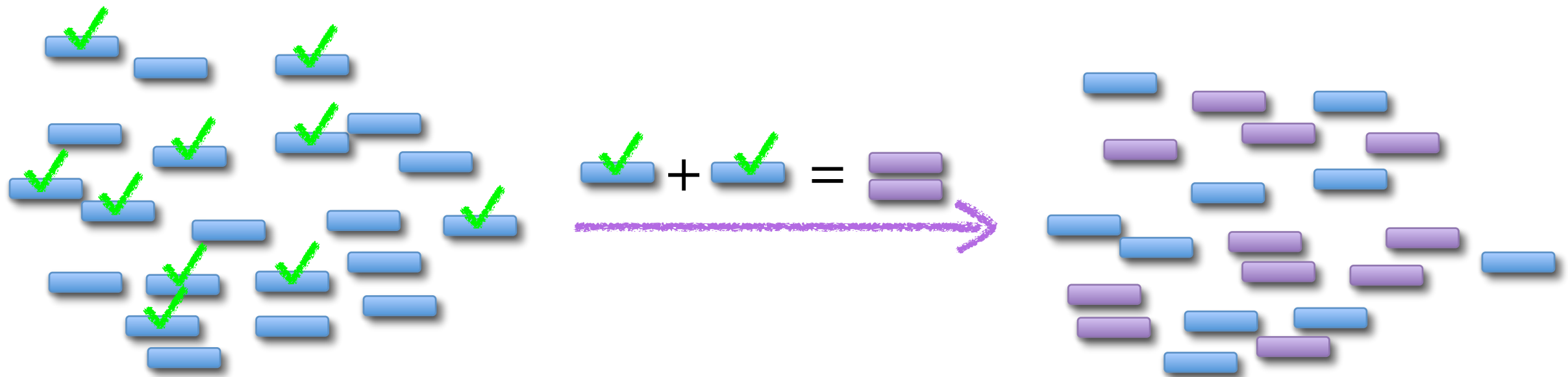
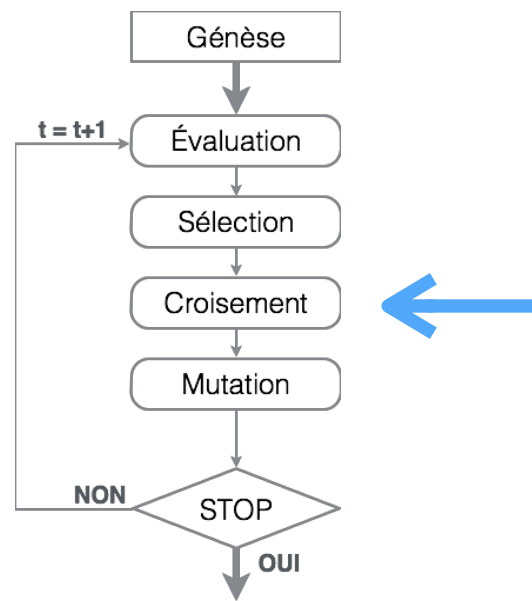


**NOUVELLE GÉNÉRATION**

## Application au voyageur de commerce

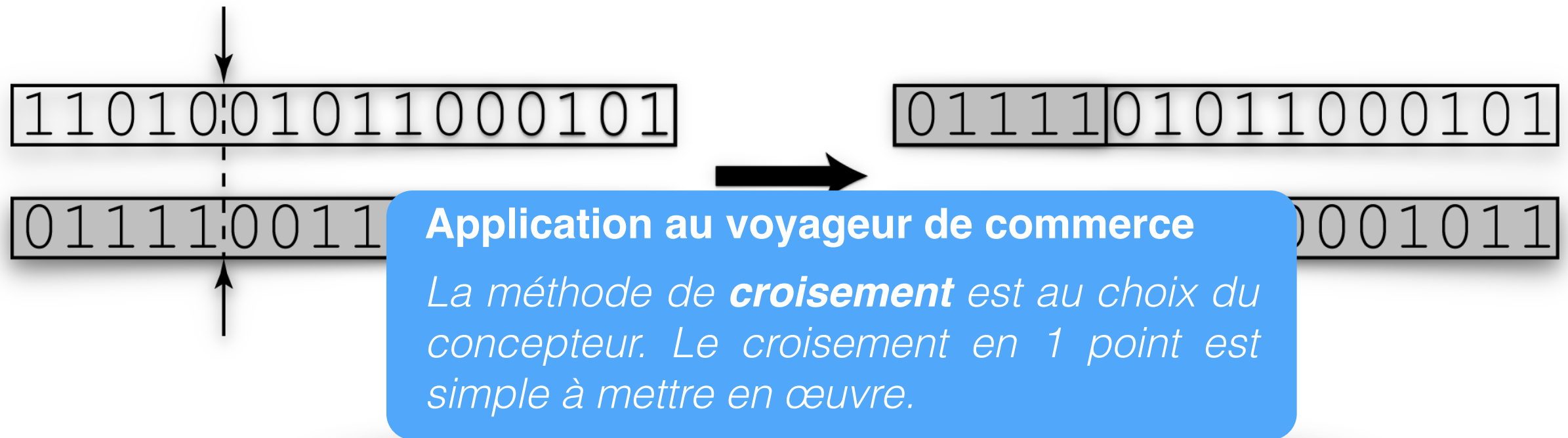
*La méthode de **sélection** est au choix du concepteur. L'élitisme est simple à mettre en œuvre.*

# ÉTAPE 4 CROISEMENT

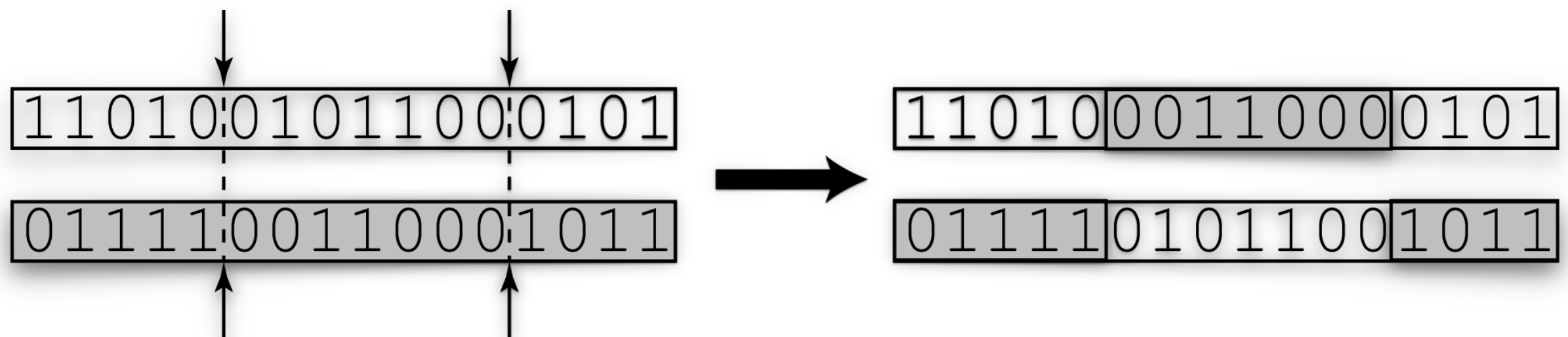


# ÉTAPE 4

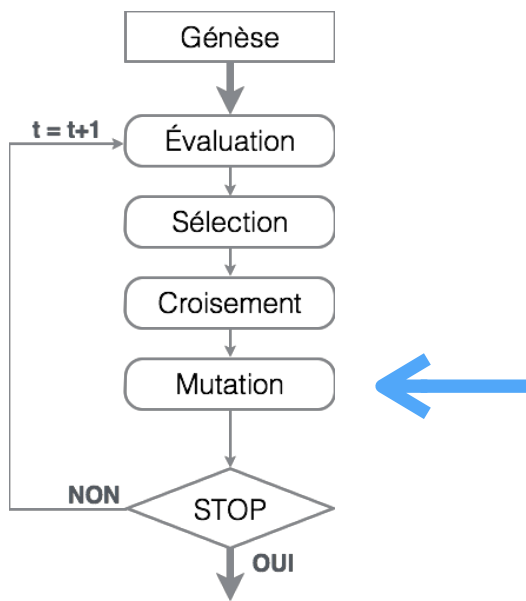
## CROISEMENT EN 1 POINT



## CROISEMENT EN 2 POINTS







# ÉTAPE 5

## MUTATION

**Stagnation de l'évolution**

**Dérive génétique**

1 1 0 1 0 0 1 0 1 1 0 0 0 1 0 1



1 1 0 1 0 0 1 0 1 1 0 1 0 1 0 1

$$\Rightarrow 0,001 < p_m < 0,01$$

$$\Rightarrow p_m = 1/\mathcal{L} \text{ où } \mathcal{L} = \text{nb de bits du chromosome}$$

$\Rightarrow p_m$  variable d'un gène à l'autre

$\Rightarrow p_m$  variable au cours du temps

# ÉTAPE 5

## MUTATION AUTO-ADAPTATIVE

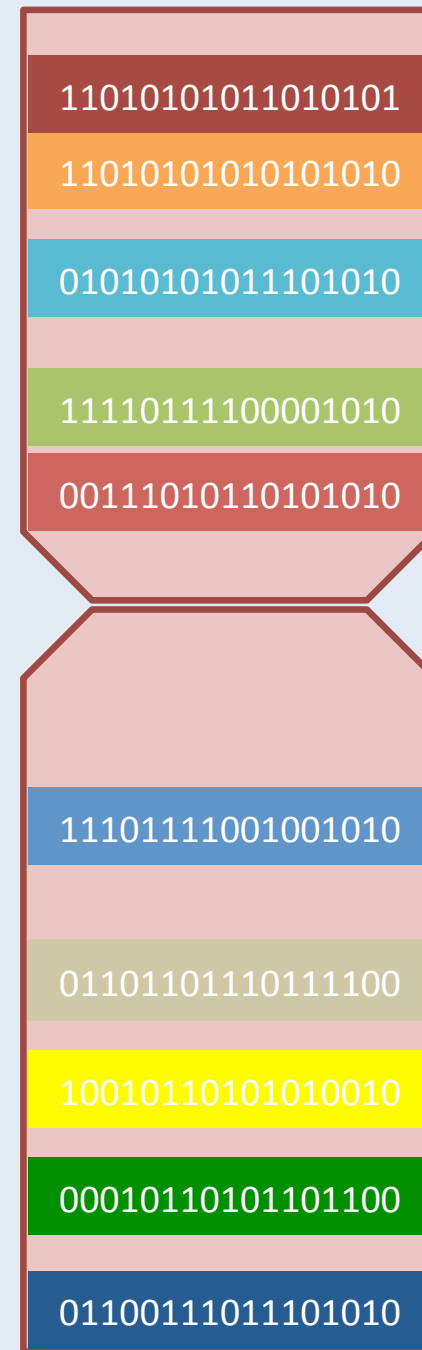
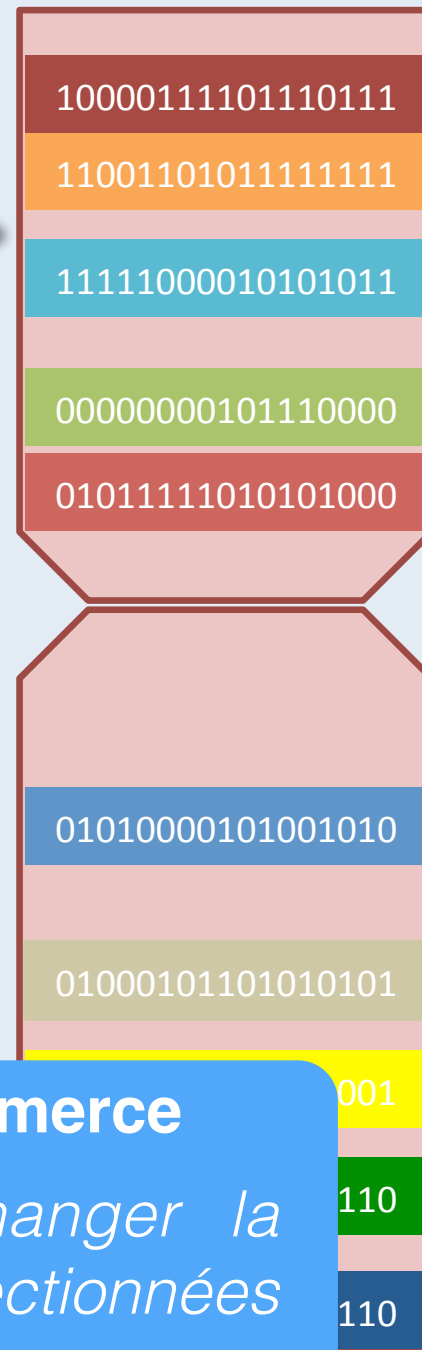
Chromosome  
codant les  
probabilités de mutation



ÉVOLUTION



Individu



Chromosome  
codant les  
données



ÉVOLUTION

Application au voyageur de commerce

La **mutation** consistera à échanger la position de couples de villes sélectionnées au hasard. On commencera par 2.

# Algorithmique génétique

Évolution

**Pratique et tâtonnement**

Valeurs-clés

**Non-déterministe**

**Proche de l'optimal**