Heuristiques

Optimisation évolutionnaire

Evelyne LUTTON

Département MIA
UMR MIA-Paris, INRAE-AgroParisTech
Équipe EKINOCS
ISC-PIF, Paris 13ème
Evelyne.Lutton@inrae.fr

INRAE

http://evelyne-lutton.fr/CoursENSTA.html

Optimiser ?

Numérique

Calcul analytique :

Max f(x), f'(x) = 0

Algorithme :

grande dimension,
fonctions irrégulières

Combinatoire

Recherche exhaustive :

Voyageur de commerce
(10 villes)

Algorithme :

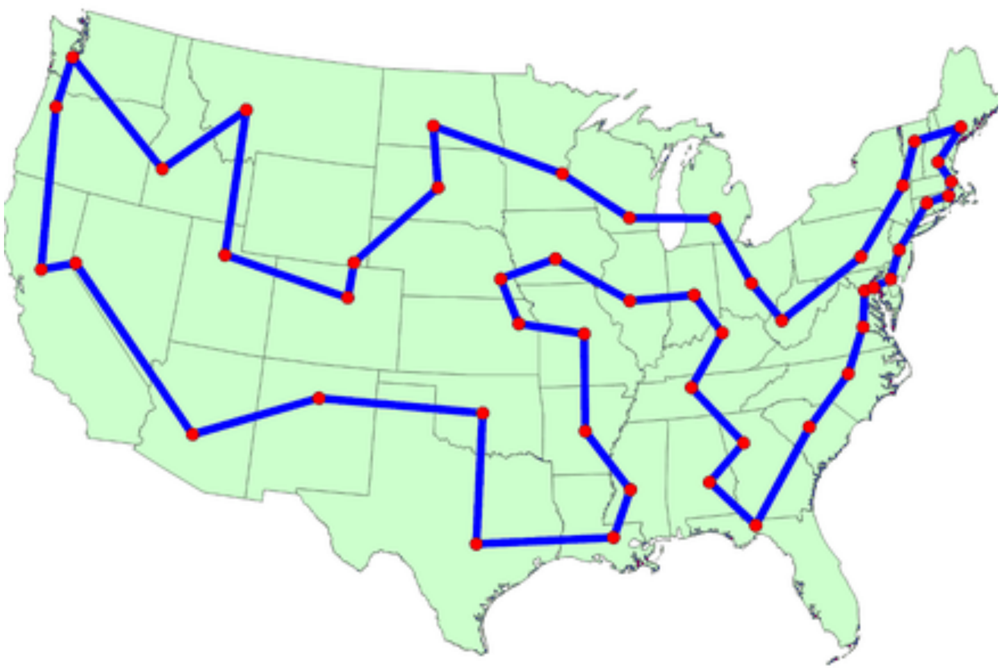
Voyageur de commerce
(100 000 villes)

Heuristiques

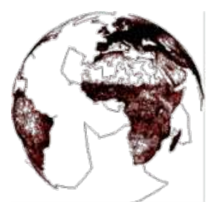
Le voyageur de commerce

William Rowan Hamilton (1859) :

« Un voyageur de commerce doit visiter une et une seule fois un nombre fini de villes et revenir à son point d'origine. Trouvez l'ordre de visite des villes qui minimise la distance totale parcourue par le voyageur »



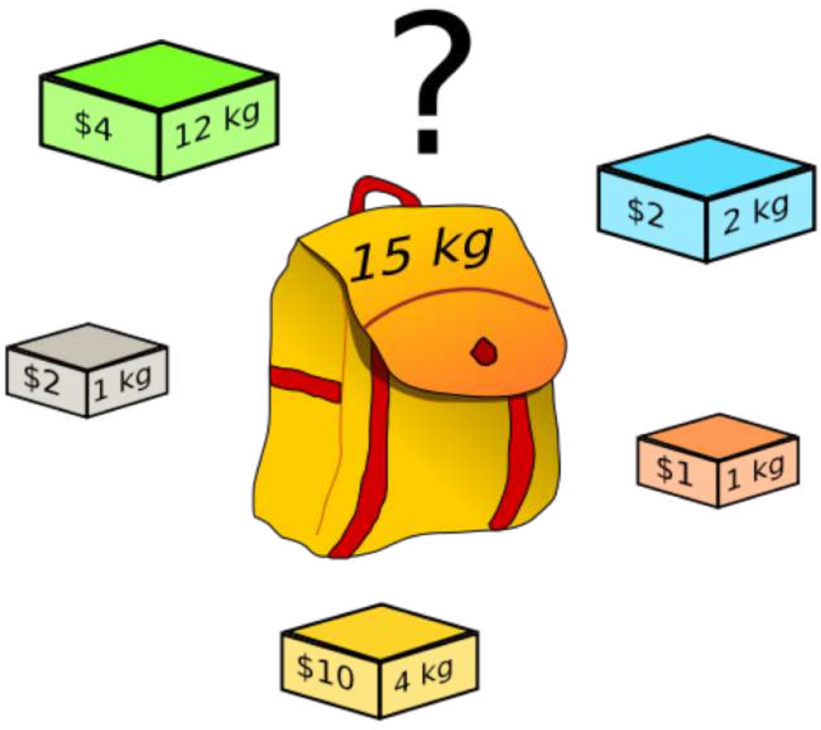
Problème NP-Complet



- 1,9x10^{6} villes en 2018
  (World TSP Tour, meilleur résultat dans les **0.05%** de la longueur théorique optimale).

- 2500 villes :
  résolution **exacte** en quelques secondes en 2020 (outils commerciaux).

Le problème du sac à dos

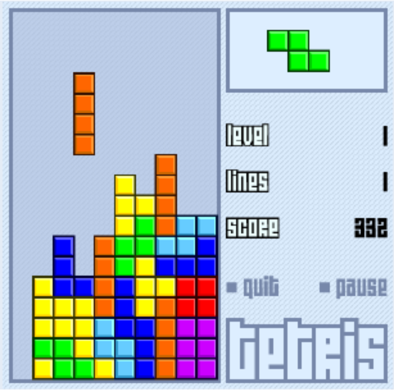


Quelles boîtes choisir afin de maximiser la somme emportée tout en ne dépassant pas les 15 kg autorisés ?

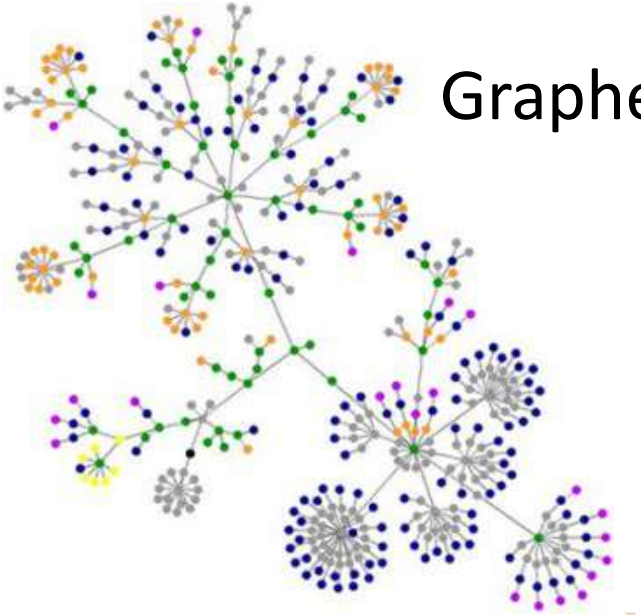
Contraintes supplémentaires :
volume limité,
catégories données d'objets,
etc …

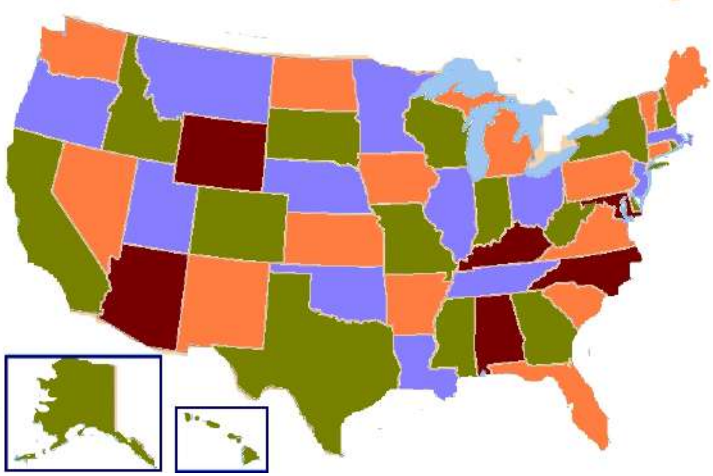
Autres problèmes combinatoires

Jeux



Graphes





Optimiser en conditions « hostiles » :

- Algorithmique heuristique :

  les vertus de l'approximation.

- Algorithmique évolutionnaire :

  le pouvoir du troupeau.

Méta heuristiques

méta : au-delà, à un plus haut niveau,

heuriskein : trouver

Wikipedia : «Une métaheuristique est un algorithme d'optimisation visant à résoudre des problèmes d'optimisation difficiles (souvent issus des domaines de la recherche opérationnelle, de l'ingénierie ou de l'intelligence artificielle) pour lesquels on ne connaît pas de méthode classique plus efficace»

Je suis le meilleur.

Souvent :

- fondées sur un **échantillonnage probabiliste**,
- inspirées par des **systèmes naturels**.

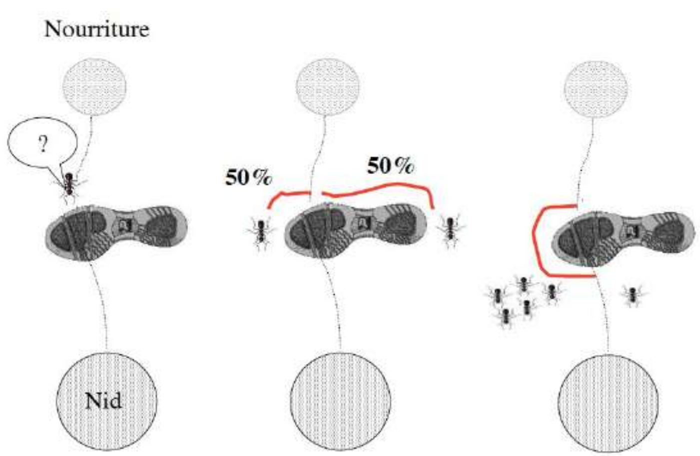
La grande famille des méthodes approchées

- **Méthodes par construction : « glouton. »**

- **Méthodes par décomposition**

- **Méthodes de voisinage (une solution courante) :**
  - heuristiques classiques
  - métaheuristiques de voisinage : recuit simulé, recherche tabou.

- **Méthodes à base de population (plusieurs solutions en //) :**
  - **algorithmes évolutionnaires**,
  - estimations de distributions,
  - colonies de fourmis,
  - essaims particulaires.

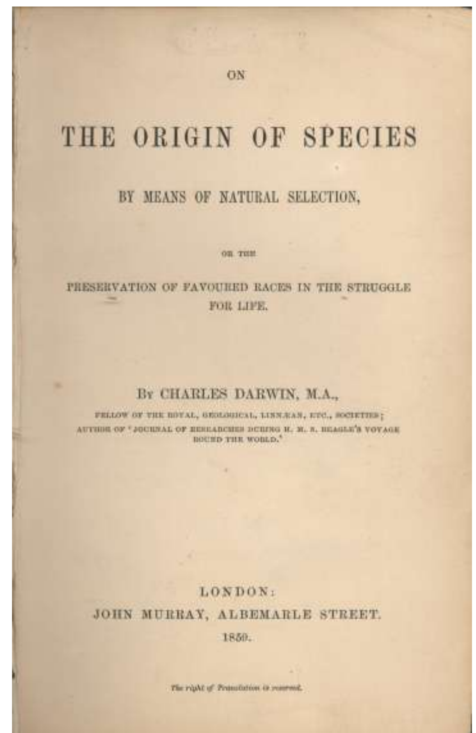


Algorithmes évolutionnaires

Darwinisme, évolutionnisme

1831 - Voyage de 5 ans sur le HMS Beagle
(îles Galapagos)

Ouvrage ``Sur l'origine des espèces''
Novembre 1859.



ON

THE ORIGIN OF SPECIES

BY MEANS OF NATURAL SELECTION,

OR THE

PRESERVATION OF FAVOURED RACES IN THE STRUGGLE FOR LIFE.

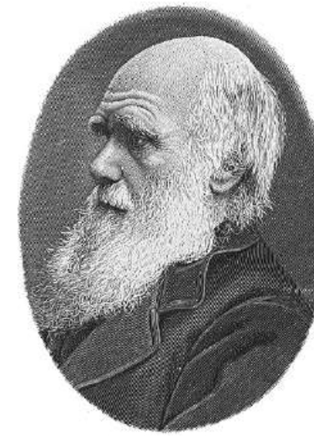
BY CHARLES DARWIN, M.A.,

FELLOW OF THE ROYAL, GEOLOGICAL, LINNÆAN, ETC., SOCIETIES;
AUTHOR OF 'JOURNAL OF RESEARCHES DURING H. M. S. BEAGLE'S VOYAGE ROUND THE WORLD.'

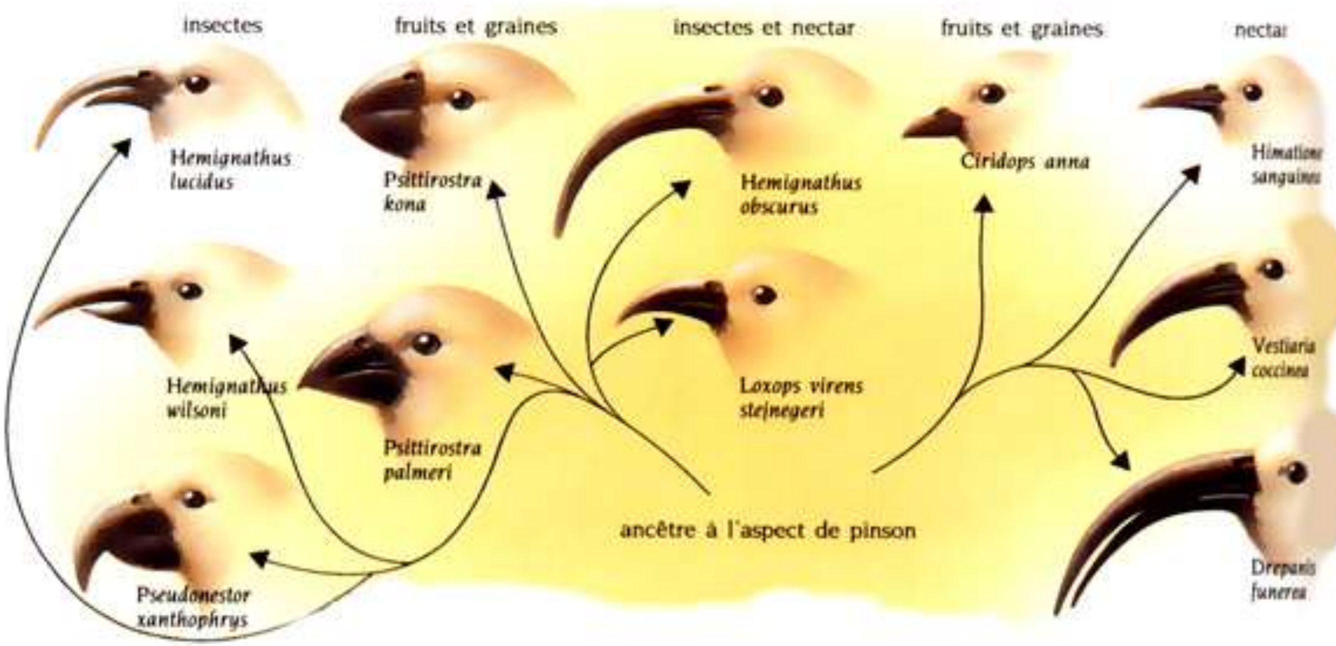
LONDON:
JOHN MURRAY, ALBEMARLE STREET.
1859.

The right of Translation is reserved.

Charles Robert Darwin
(1809-1882).



Le pinson de Darwin



« On peut dire, par métaphore, que la sélection naturelle recherche, à chaque instant et dans le monde entier, les variations les plus légères ; elle repousse celles qui sont nuisibles, elle conserve et accumule celles qui sont utiles ; elle travaille en silence, insensiblement, partout et toujours, dès que l'occasion s'en présente, pour améliorer tous les êtres organisés relativement à leurs conditions d'existence organiques et inorganiques » (darwin, 1859).

Le Darwinisme artificiel

Utiliser des principes d'évolution organique en tant que technique d'optimisation globale.

Imiter les phénomènes d'apprentissage collectif (adaptation) des populations naturelles.

Deux grands courants initiaux :

algorithmes génétiques et systèmes de classeurs :

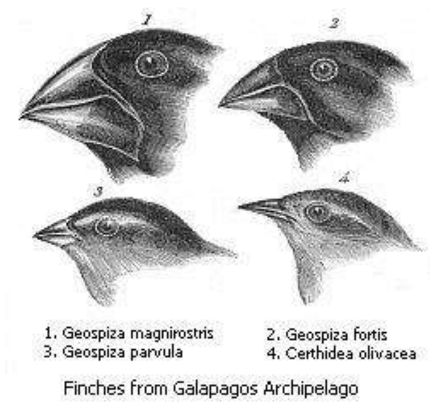
John Holland, David Goldberg.

stratégies d'évolution et evolutionary computation :

Ingo Rechenberg, Hans-Paul Schwefel, Lawrence Fogel.

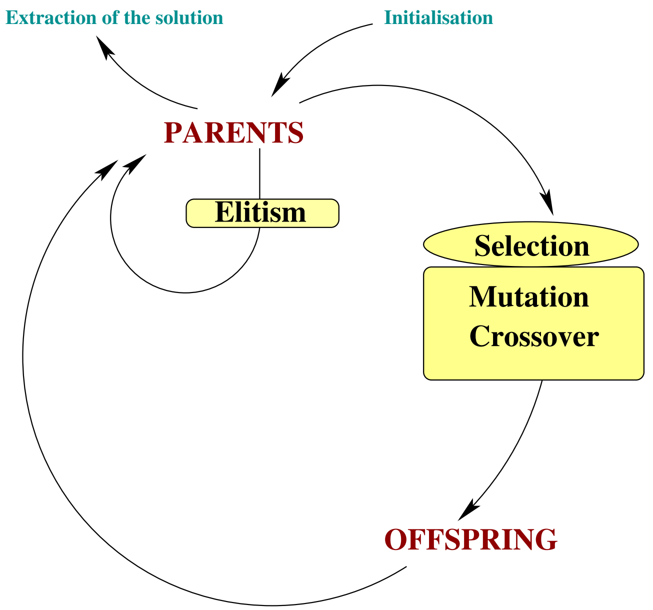
Evolution artificielle

Darwinisme, Evolutionnisme



- **variations,** macroscopiques et microscopiques, au sein des espèces,
- **hérédité,**
- **sélection naturelle** : triomphe de la lignée qui possède une variation utile dans son environnement

Algorithmique Evolutionnaire



Technique d'optimisation globale :
Imitation des phénomènes d'apprentissage collectifs (adaptation) des populations naturelles

Ingrédients

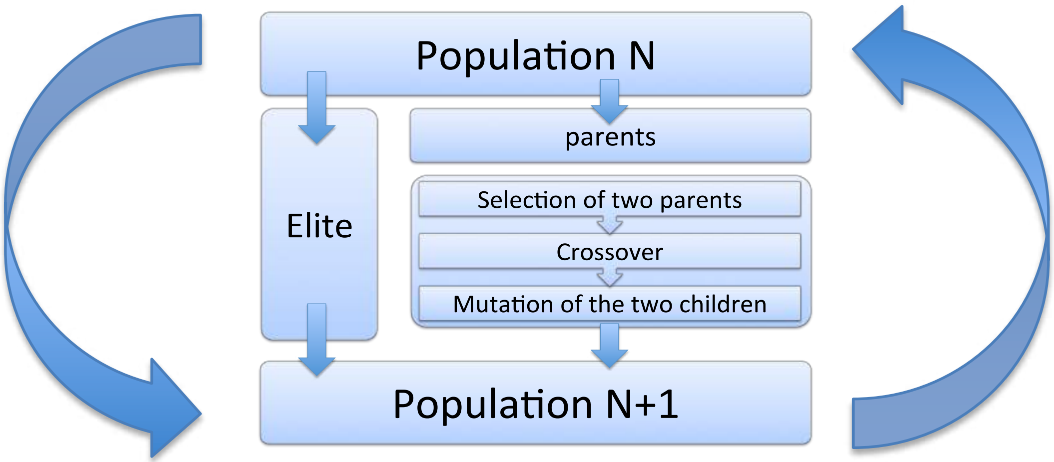
Population

Sélection

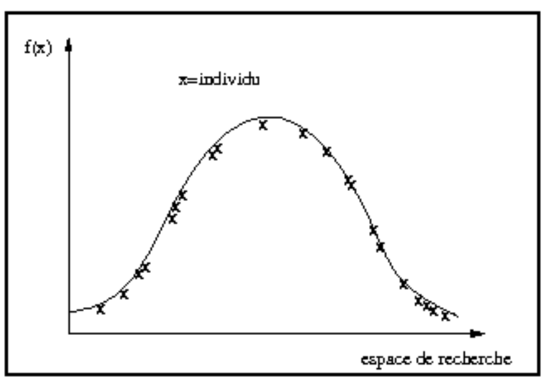
Opérateurs génétiques

Évolution simulée

Optimisation évolutionnaire

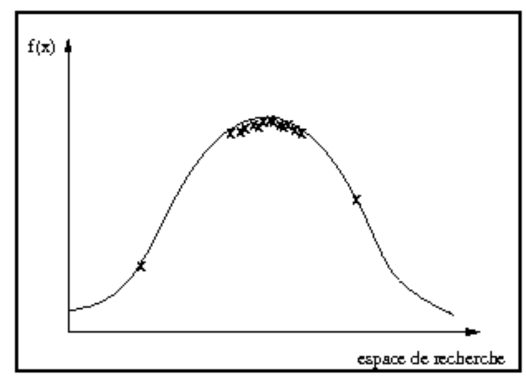


Population initiale



Un individu = une solution

Population évoluée



Algorithmes évolutionnaires

Un ensemble de techniques regroupées sous un terme générique

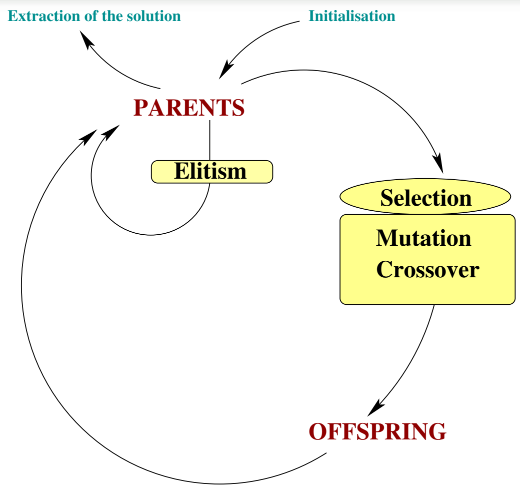
AE

- Algorithmes Génétiques (**AG**)
- Stratégies d'Évolution (**SE**)
- Programmation Évolutionnaire (**PE**)
- Programmation Génétique (**PG**)
- Systèmes de Classeurs (**CS**)

Un peu de vocabulaire

Algorithme Évolutionnaire	Méthode d'optimisation
Individu	Solution (vecteur)
Population	Ensemble de solutions
Chromosome	Codage de la solution (Exemple du codage binaire)
Croisement ou recombinaison	Opération sur deux codes
Mutation	Opération sur un code
Environnement	Espace de recherche
Fitness Degré d'adaptation à l'environnement	Valeur de la fonction à optimiser
Évolution	Maximisation de la fonction

Le moteur évolutionnaire



Évaluation : estimer la qualité d'un individu.

➡ *fitness, performance, fonction d'évaluation, adaptation à l'environnement.*

Sélection : trier les meilleurs individus.

➡ *tirages aléatoires biaisés (roulette wheel), sélection sur le rang, par tournoi.*

Reproduction : appliquer les opérateurs génétiques, croisements et mutations, avec des probabilités $\mathbf{p_c}$ et $\mathbf{p_m}$.

➡ *Lié à la représentation de l'espace de recherche :*

Solutions <—> chromosomes
Phénotypes <—> génotypes

Remplacement : fabriquer la génération suivante.

➡ élitisme, pourcentage de renouvellement de la population, *stratégies* $(\mu + \lambda)$ *ou* (μ, λ).

Initialisation / arrêt du processus

Initialisation

- échantillonnage l'espace de recherche (aléatoire, régulier),
- introduction de solutions initiales,
- restrictions de la recherche.

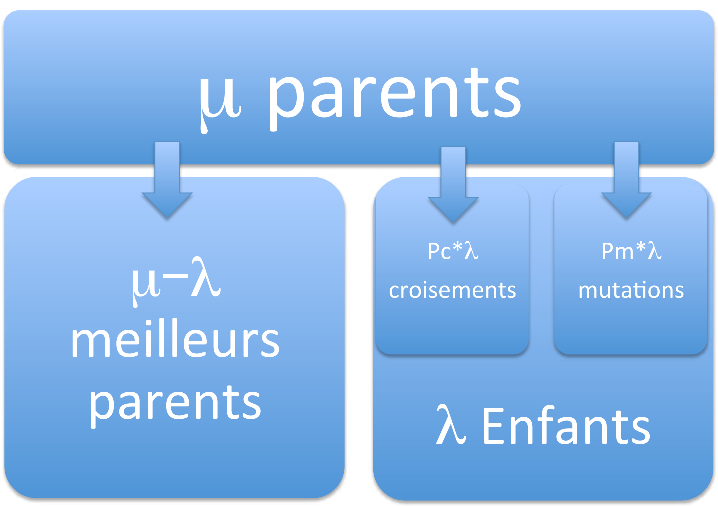
Arrêt, extraction des solutions

- nombre de générations limite, critères de stagnation.
- le meilleur individu de la dernière génération !

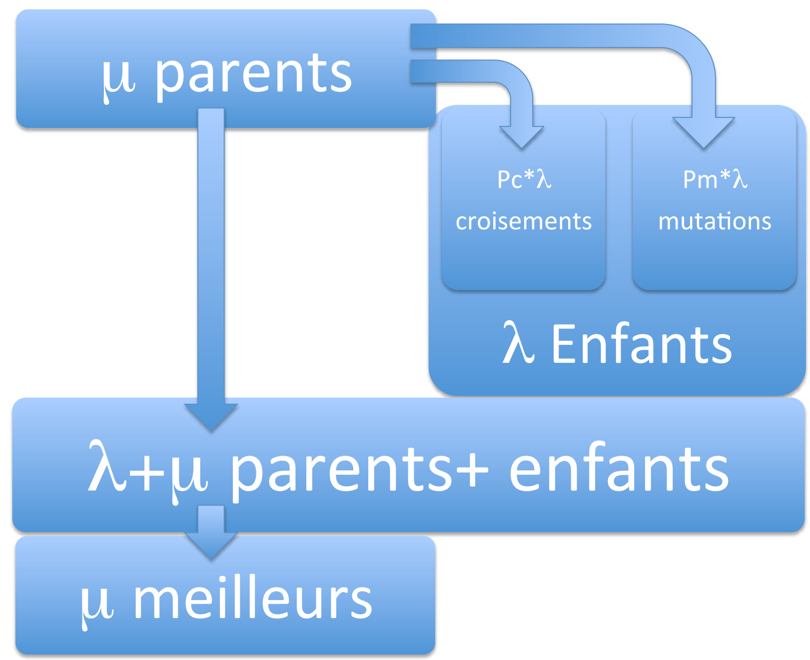
Renouvellement de la population

Conserver la mémoire des bonnes solutions et renouveler la population.

Taux de renouvellement fixe *(μ, λ)*



Taux de renouvellement variable *(μ + λ)*

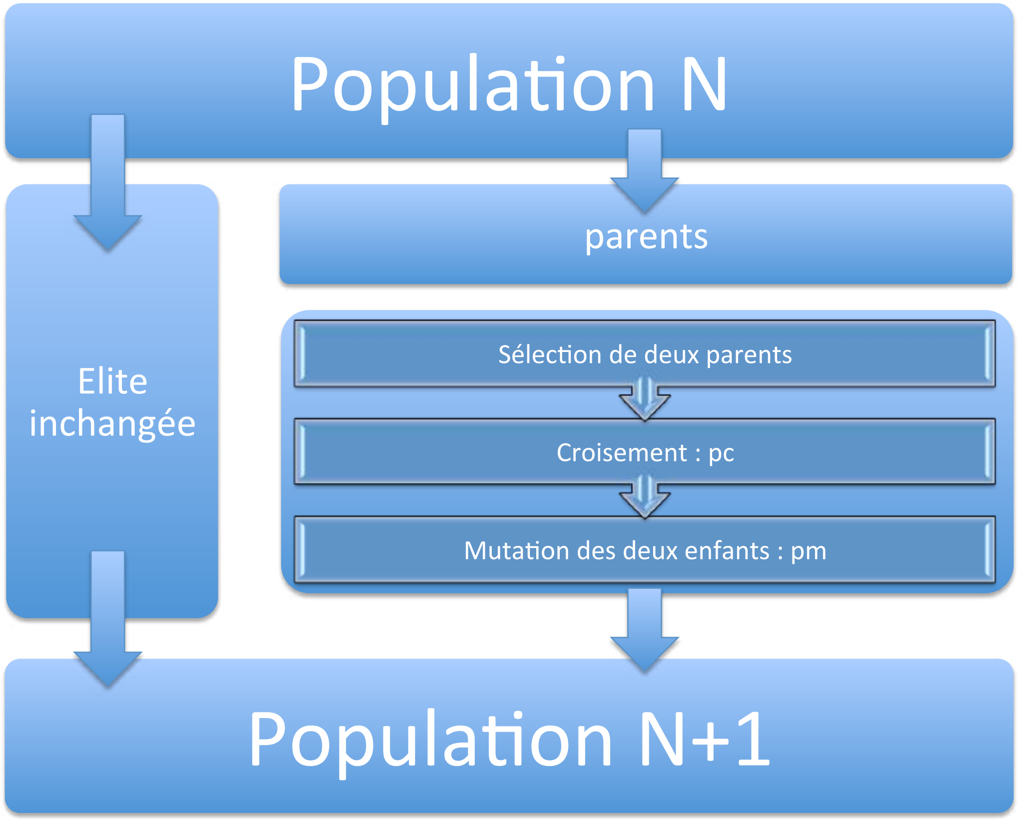


Enchaînement des opérations :
en parallèle ou en cascade

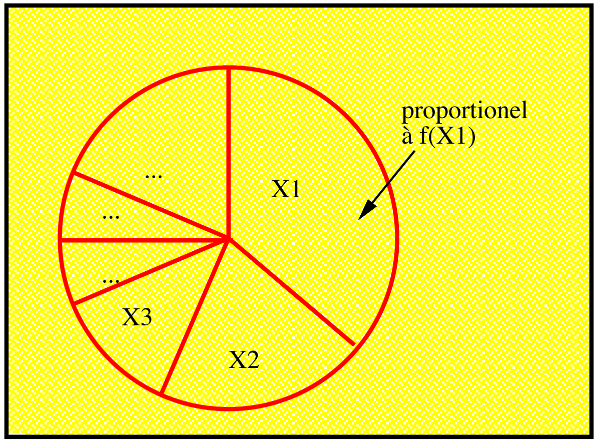
Les ES- ***(μ + λ)*** *et* ***(μ, λ)*** appliquent les opérateurs **en parallèle** : $p_c + p_m = 1$

Les AG appliquent les opérateurs **en cascade** : $p_c + p_m < 1$

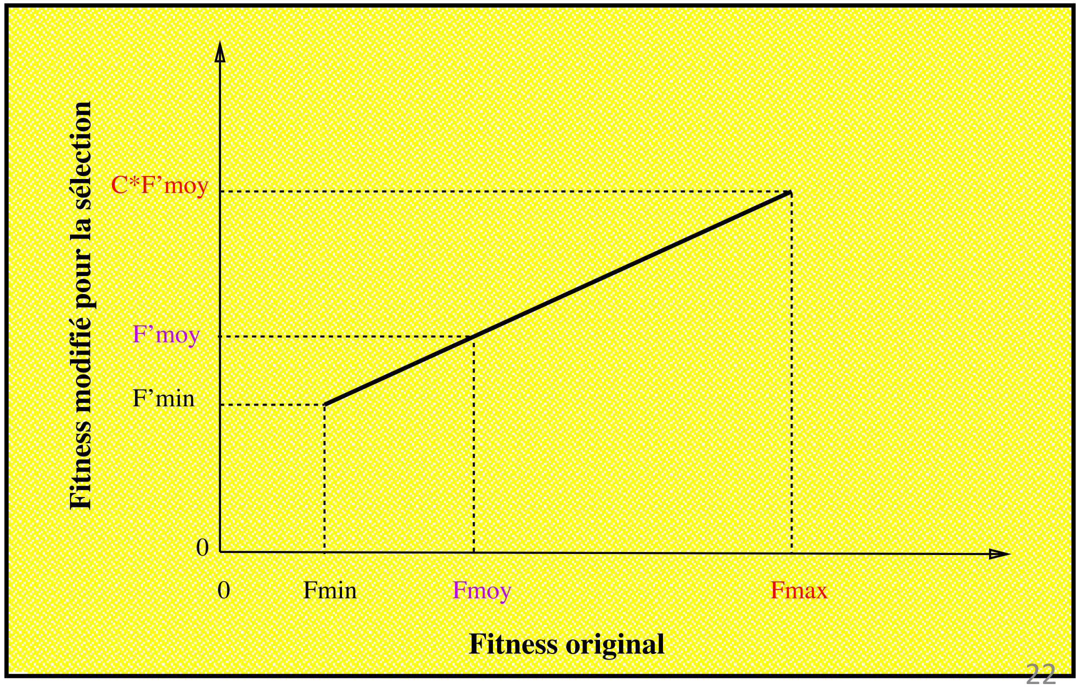
- croisés et mutés : $p_m * p_c$
- croisés et non mutés : $(1-p_m) * p_c$
- mutés et non croisés : $p_m * (1-p_c)$
- inchangés : $(1-p_m) * (1-p_c)$



Pression sélective



$$P(x) = \frac{f(x)}{\sum_{y \in Pop} f(y)}$$



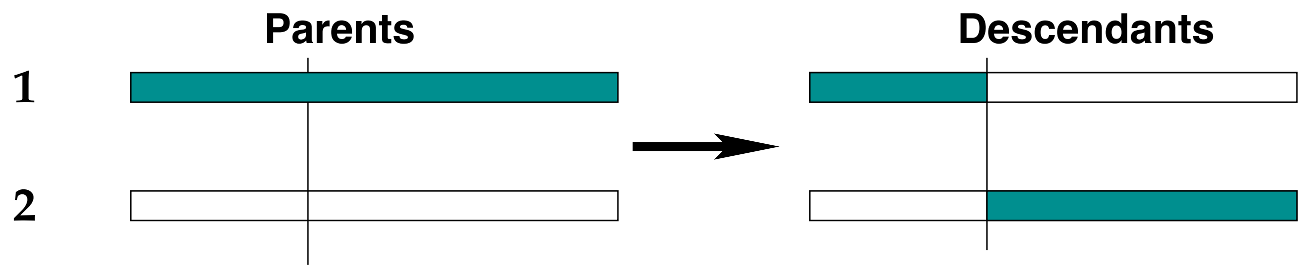
Ni trop, ni trop peu !

- Scaling
- Ranking
- Tournoi

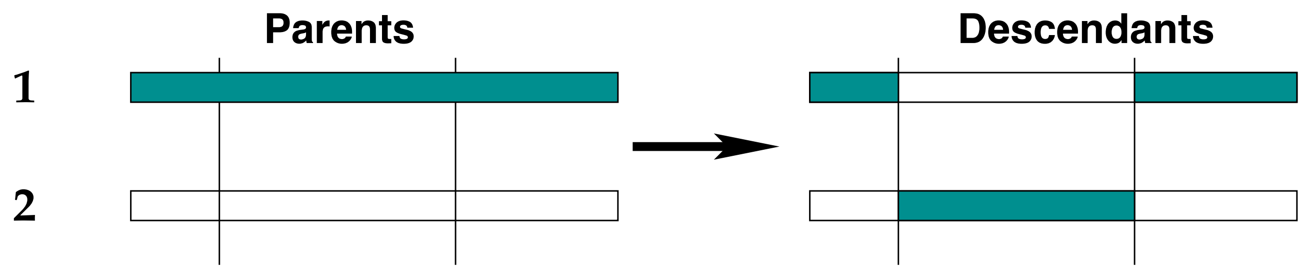
Représentation discrète : algorithmes génétiques

Chaque individu est représenté par une chaîne (binaire) de longueur fixe.

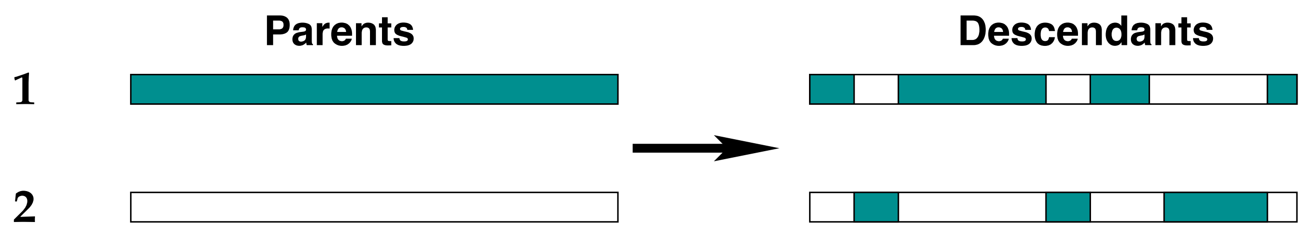
Croisement a 1 point



Croisement à 2 points



Croisement uniforme



Mutation discrète

Une petite perturbation du génôme :

Parent → **Descendant**

01001101001110100111011110 → 01001**0**01001110100111011110

p_m est usuellement très faible et fixée

Convergence prouvée si p_m (k) décroît à chaque génération k en respectant :

$$p_m(k) \geq \frac{1}{2} k^{-1/_{ML}}$$

M est la taille de la population et L est la longueur des chromosomes.

Exemple : le problème « onemax »

= 0101110010100111

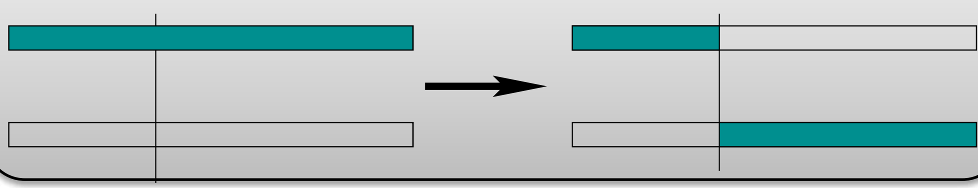
On optimise le nombre de 1

1111111111111111

Mutation

0101110010100111 → 0101110011100111

Croisement



Exemple : le problème « onemax »

Gen : 0	Max : 21.000000	Min : 11.000000	Moy : 15.760000
Gen : 1	Max : 23.000000	Min : 11.000000	Moy : 17.180000
Gen : 2	Max : 22.000000	Min : 13.000000	Moy : 18.340000
Gen : 3	Max : 24.000000	Min : 15.000000	Moy : 19.460000
Gen : 4	Max : 25.000000	Min : 16.000000	Moy : 20.380000
Gen : 5	Max : 26.000000	Min : 17.000000	Moy : 21.320000
Gen : 6	Max : 27.000000	Min : 20.000000	Moy : 23.300000
Gen : 7	Max : 28.000000	Min : 22.000000	Moy : 24.220000
Gen : 8	Max : 28.000000	Min : 21.000000	Moy : 24.820000
Gen : 9	Max : 29.000000	Min : 23.000000	Moy : 25.720000
Gen :10	Max : 30.000000	Min : 22.000000	Moy : 26.260000
Gen :11	Max : 30.000000	Min : 23.000000	Moy : 27.000000
Gen :12	Max : 32.000000	Min : 23.000000	Moy : 27.560000
Gen :13	Max : 32.000000	Min : 24.000000	Moy : 27.840000
Gen :14	Max : 32.000000	Min : 25.000000	Moy : 28.260000
Gen :15	Max : 32.000000	Min : 25.000000	Moy : 28.840000
Gen :16	Max : 32.000000	Min : 25.000000	Moy : 29.540000
Gen :17	Max : 32.000000	Min : 27.000000	Moy : 30.040000
Gen :18	Max : 32.000000	Min : 27.000000	Moy : 30.480000
Gen :19	Max : 32.000000	Min : 29.000000	Moy : 30.700000
Gen :20	Max : 32.000000	Min : 30.000000	Moy : 30.980000
Gen :21	Max : 32.000000	Min : 30.000000	Moy : 31.680000
Gen :22	Max : 32.000000	Min : 31.000000	Moy : 31.920000
Gen :23	Max : 32.000000	Min : 32.000000	Moy : 32.000000
Gen :24	Max : 32.000000	Min : 32.000000	Moy : 32.000000
Gen :25	Max : 32.000000	Min : 32.000000	Moy : 32.000000

Les stratégies d'évolution

La recherche se fait dans $\mathbb{R}^n$

Croisement barycentrique :

$$\forall i \in 1,..,n, \qquad x'_i = \alpha x_i + (1-\alpha) y_i$$

α, **choisi par tirage unifome dans** $[0,1]$ **ou** $[-\epsilon, 1+\epsilon]$

Mutation Gaussienne :

$$\forall i \in 1,...,n, \qquad x'_i = x_i + N(0,\sigma)$$

deux paramètres p_m **et** σ**.**

Mutation Log-normale auto-adaptative :

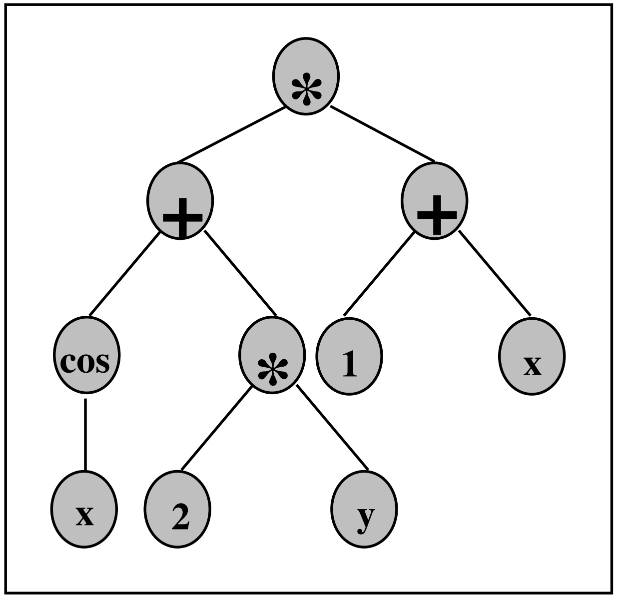
σ **est intégré au code génétique :** (x,σ)

$$\forall i \in 1,...,n, \qquad \sigma'_i = \sigma_i exp(N(0,\tau))$$

$$x'_i = x_i + N(0,\sigma'_i)$$

La programmation génétique

Un individu est une formule mathématique, représentée à l'aide d'un arbre.



$(1 + x)\ (2y + \mathrm{Cos}\ x)$

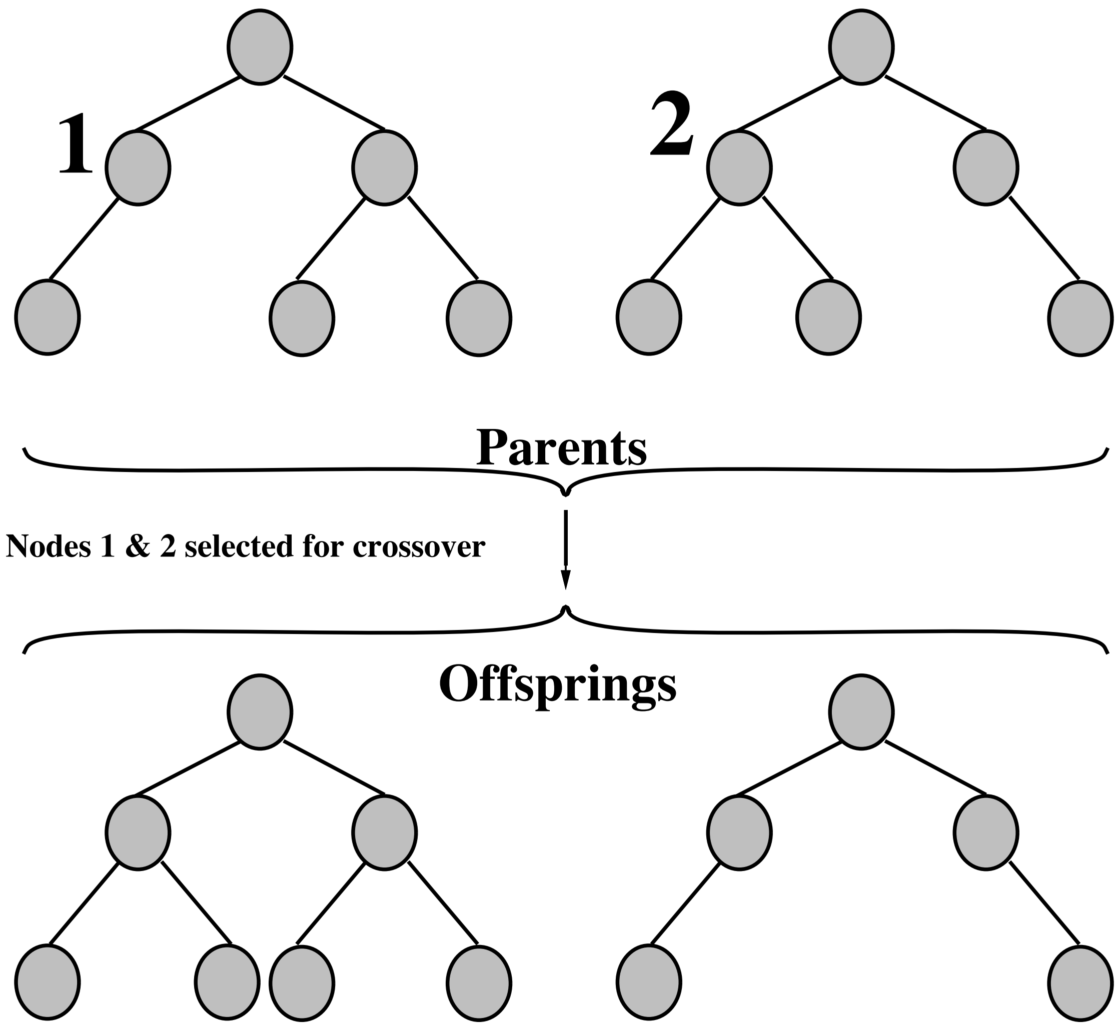
→**Créer des programmes …**

… sans programmer !

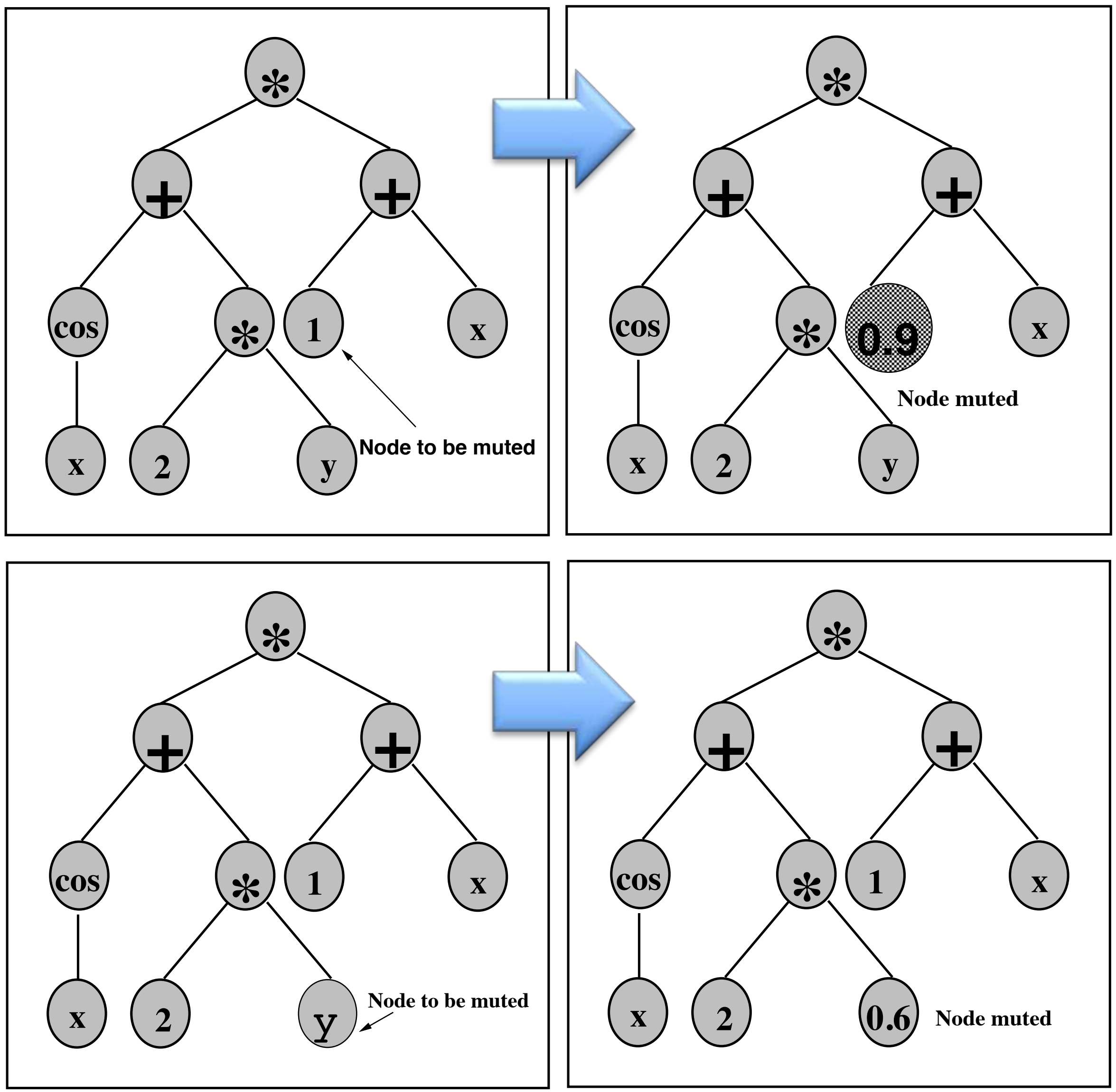
(John KOZA)

Evolution de structures de tailles variables !

Le croisement PG



Les mutations PG



Résumé

- Les AE fonctionnent sur une population de solutions (phénotype / génotype).

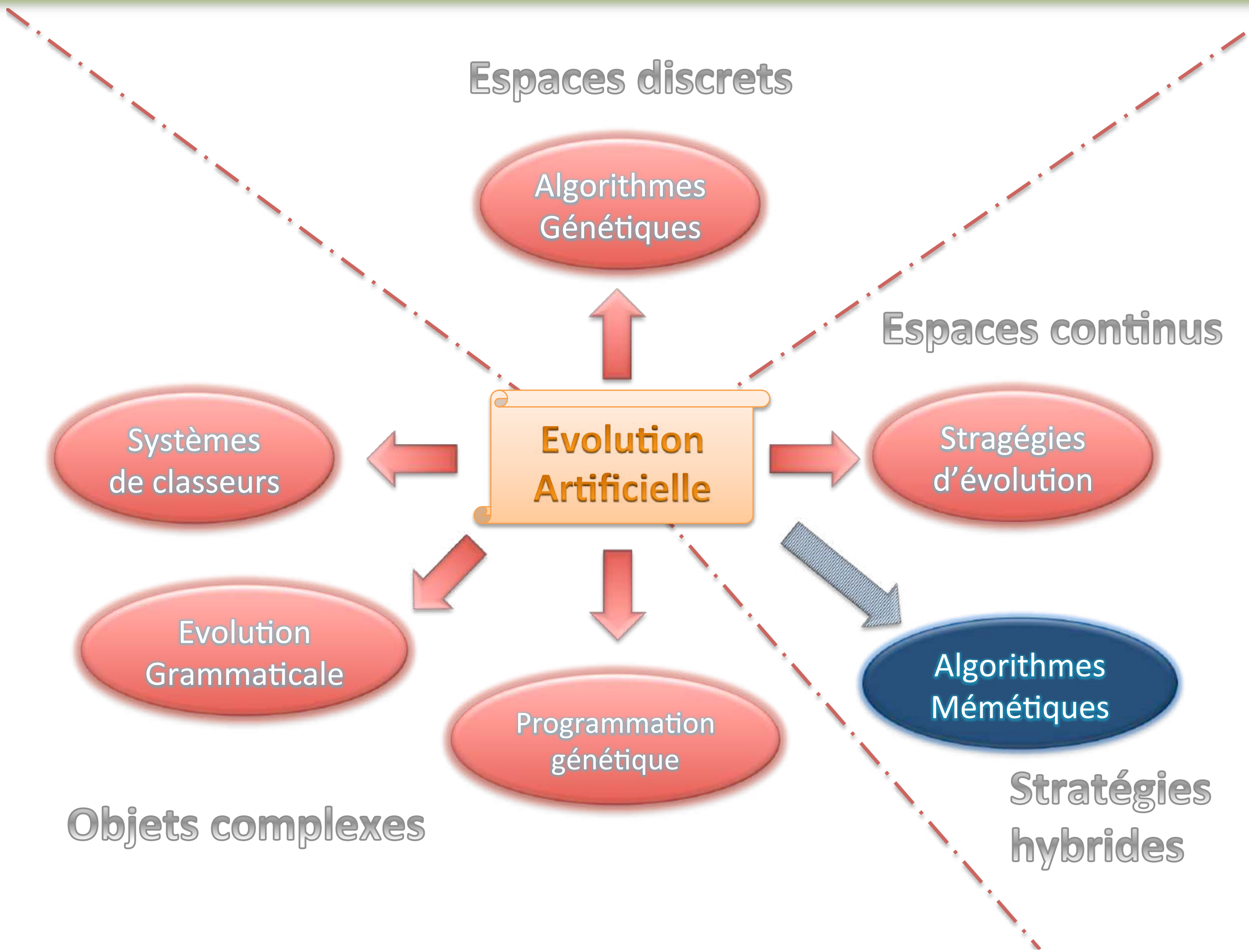
- La fonction à optimiser est utilisée pour donner un classement (sélection): pas de contrainte de continuité ni de dérivabilité.

- La convergence se traduit par une concentration de la population autour de l'optimum global.

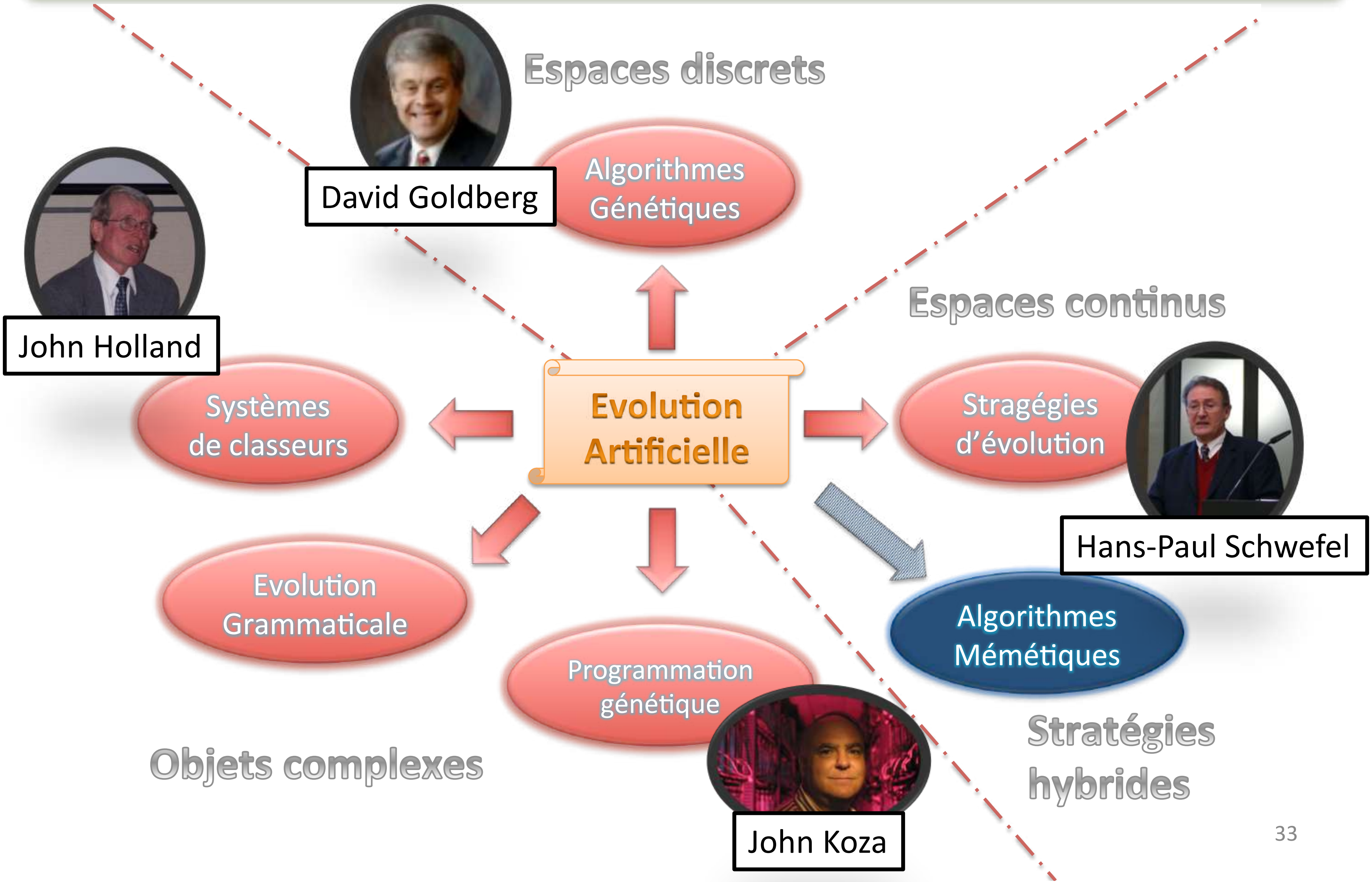
- La vitesse de convergence et la précision de la solution dépendent :
    - de la représentation (codage) des individus,
    - de la forme de la fonction à optimiser,
    - des paramètres de l'AE.

- Les temps de calculs en version séquencielle peuvent être longs.

Un ensemble d'approches



Le Who's who


Espaces discrets
David Goldberg
Algorithmes Génétiques
John Holland
Espaces continus
Systèmes de classeurs
Evolution Artificielle
Stragégies d'évolution
Hans-Paul Schwefel
Evolution Grammaticale
Algorithmes Mémétiques
Programmation génétique
Stratégies hybrides
Objets complexes
John Koza

Comment fonctionne un AE ?

Le compromis exploration exploitation

Recherche aléatoire

Mutation
Croisement
(tirages aléatoires)

Echantillonnage
de l'espace
Par une population

Recherche en parallèle

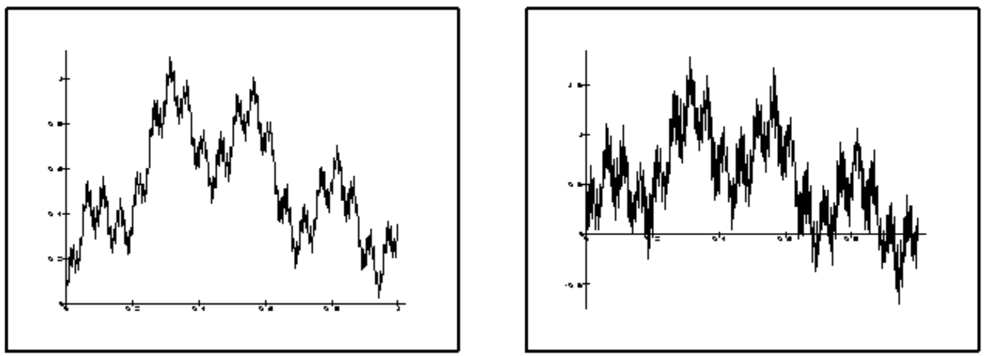
Recherche dirigée

Sélection
Croisement
(concentration)

Quand utiliser un AE ?

cas linéaire, convexe : trop lent !

cas plus complexe : exemple des fonctions fractales



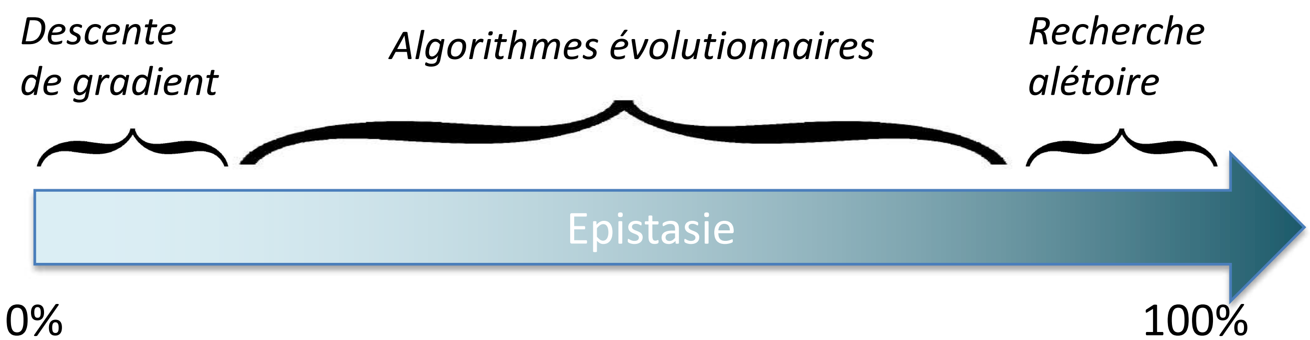
Fonctions de Weierstrass de dimension 1.5 et 1.7

cas pseudo-aléatoire : exemple d'une fonction en cryptographie

$$D_p\{x^4 \text{ mod } p, a\} \text{ avec } a \text{ et } p \text{ entiers codés sur 500 bits, } a < p$$
$$D_p\{x, y\} = Min\{(x-y) \text{ mod } p, \{(y-x) \text{ mod } p\}$$

Recommandations pratiques

Optimisation statique, environnement complexe et multidimensionnel.



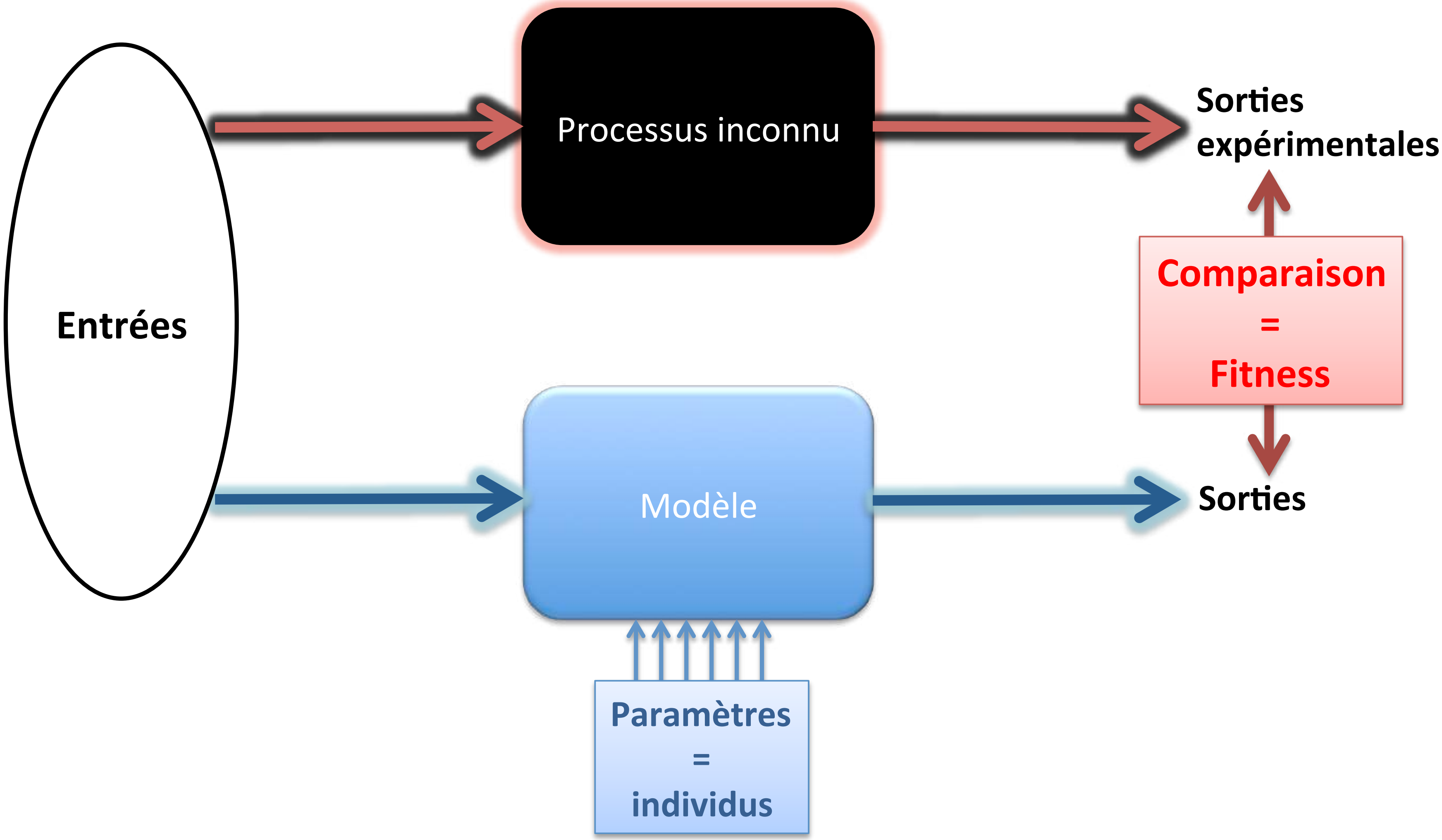
Environnement variable : IA, contrôle, commande de processus.

Des applications très variées

théorie des graphes
théorie des jeux
robotique
modélisation de données 3D
traitement d'images
synthèse d'images
traitement de la parole
applications militaires
économie, finance

fiabilité, test de logiciel
linguistique
science des matériaux, mécanique
physique théorique (synchrotron)
géophysique
architecture durable
écologie
génétique
médecine

L’approche inverse en « boîte noire »



Construire un AE efficace

- Identifier l'espace de recherche et les contraintes du problème.
- Choisir un codage efficace (gérer la redondance).
- Construire une fonction de fitness économique en temps de calcul.
- Bien **régler** les opérateurs, stratégies, paramètres.

Algorithmes évolutionnaires « non standards »

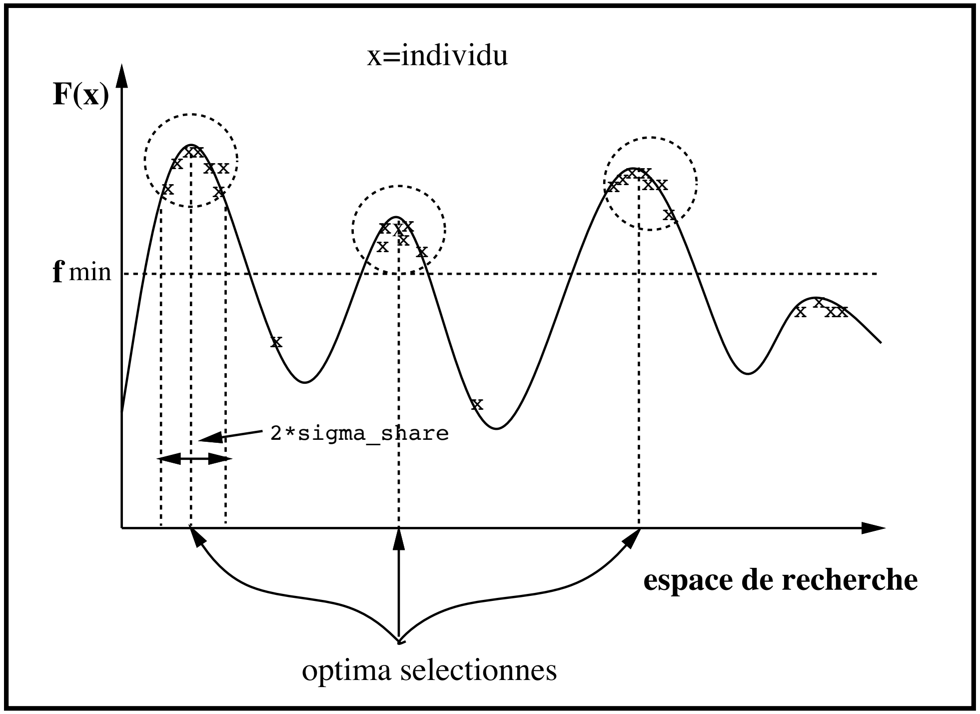
Niches écologiques, pour détecter plusieurs optima.

Optimisation multi-objectif, dominance de Pareto.

Coopération coévolution.

Nichage en environnement multimodal

Pour ``coloniser'' plusieurs optima



- Par maintien de la diversité génétique *(Caviccio, De Jong, Mauldin)*

- Par partage des ressources *(Goldberg, Richardson)*

$$\text{Fitness'}(x) = \frac{\text{Fitness}(x)}{\sum_{x' \in Vois(x)} Sh(d(x,x'))}$$

$$Sh(d) = \begin{cases} 1 - (\frac{d}{\sigma_{share}})^{\alpha} & \textbf{si } d < \sigma_{share} \\ 0 & \textbf{sinon} \end{cases}$$

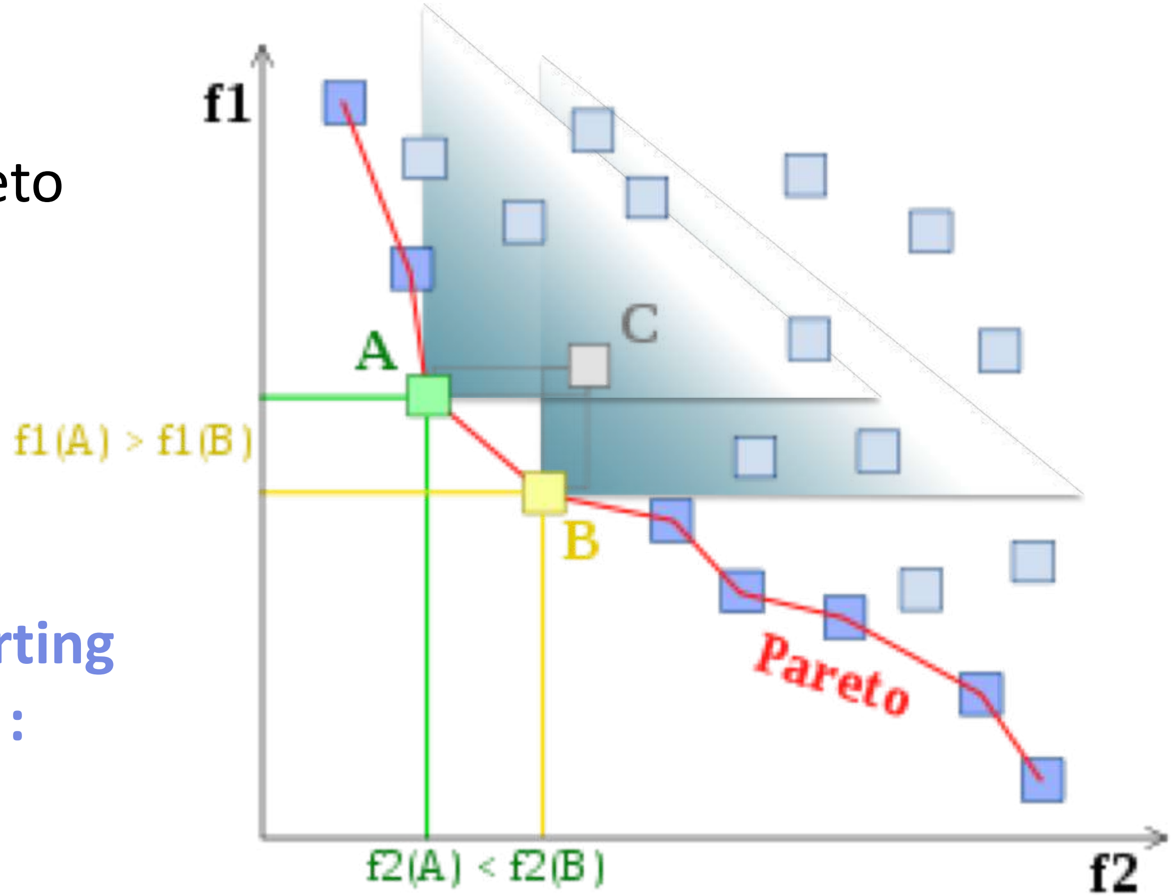
- Par restriction des croisements.

Multi-objectif évolutionnaire

Front de Pareto = ensemble des solutions non dominées

- Sélection par dominance de Pareto

  \+ nichage



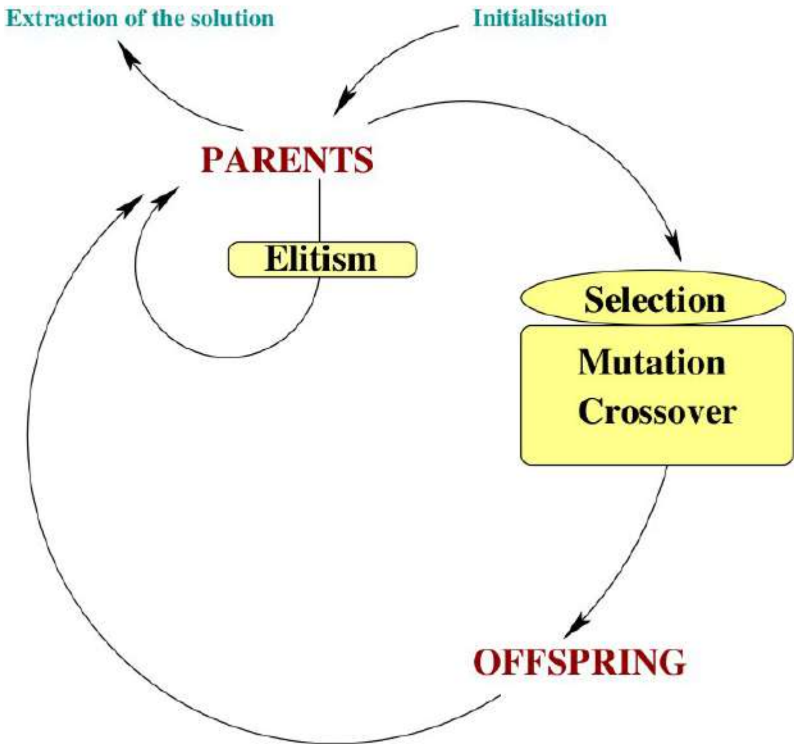
- **Nondominated Sorting Genetic Algorithm :** NSGA-2

  (K. Deb et al)

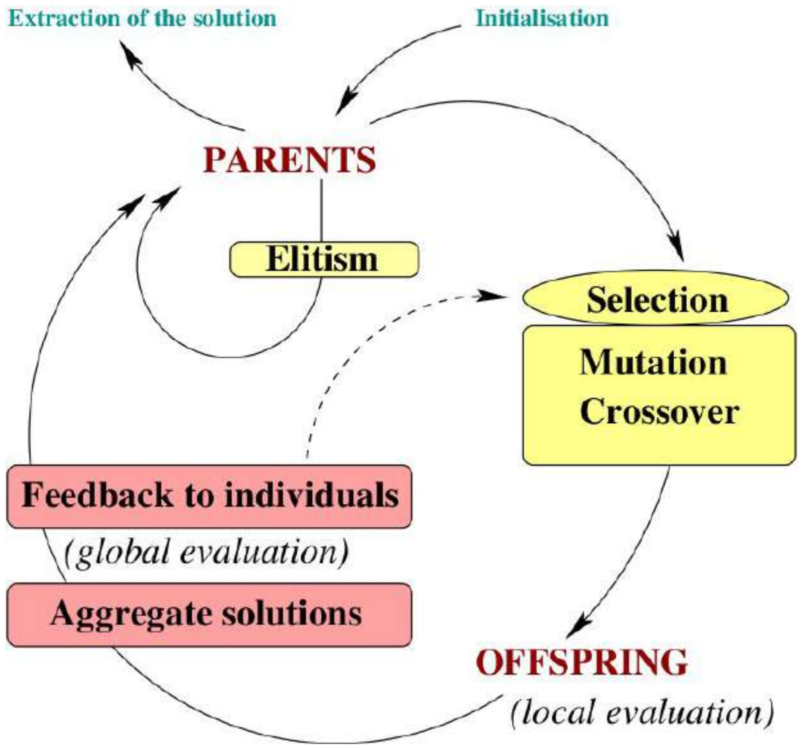
En une fois, un EA produit un échantillonnage du front de Pareto.

Coévolution coopérative

Une solution = plusieurs individus



Approche classique



Coévolution coopérative
ou « approche Parisienne »

Résoudre des problèmes en coévolution

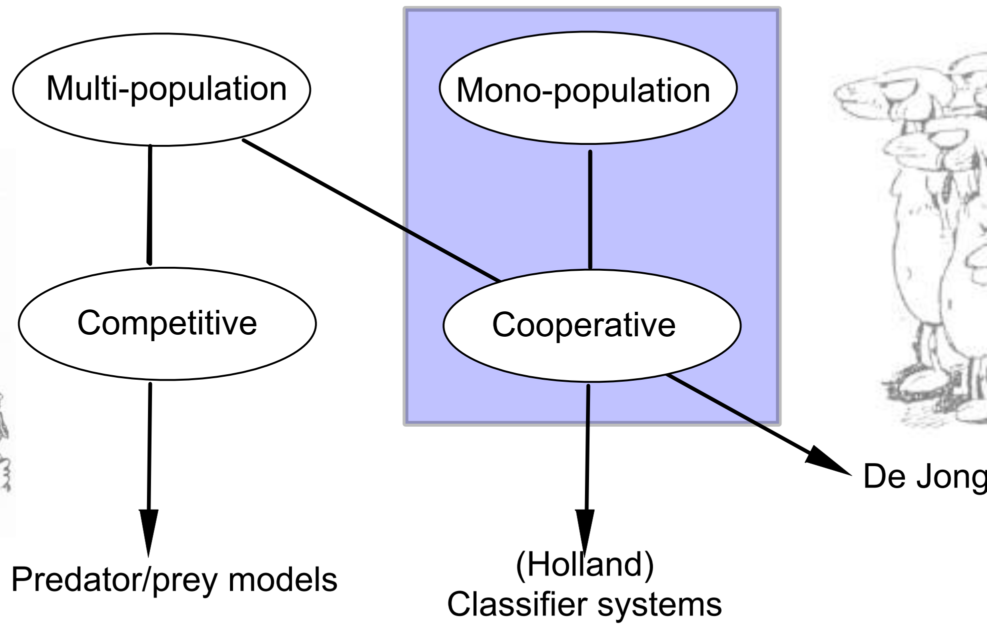
Une solution = plusieurs individus

- La fonction à optimiser est calculée au niveau de la population.

- Il est possible (mais pas obligatoire) d'utiliser deux fonctions de fitness :

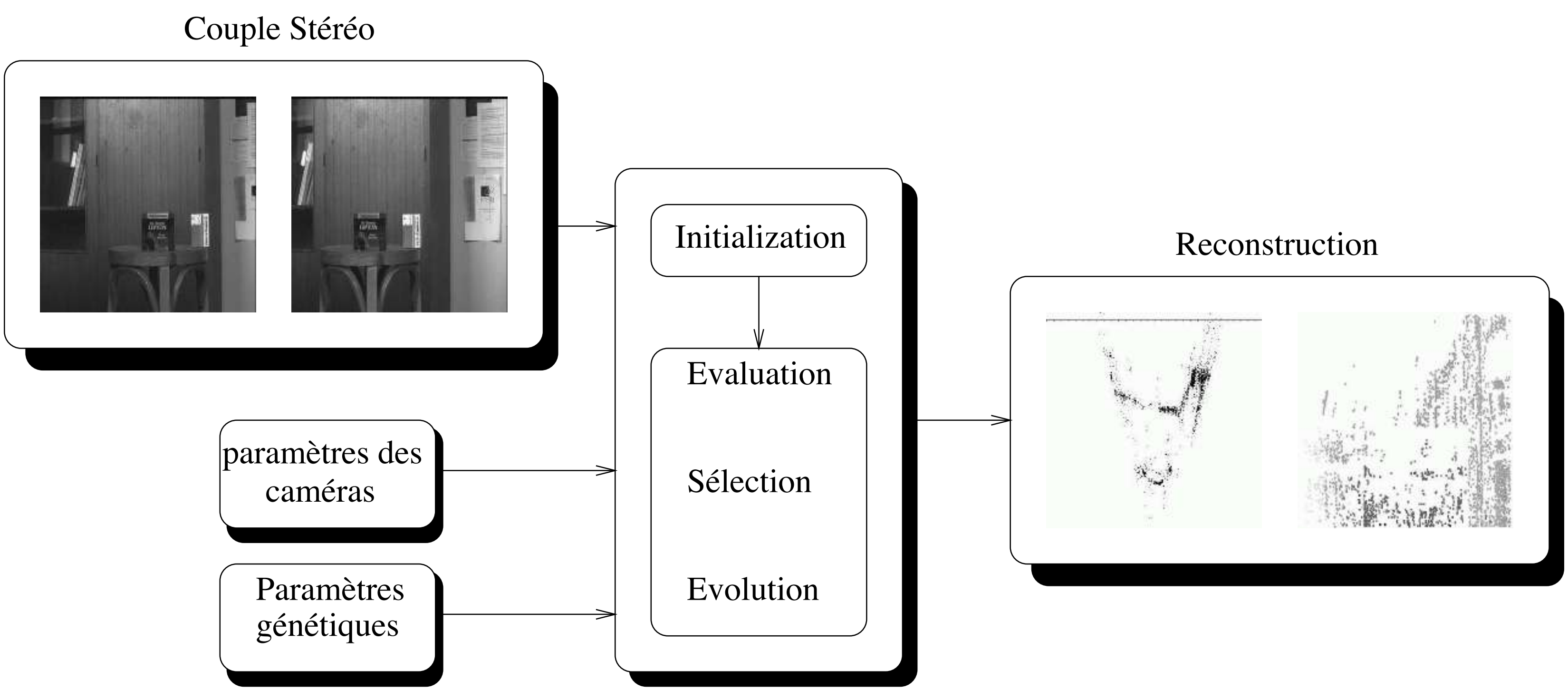
  fitness locale et fitness globale.

Co-évolutions



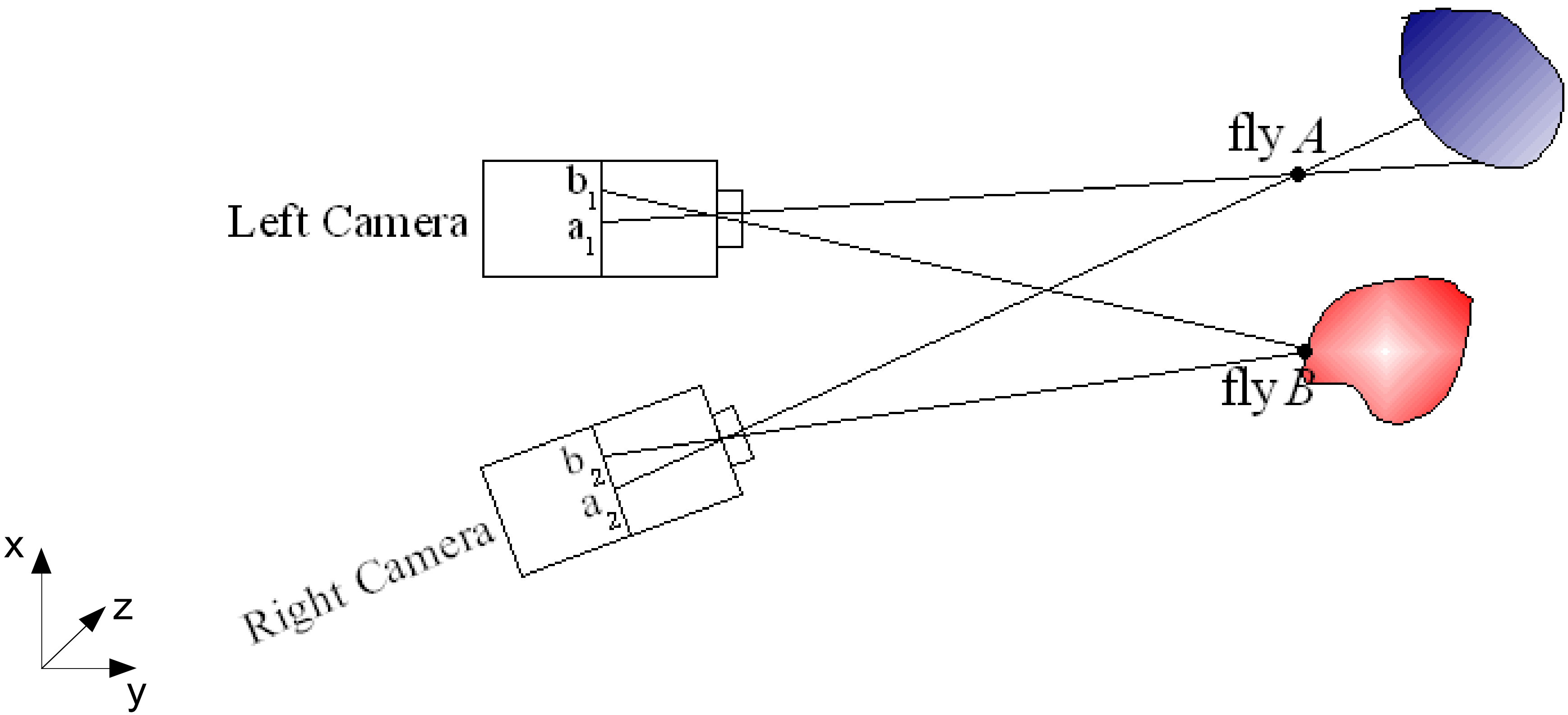
Parisian approach

Exemple : l'algorithme des mouches

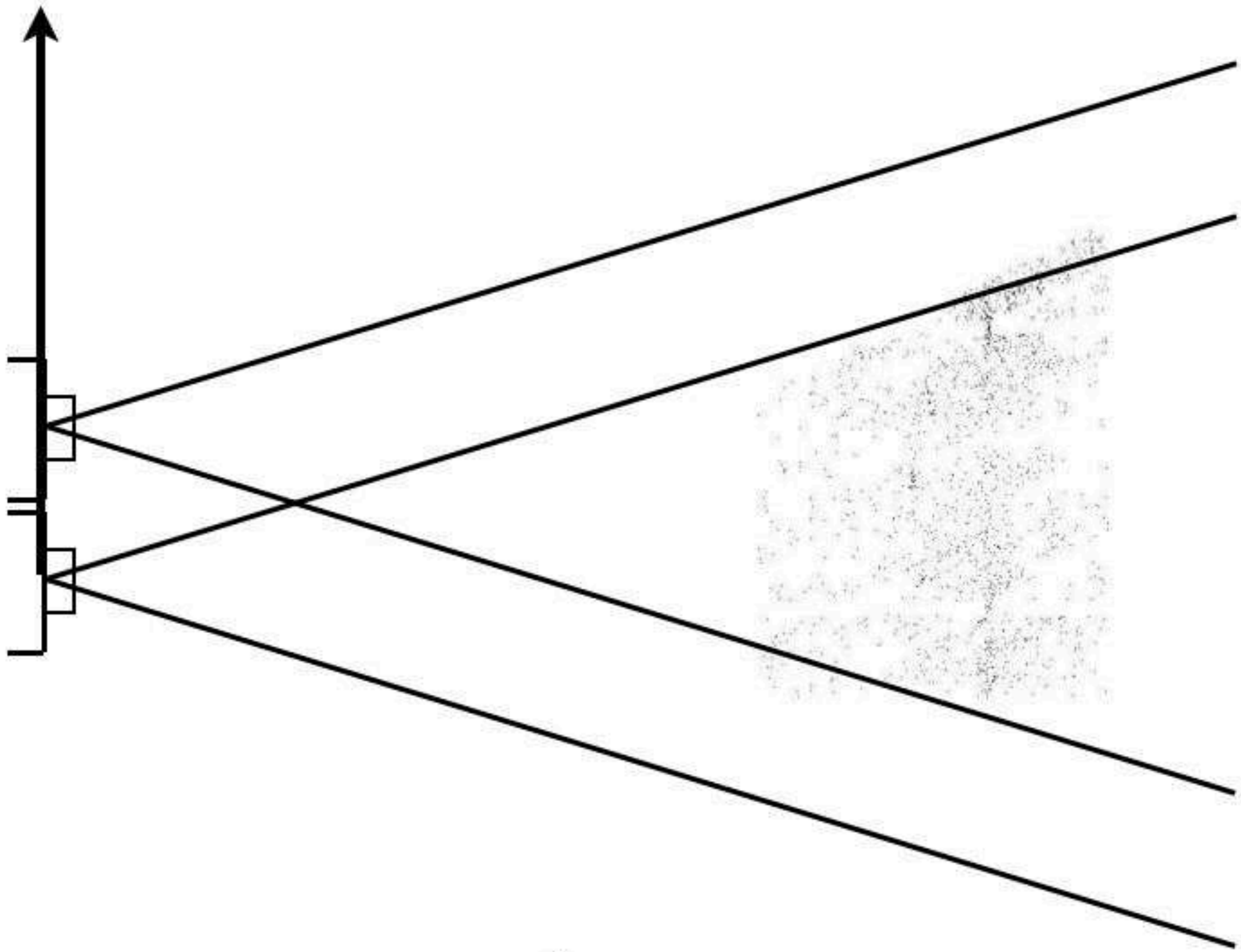


- Un fitness local seulement …
- Une application temps-réel !

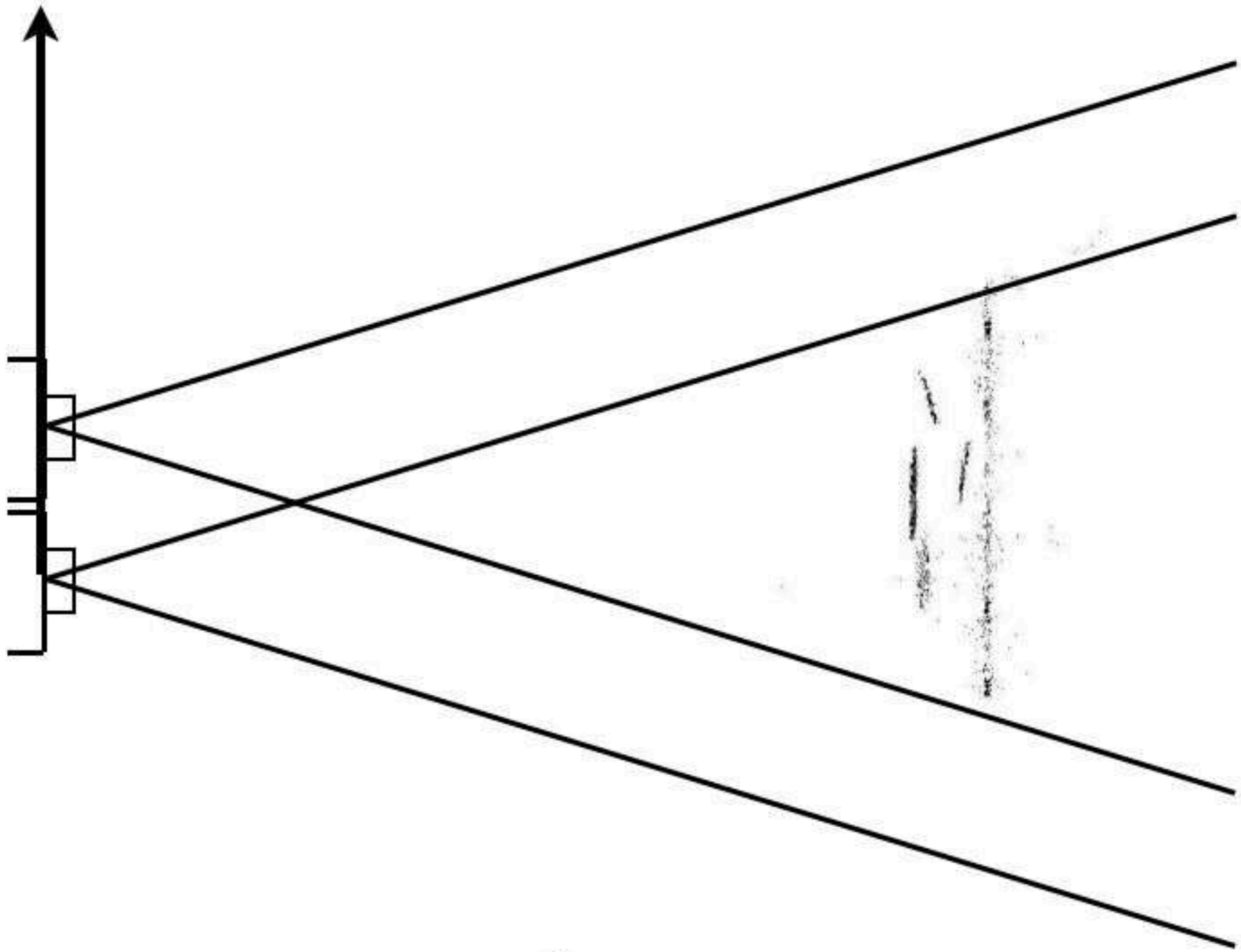
Algorithme des mouches



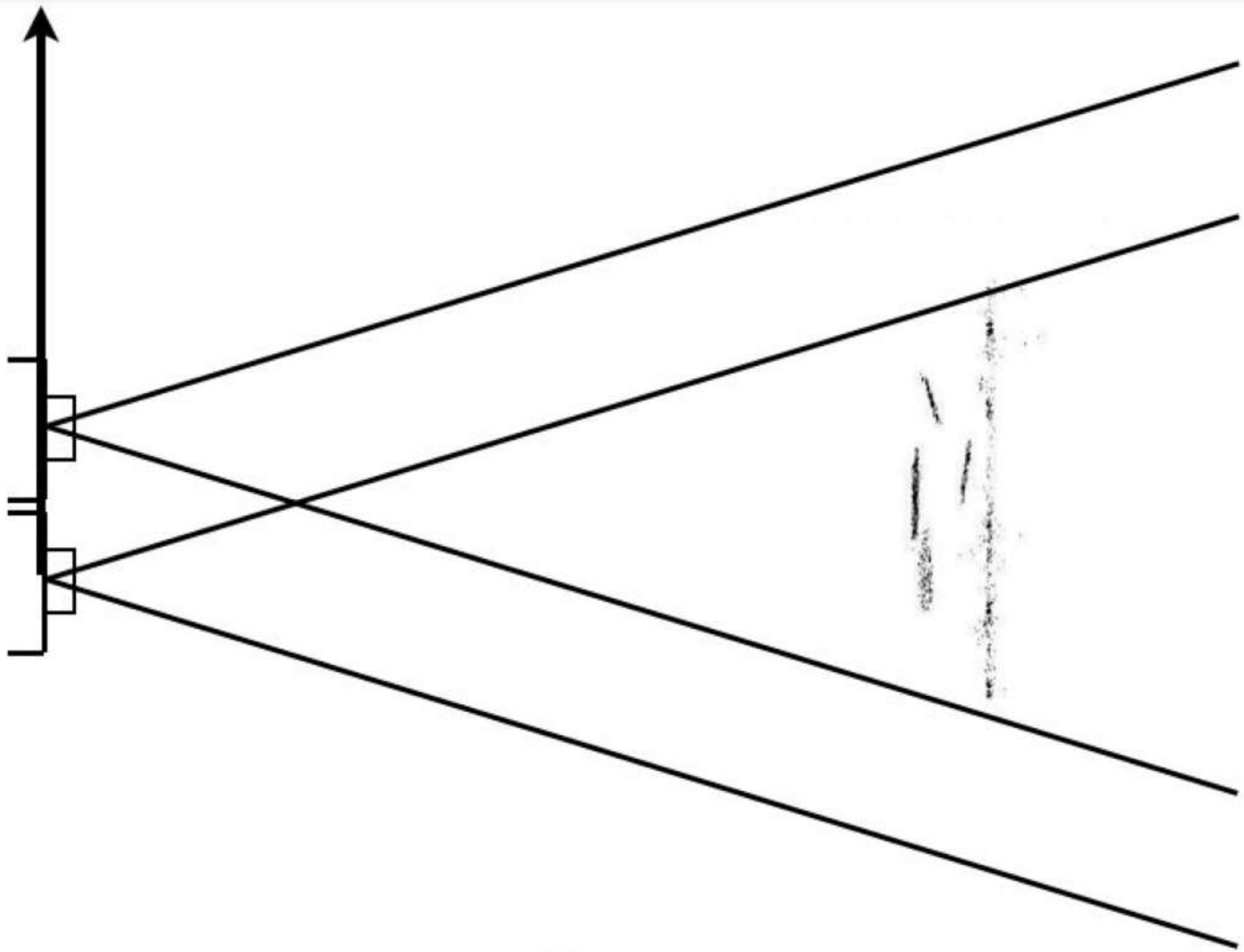
Algorithme des mouches



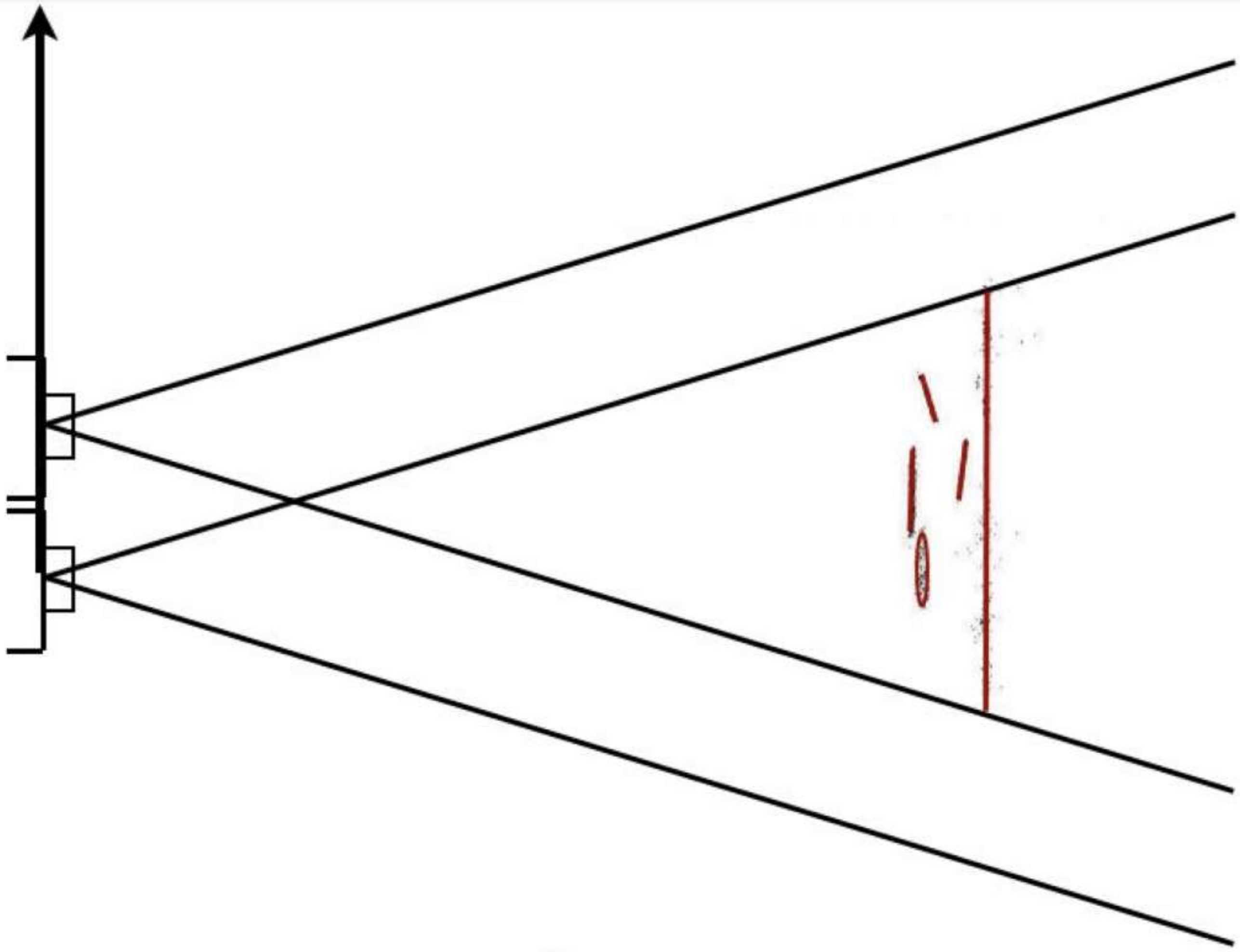
Algorithme des mouches



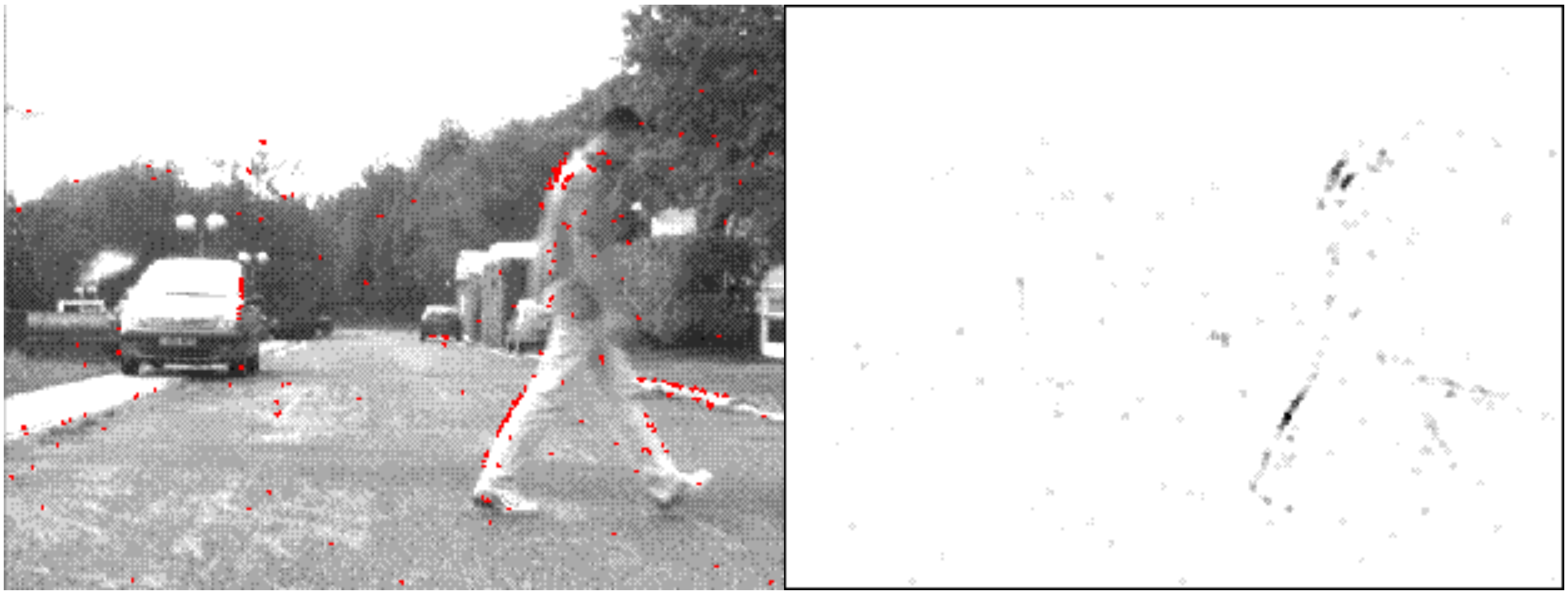
Algorithme des mouches



Algorithme des mouches



Application à la détection d'obstacles



Pourquoi ça fonctionne ?

- L'évolution est toujours conduite par le fitness vu par un individu : *Fitness = F(local, global)*
- La résolution repose toujours sur la capacité de la population à se concentrer dans de « bonnes » zones.
- La diversité est encore plus importante pour éviter des solutions « dégénérées ».

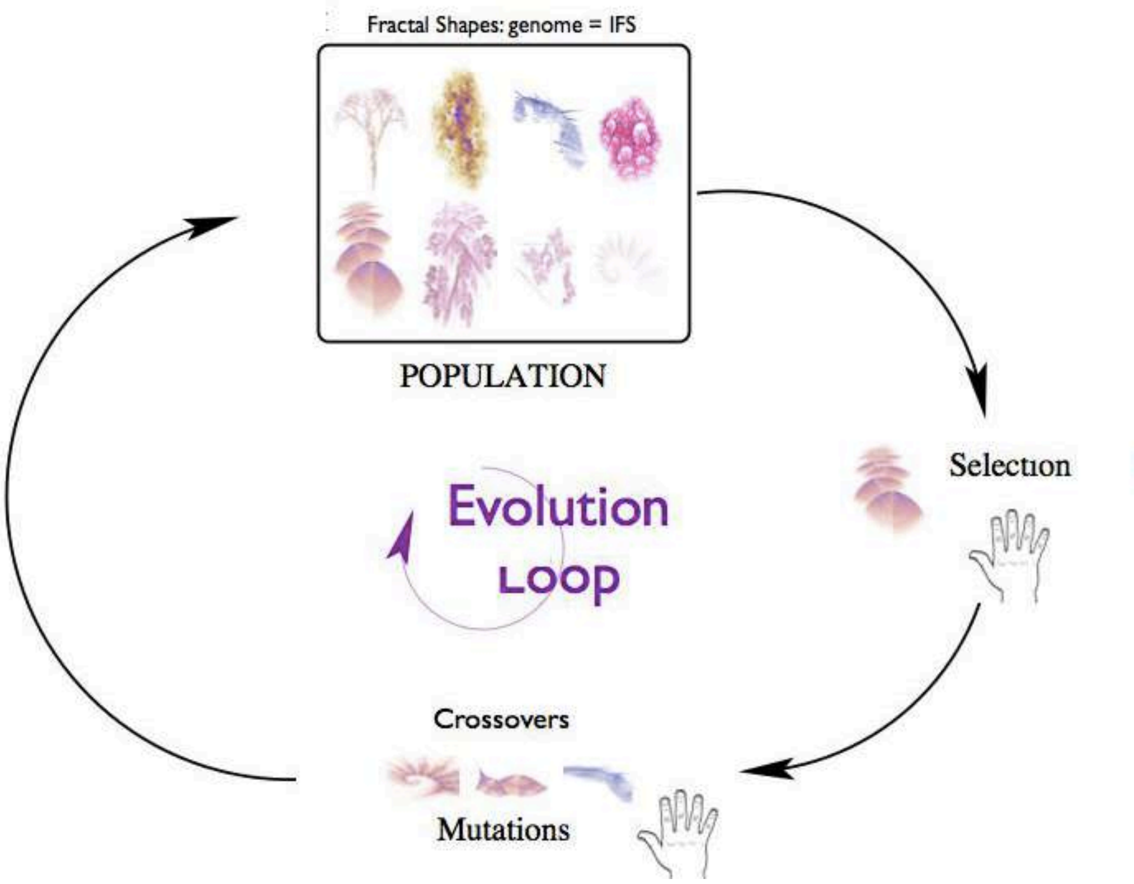
Le design d'un CCEA est plus délicat

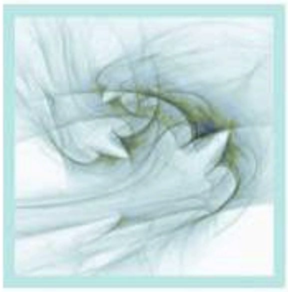
- Espace de recherche : comment découper la solution en (petites) composantes identiques qui interagissent ?
- Equilibre entre fitness local et fitness global (design et calcul) : calculs courts au niveau local vs calculs longs au niveau global
- Métrique pour la gestion de la diversité et l'identification des clusters.

Mais le bénéfice peut être grand !

- Rapidité : EAs en temps réel ! *(algorithme des mouches)*
- Gestion des contraintes au niveau local.
- Robustesse.
- Scalabilité.

Optimisation interactive



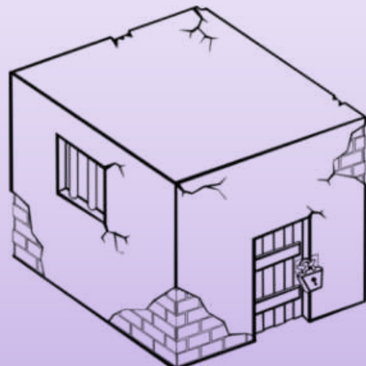


→ **Prise en compte d'une expertise humaine dans la boucle évolutionnaire**

→ **Mais pour interagir efficacement, il faut visualiser !**

Fatigue, surcharge de l'utilisateur

TROP PEU D'INTERACTION, RÉPÉTITIVITÉ :
LASSITUDE, ENNUI.

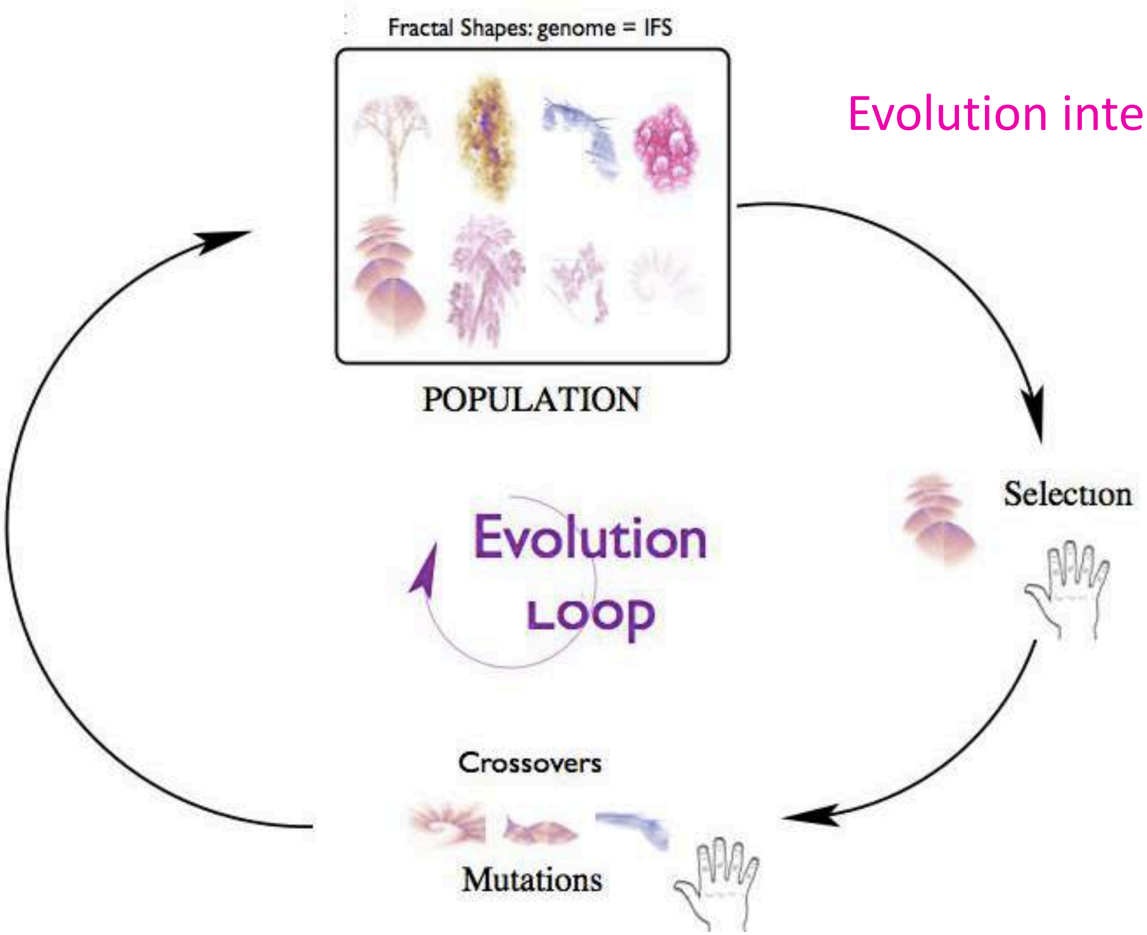


TROP DE CHOIX POSSIBLES :
ÉGAREMENT DANS UN OCÉAN D'INFORMATIONS

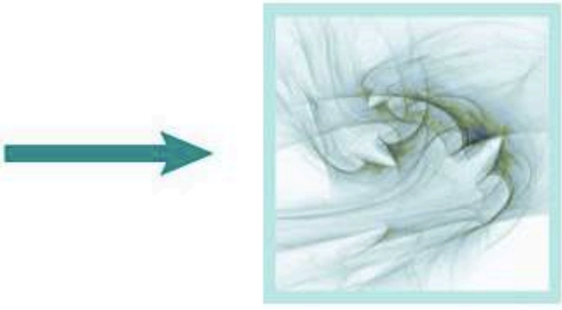
Compromis :

- fluidité / adéquation des visualisations *(interaction)*
- liberté de navigation *(exploration)*
- focalisation sur les zones intéressantes *(optimisation)*

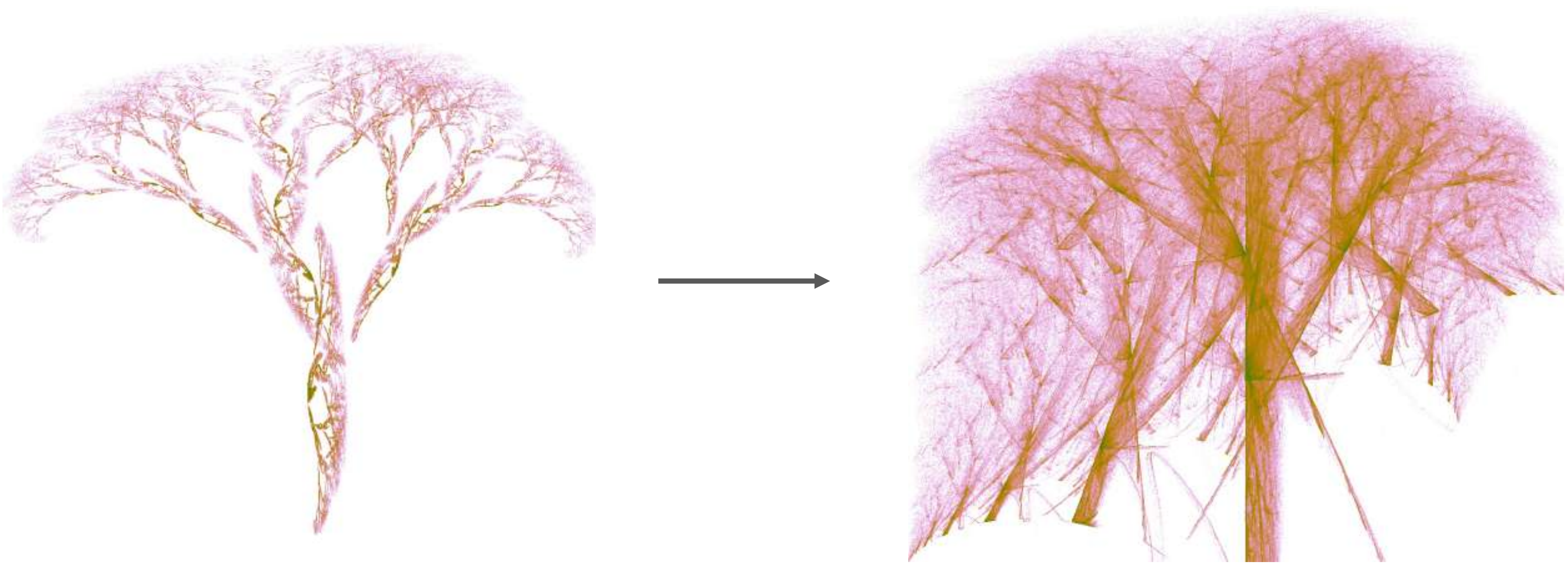
L'élevage de formes fractales …



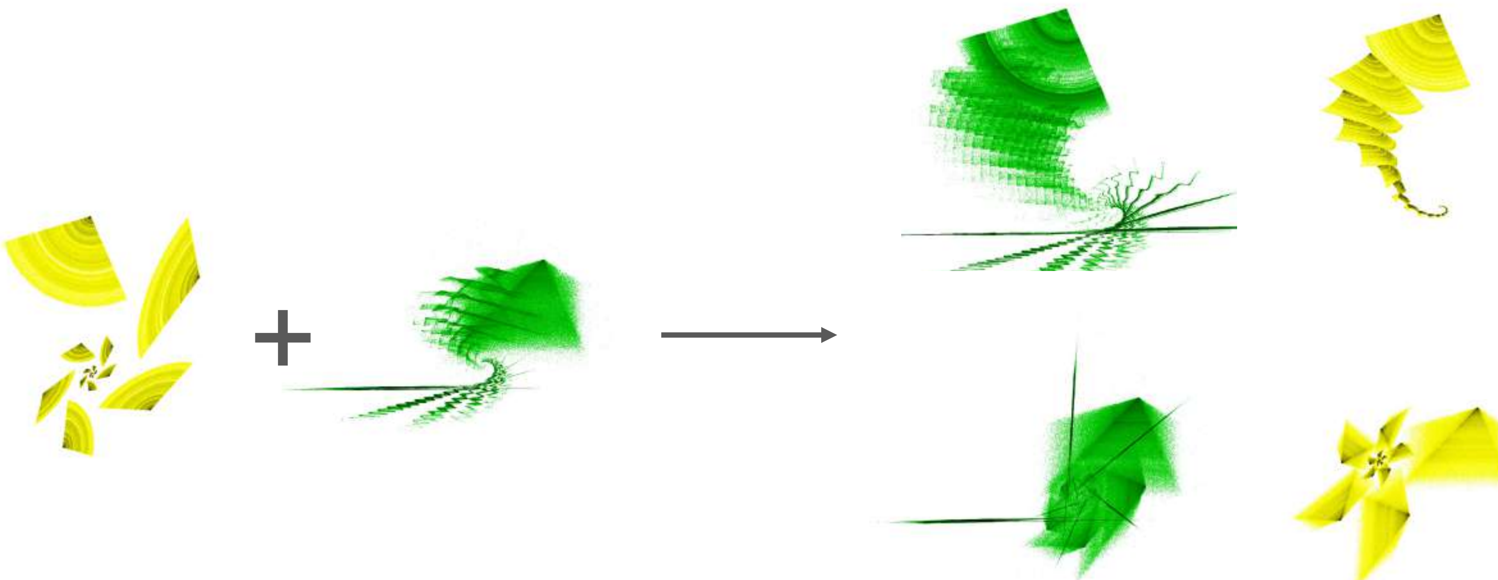
Evolution interactive de formes fractales



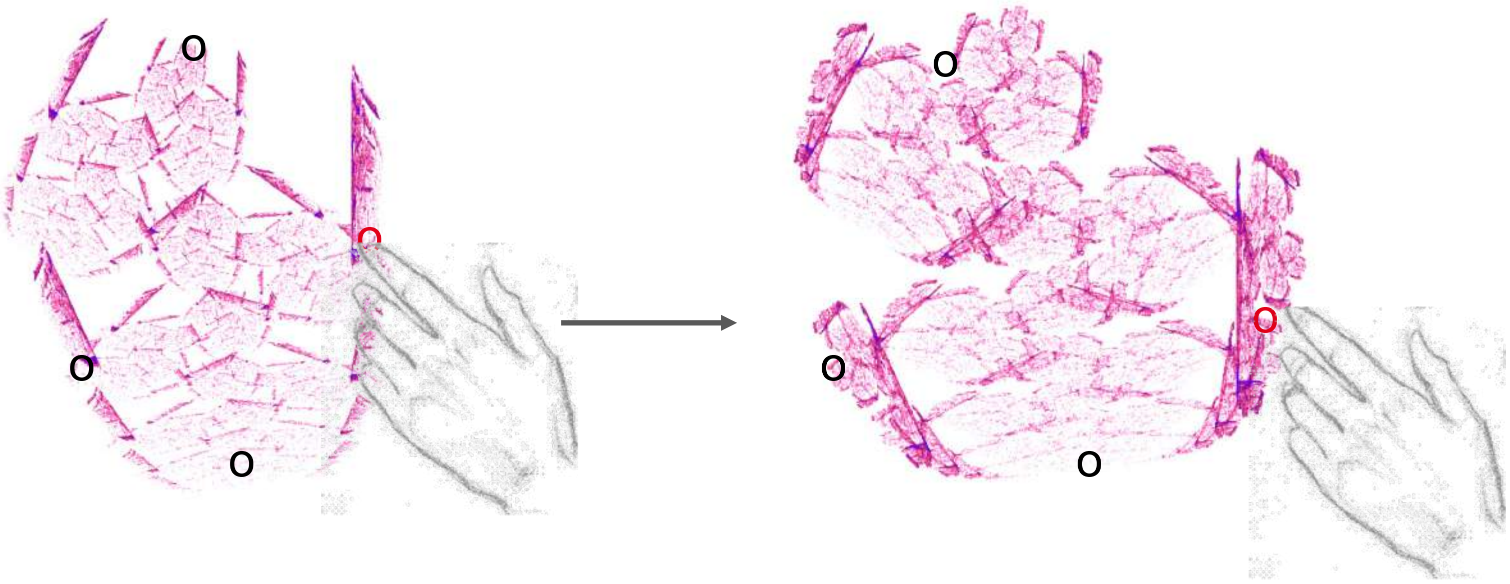
Mutation aléatoire



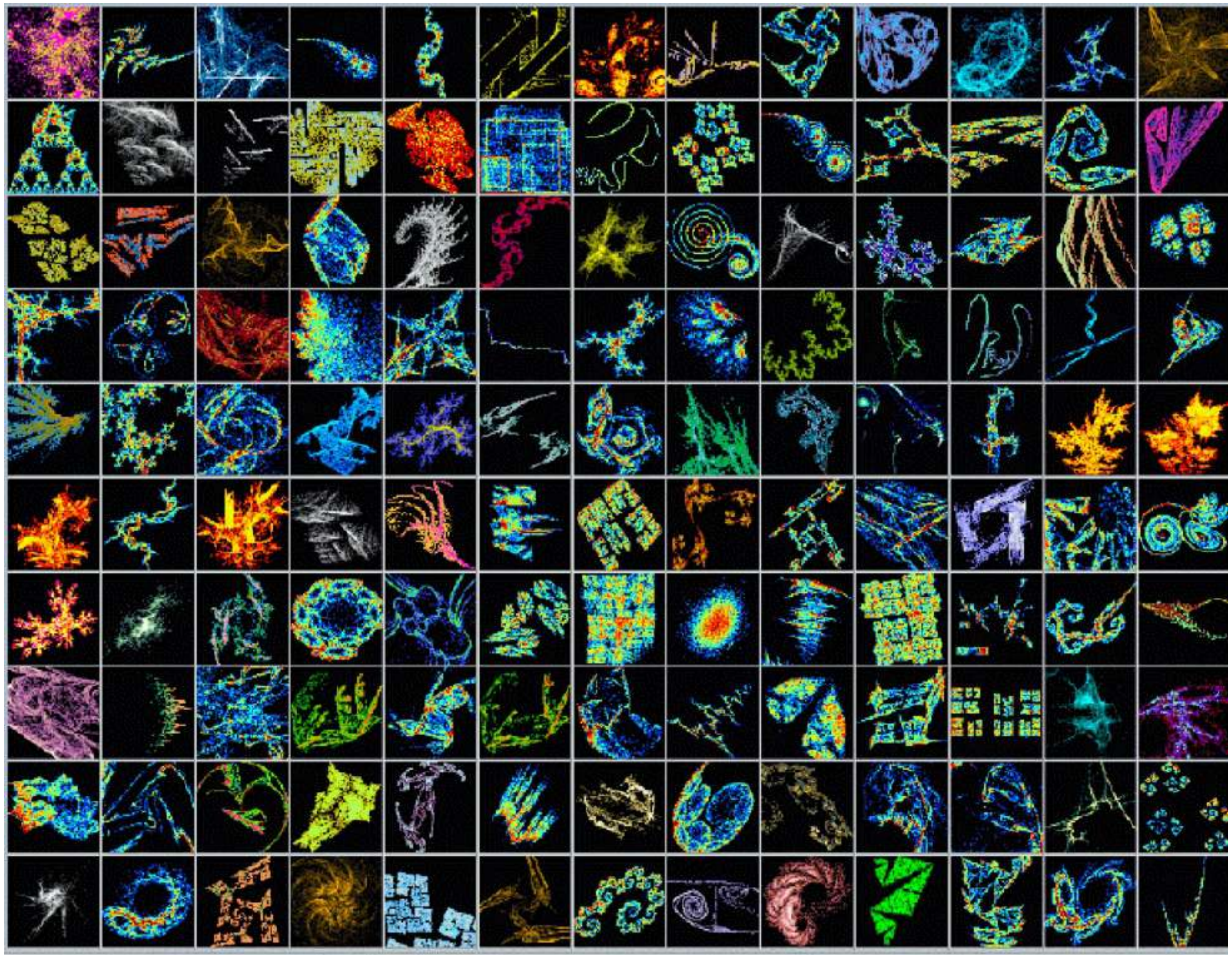
Croisement aléatoire



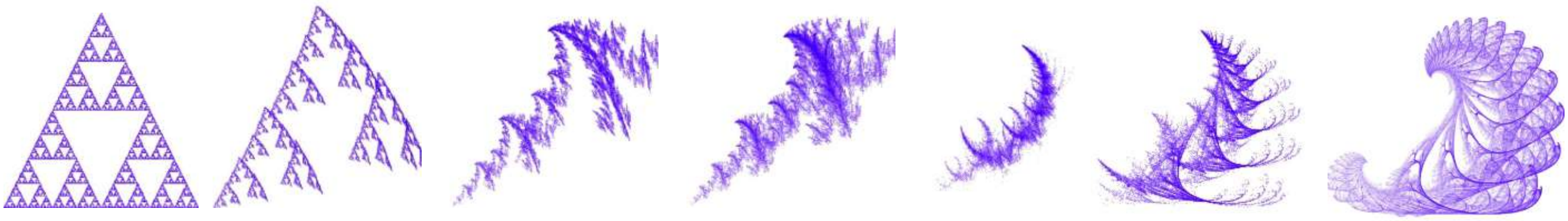
Mutation interactive



Le designer oriente l'évolution de sa population de formes

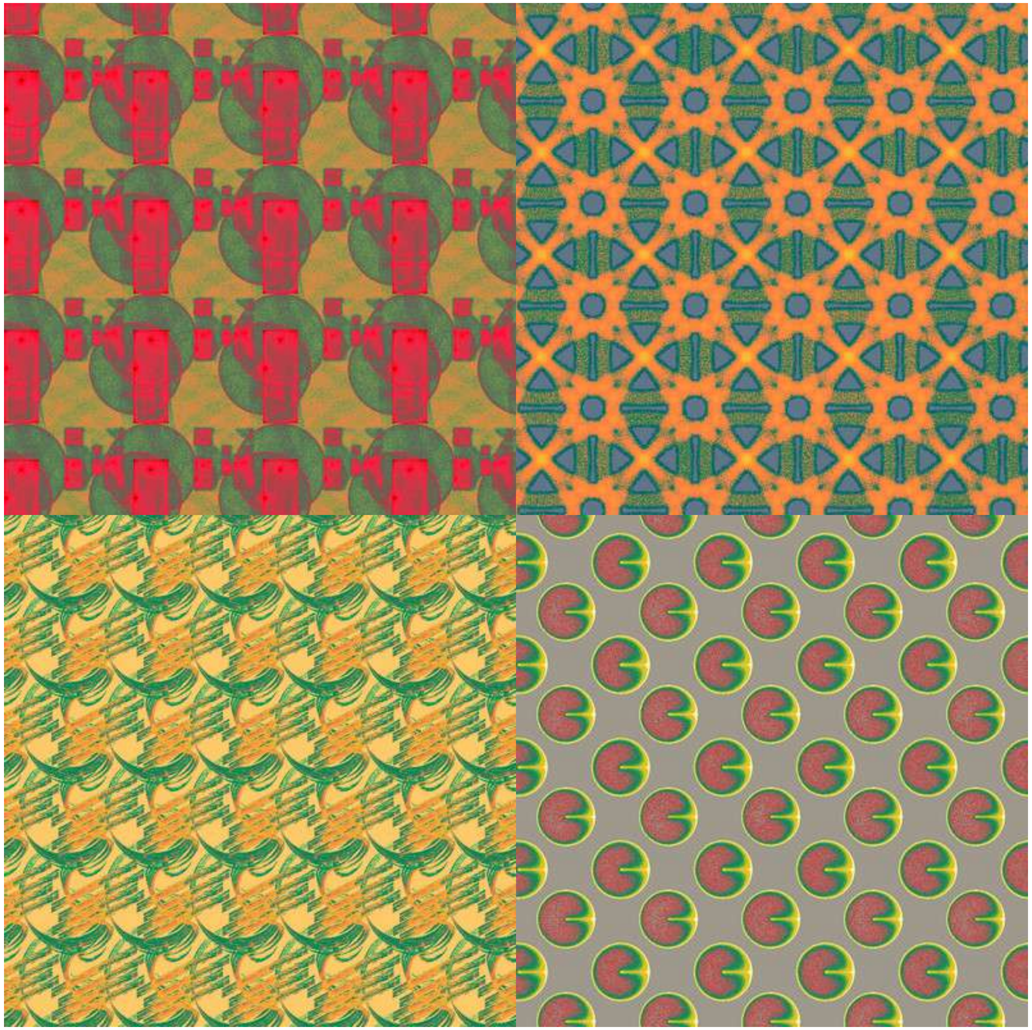


On peut aussi fabriquer des animations

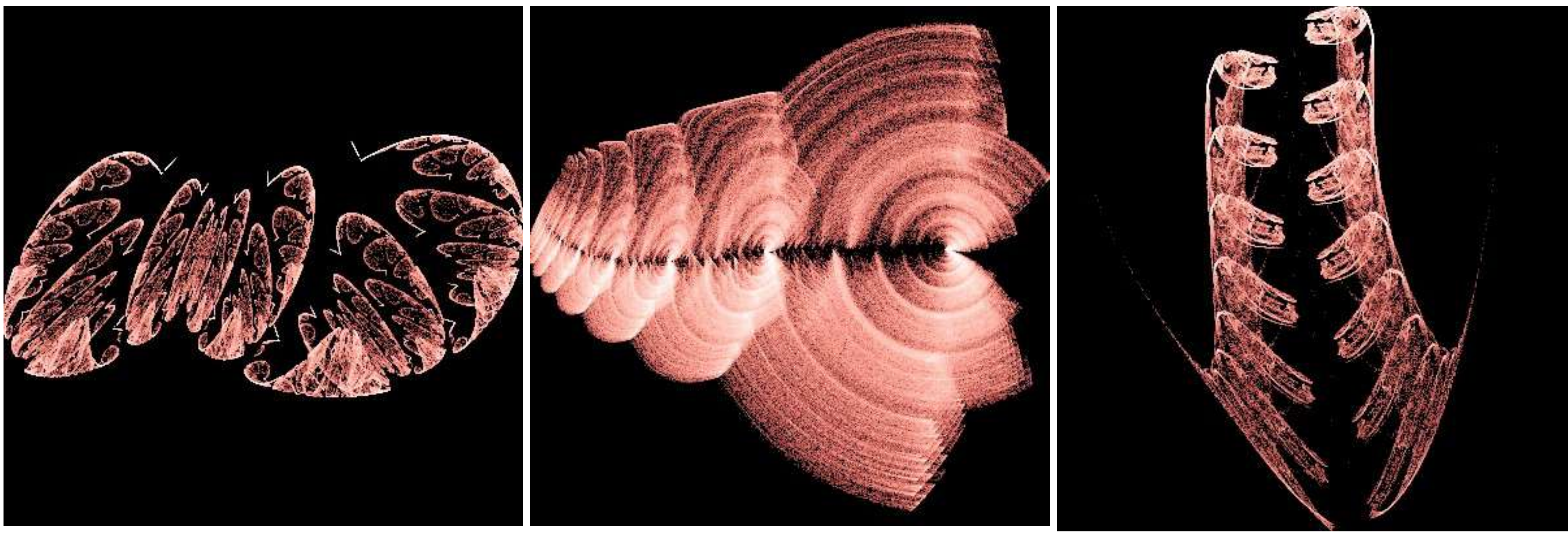


Morphing fractal

Ou des motifs répétitifs



L’outil s’adapte au style de l’utilisateur



Jonathan Chapuis, ingénieur développeur

Fond d’écran, Pierre Grenier, ingénieur développeur

Logos, Evelyne Lutton

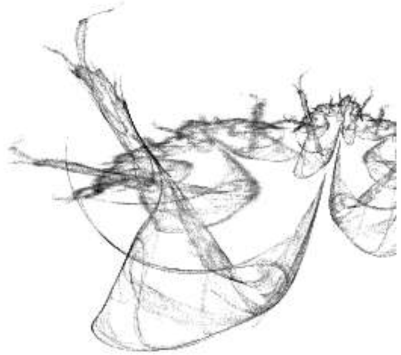
Complex

APIS

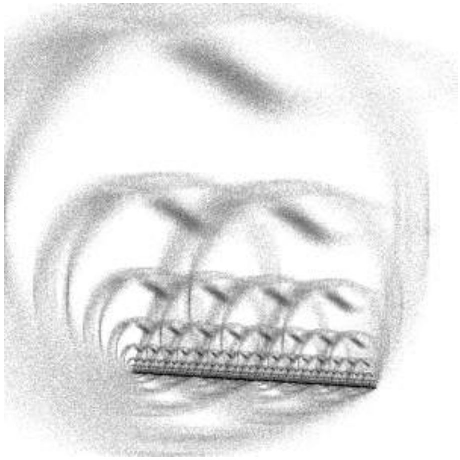
IDEAS

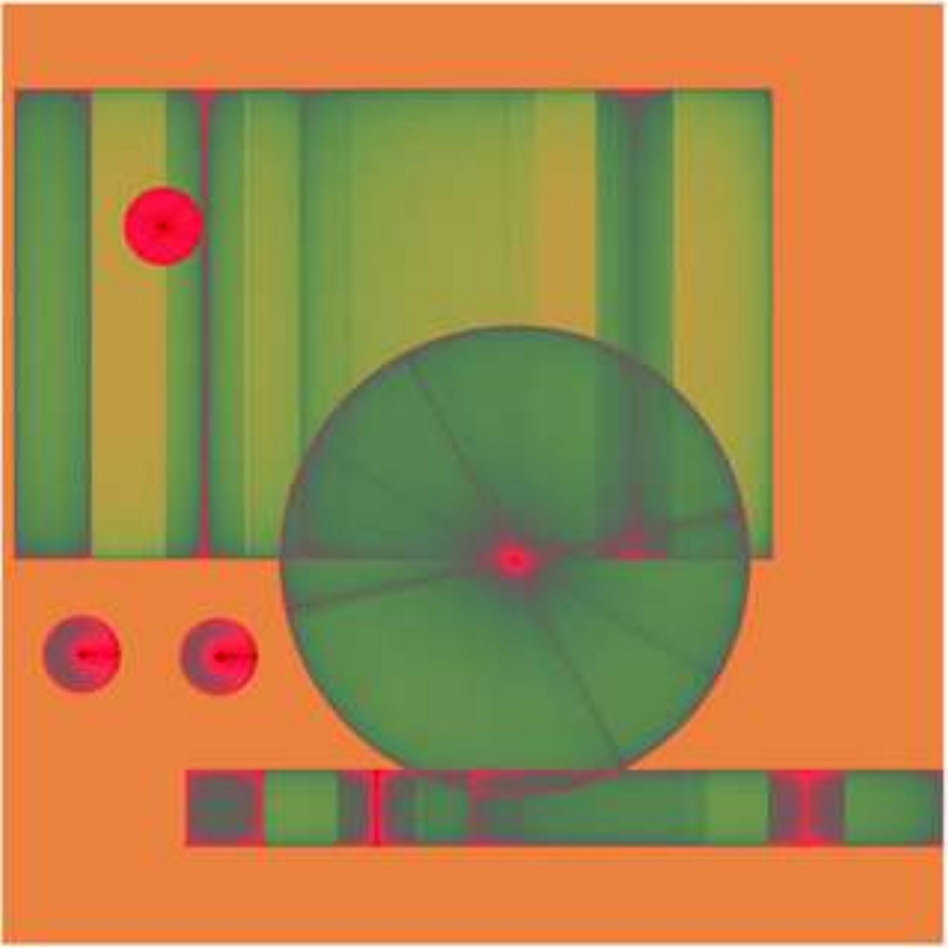
INCALIN

Attracteurs fractals par Emmanuel Cayla







LENTIL_DEFACE_BOZ_002 *Cayla, l'atelier des fractales*

$$f_1 = \begin{cases} exp\left(\left(\left(noisex\left(\theta,\rho\right)\right)0.57\right)^2\right) - 1\right) \\ noisex\left(-0.0018, log\left(noisex\left(\rho,-0.27\right)\right)\right) \end{cases}$$

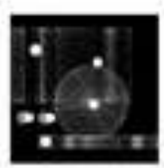
$$f_2 = \begin{cases} sin\left(noisexn\left(0.083,-1.4\right)\right) \\ \left(\left(\left(\left(-8.5-\rho\right)^4\right)+0.11\right)^{0.5}\right)+1.7 \end{cases}$$

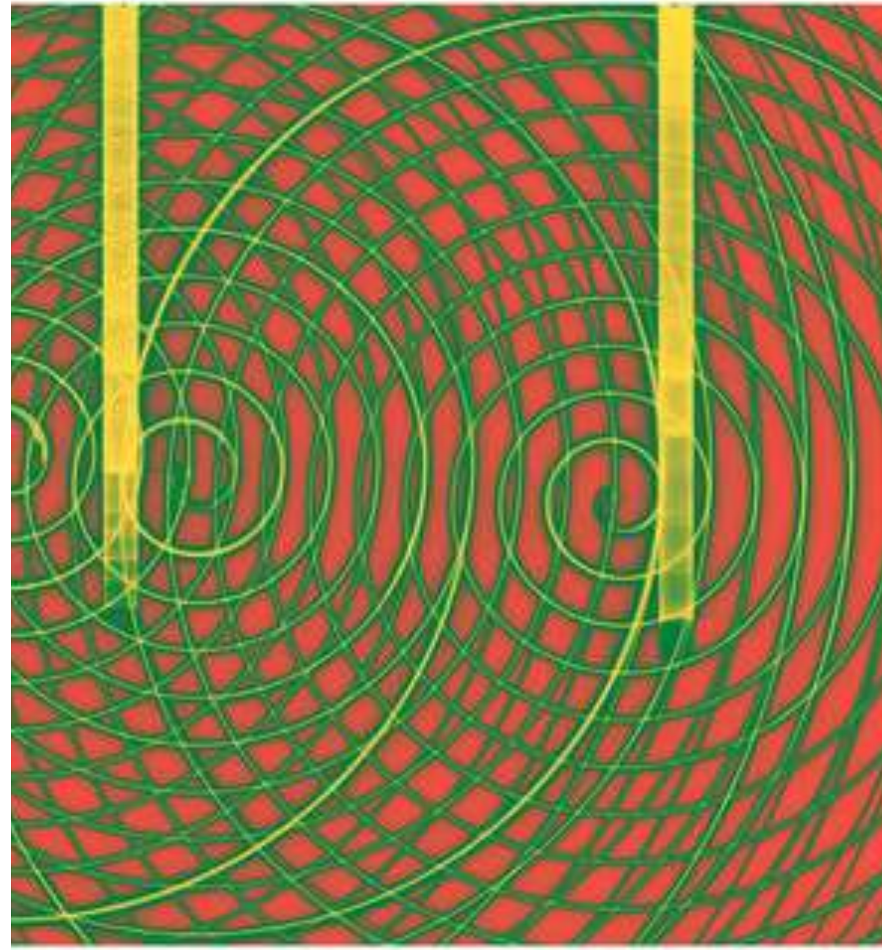
$$f_1 = \begin{cases} y+1 \\ sin\left(noisexn\left(0.083,-1.4\right)\right) \end{cases}$$

$$f_1 = \begin{cases} y+1 \\ sin\left(noisexn\left(0.083,-1.4\right)\right) \end{cases}$$

$$f_2 = \begin{cases} exp\left(\left(\left(noisex\left(\theta,\rho\right)\right)0.57\right)^2\right) - 1\right) \\ noisex\left(-0.0018, log\left(noisex\left(\rho,-0.27\right)\right)\right) \end{cases}$$

$$f_2 = \begin{cases} sin\left(noisexn\left(0.083,-1.4\right)\right) \\ \left(\left(\left(\left(-8.5-\rho\right)^4\right)+0.11\right)^{0.5}\right)+1.7 \end{cases}$$





TILLON_VERT_ET_SPIRALE-1.05 *Cayla, l'atelier des fractales*

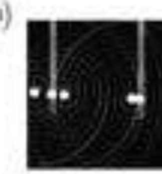
$$= \begin{cases} noisex\left(\rho-0.74, 0.063\right) \\ noisex\left(\left(\left(\left(\rho-0.5\right)^2\right)+\left(\left(\theta-0.5\right)^2\right)\right)^{0.5}, 0.063\right) \end{cases}$$

$$= \begin{cases} noisex\left(\left(2.2\left(cos\left(10\left(root\left(\left(\left(x-0.037\right)^2\right)+\left(\left(y-0.39\right)^2\right)\right)\right)\right)\right)\right)+0.037, 0.063\right) \\ noisex\left(root\left(\left(\left(x+1\right)^2\right)+\left(y^2\right)\right), 0.063\right) \end{cases}$$

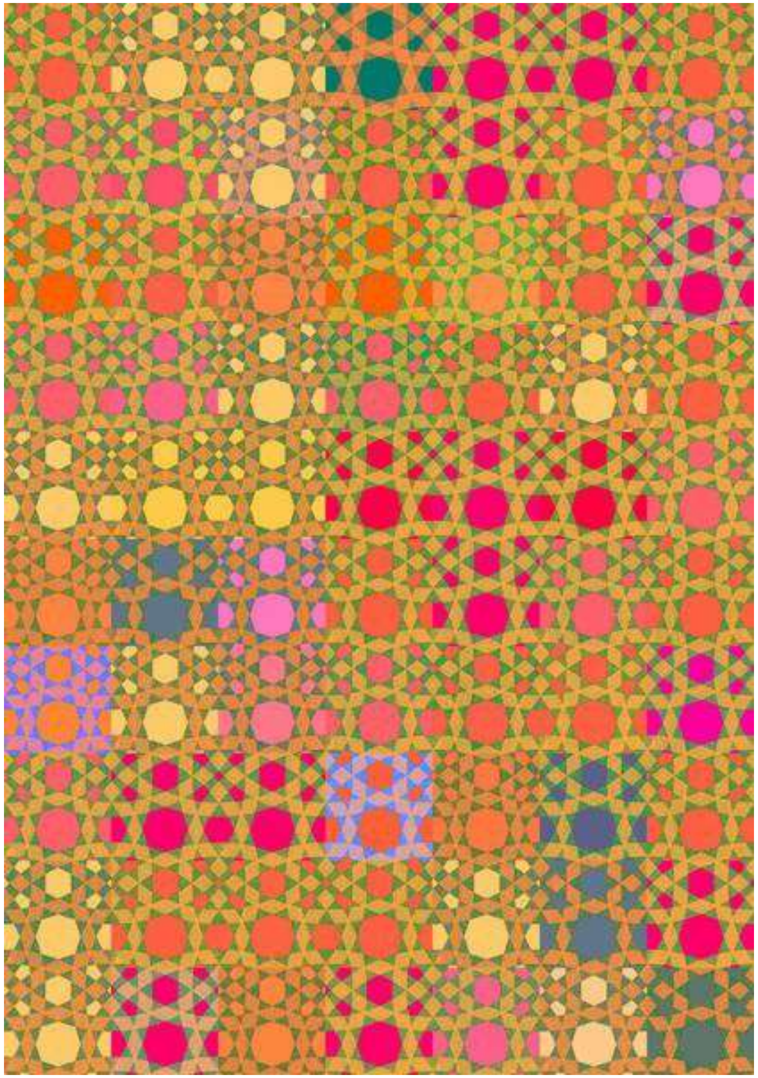
$$= \begin{cases} noisex\left(\rho-0.74, 0.063\right) \\ noisex\left(\left(\left(\left(\rho-0.5\right)^2\right)+\left(\left(\theta-0.5\right)^2\right)\right)^{0.5}, 0.063\right) \end{cases}$$

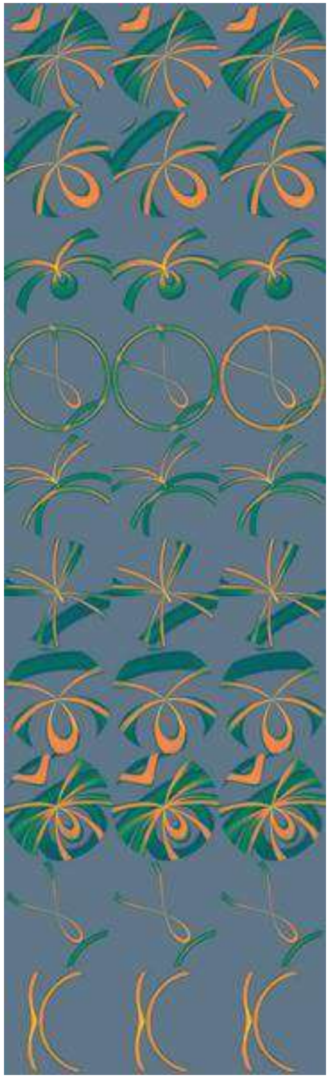
$$= \begin{cases} noisex\left(\rho-0.74, 0.063\right) \\ noisex\left(\left(\left(\left(\rho-0.5\right)^2\right)+\left(\left(\theta-0.5\right)^2\right)\right)^{0.5}, 0.063\right) \end{cases}$$

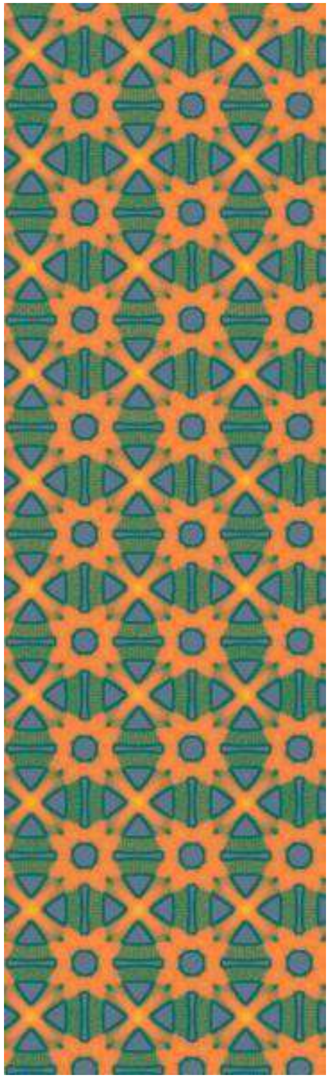
$$= \begin{cases} noisex\left(\left(2.2\left(cos\left(10\left(root\left(\left(\left(x-0.037\right)^2\right)+\left(\left(y-0.39\right)^2\right)\right)\right)\right)\right)\right)+0.037, 0.063\right) \\ noisex\left(root\left(\left(\left(x+1\right)^2\right)+\left(y^2\right)\right), 0.063\right) \end{cases}$$



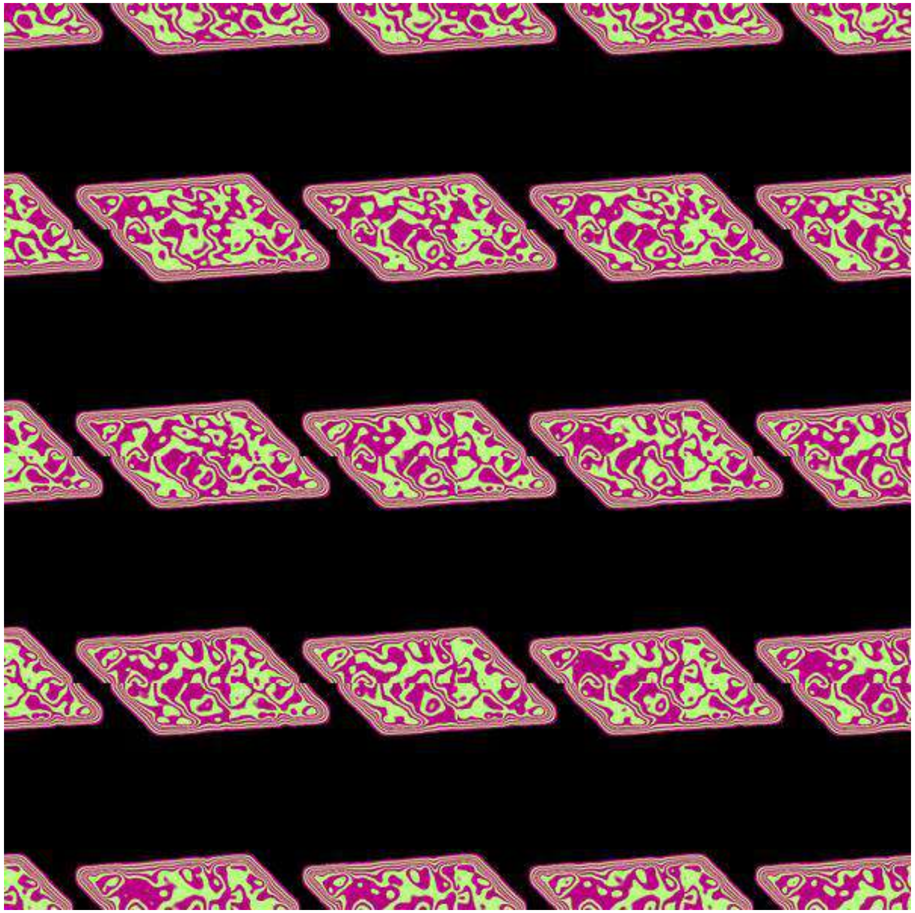
Etudes de textures, Emmanuel Cayla

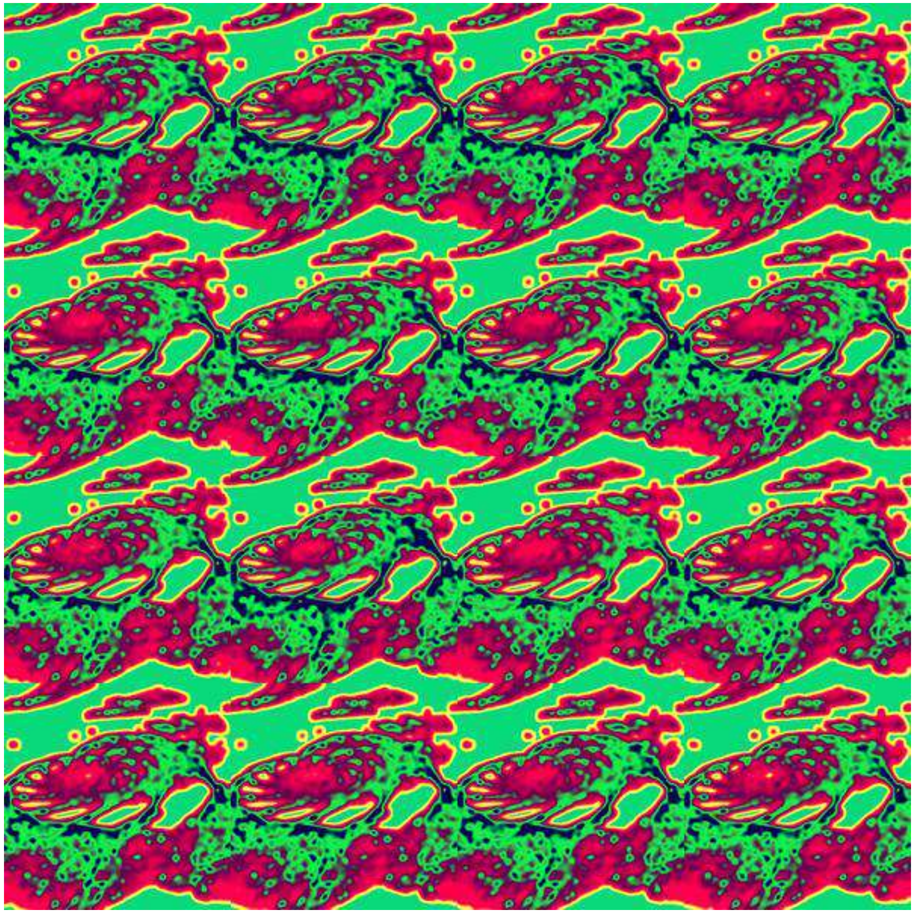






Etudes textile, Marie-Amélie Porcher





Cétoine
inventeurs d'images

Collection prototype, Emmanuel Cayla

MERCI
de votre attention