



Métaheuristiques

Algorithmes itératifs (et stochastiques) d'optimisation globale



THALES

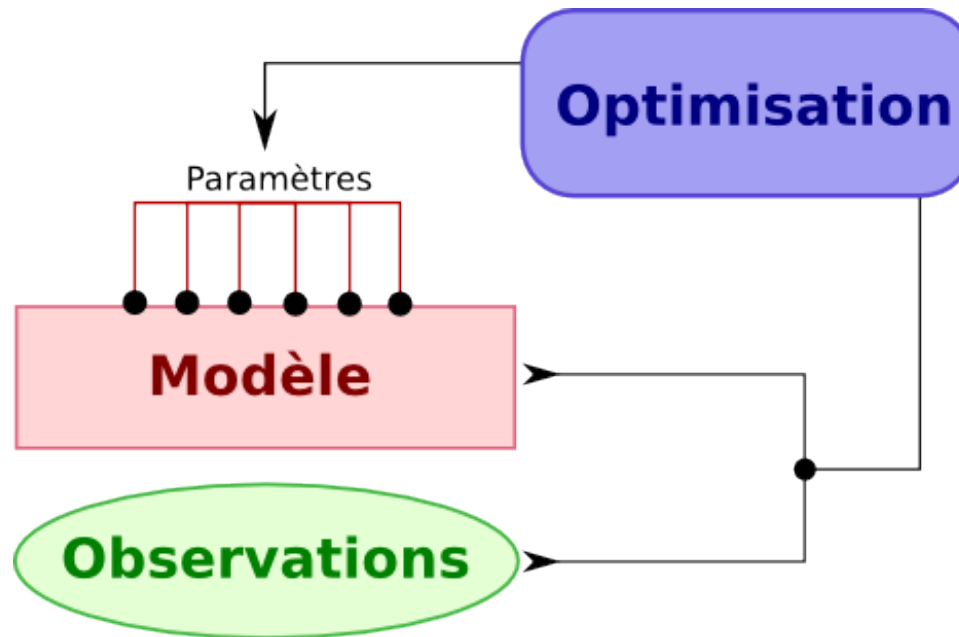


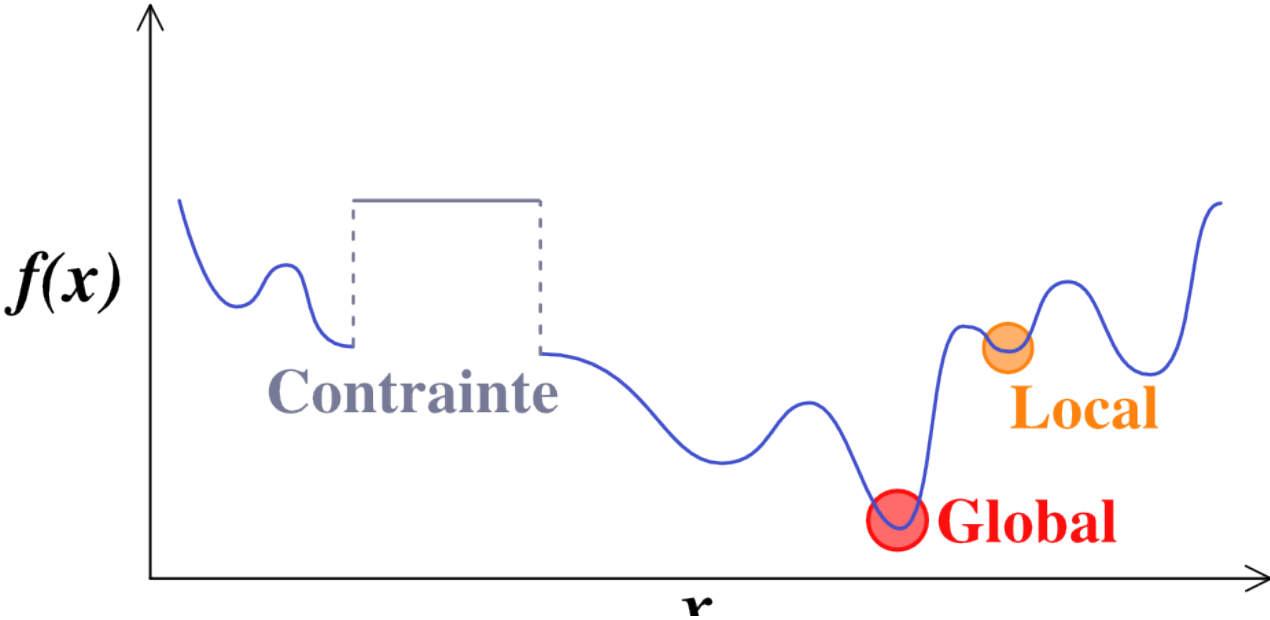
Johann Dréo

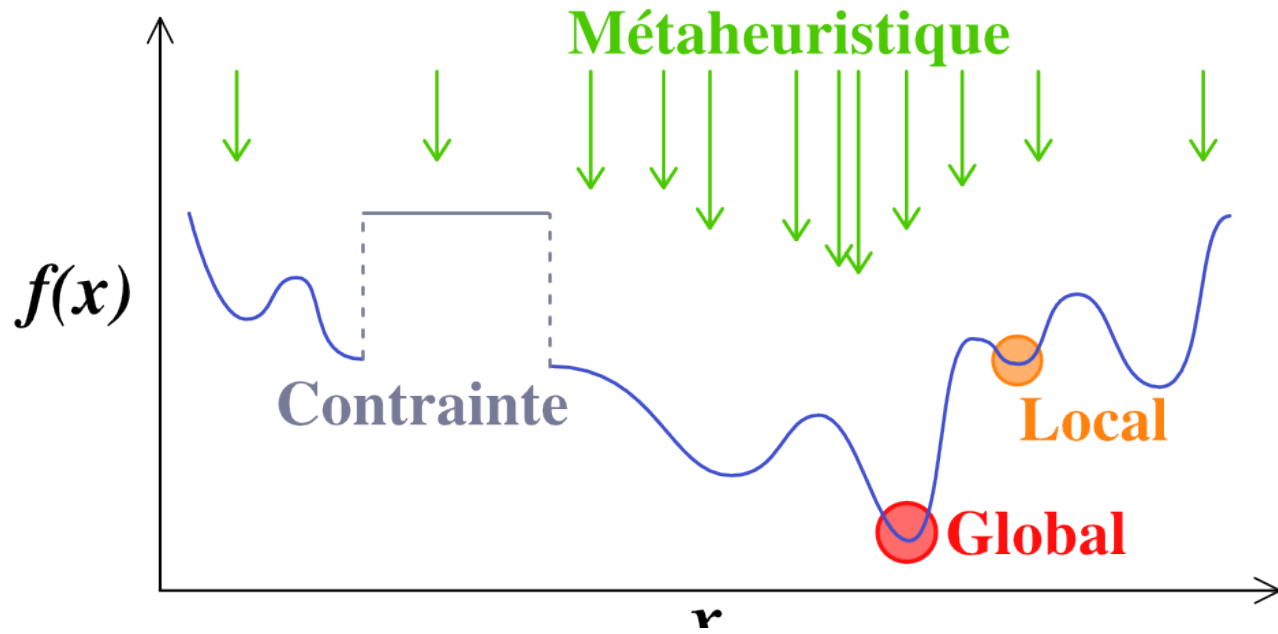
- ▶ THALES Research & Technology
 - ▶ Laboratoire mathématiques & techniques de la décision
 - Équipe optimisation

Yann Collette

- ▶ Renault









Problèmes

- ▶ Exemples
- ▶ Caractéristiques

Métaheuristiques

- ▶ Exemples
- ▶ Synthèse

Applications

- ▶ Validation
- ▶ Exemples



Problèmes

- ▶ Exemples
 - ▶ Recherche opérationnelle
 - ▶ Ingénierie
 - ▶ Intelligence artificielle
- ▶ Caractéristiques

Métaheuristiques

- ▶ Exemples
- ▶ Synthèse

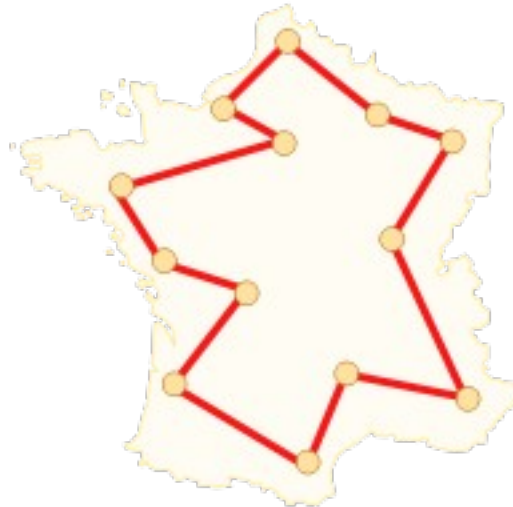
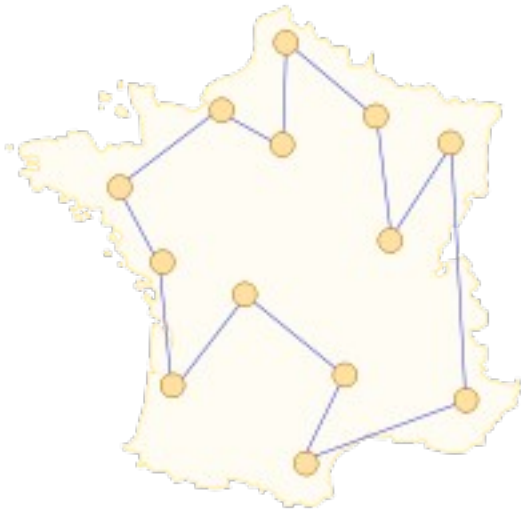
Applications

- ▶ Validation
- ▶ Exemples



Voyageur de commerce

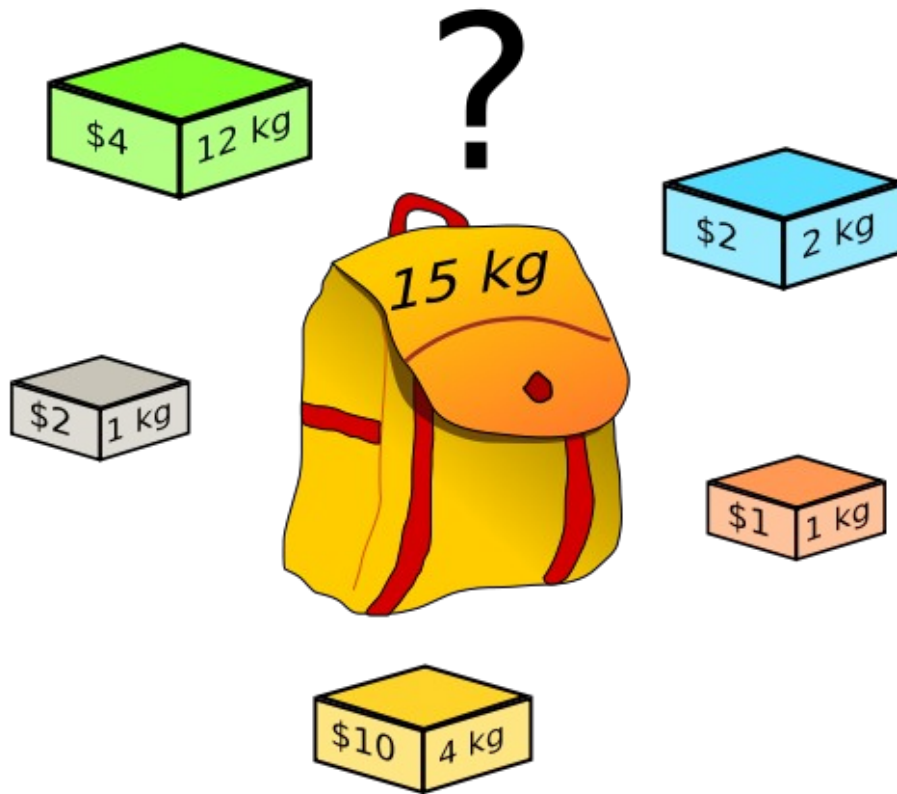
- ▶ Minimiser la longueur du trajet





Sac à dos

- ▶ Maximiser l'intérêt de la sélection d'objets



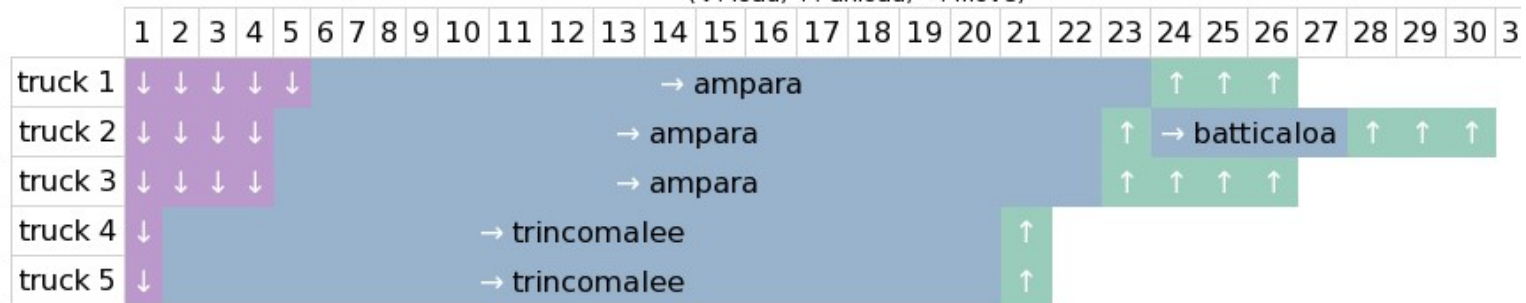
Planification temporelle

- Déterminer un emploi du temps

Running planner ... please, wait ... done.

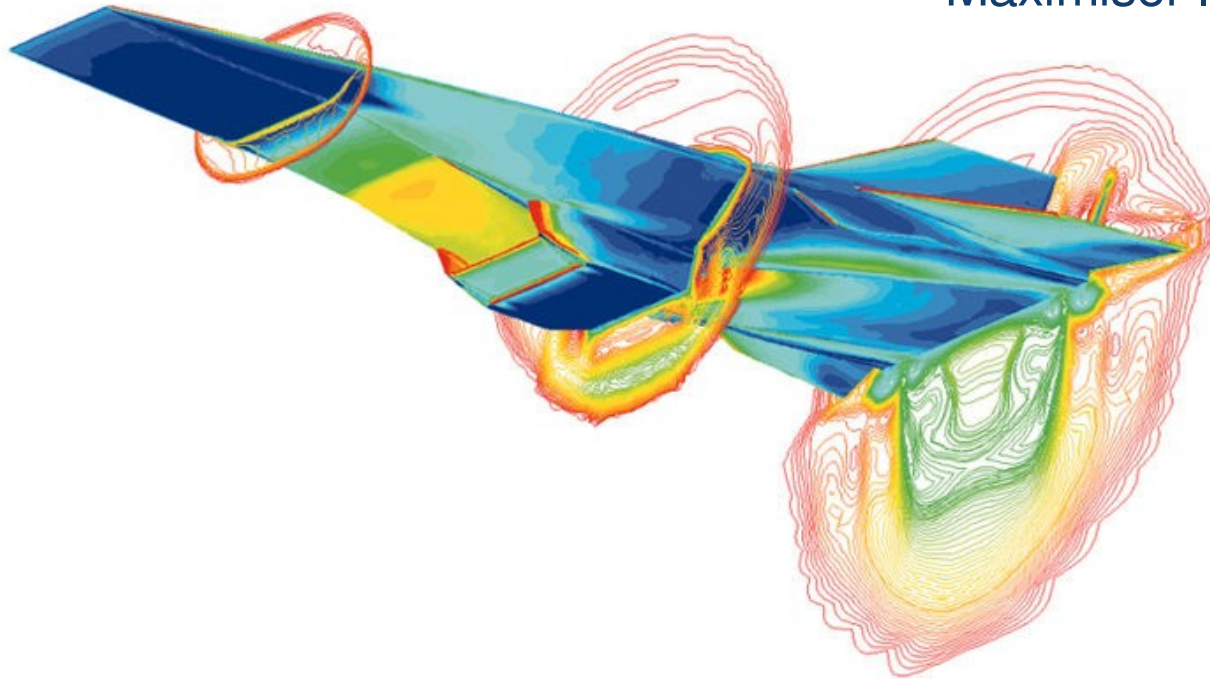
Resource allocation planning, total duration: 36

(↓ : load, ↑ : unload, → : move)



Aérodynamique

- ▶ Maximiser la portance



Dryden Flight Research Center ED97 43968-01

HYPER-X AT MACH 7: This computational fluid dynamic (CFD) image is of the Hyper-X vehicle at the Mach 7 test condition with the engine operating.

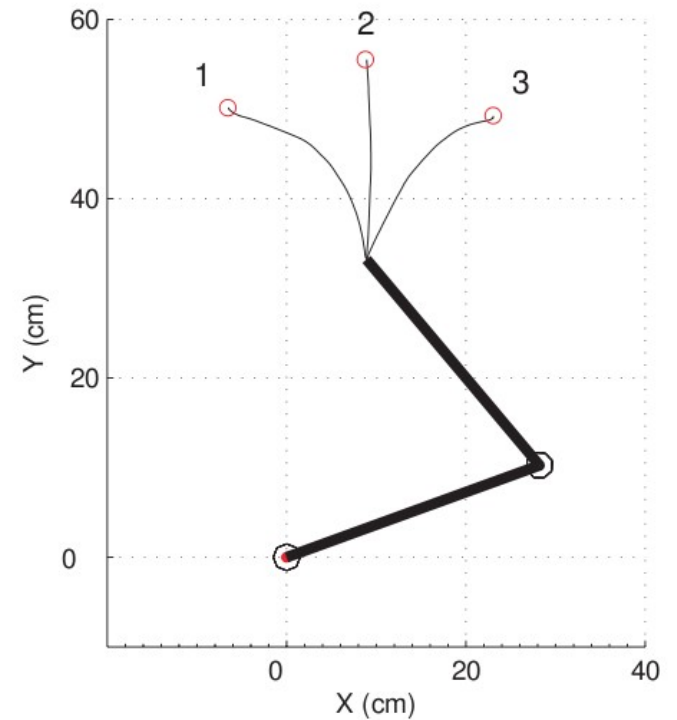
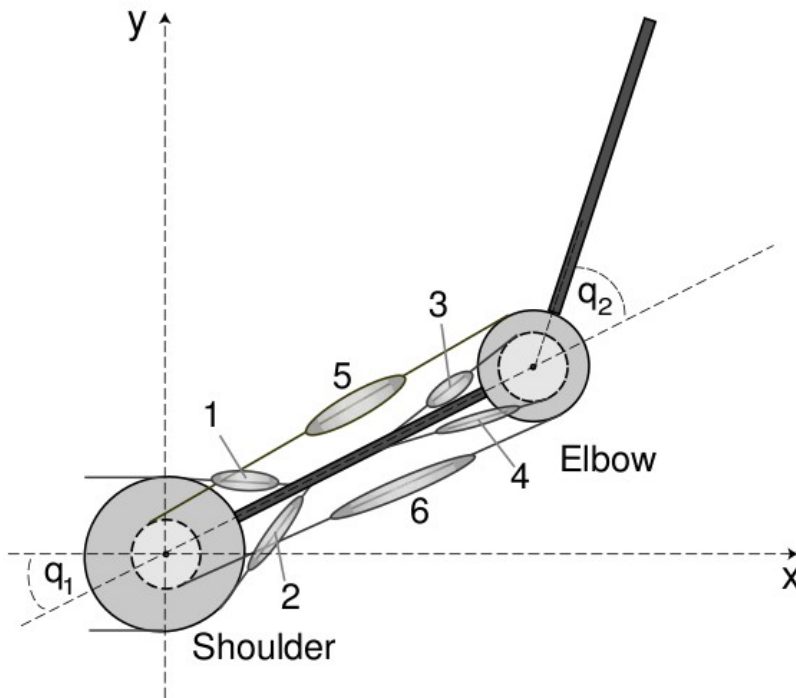


THALES



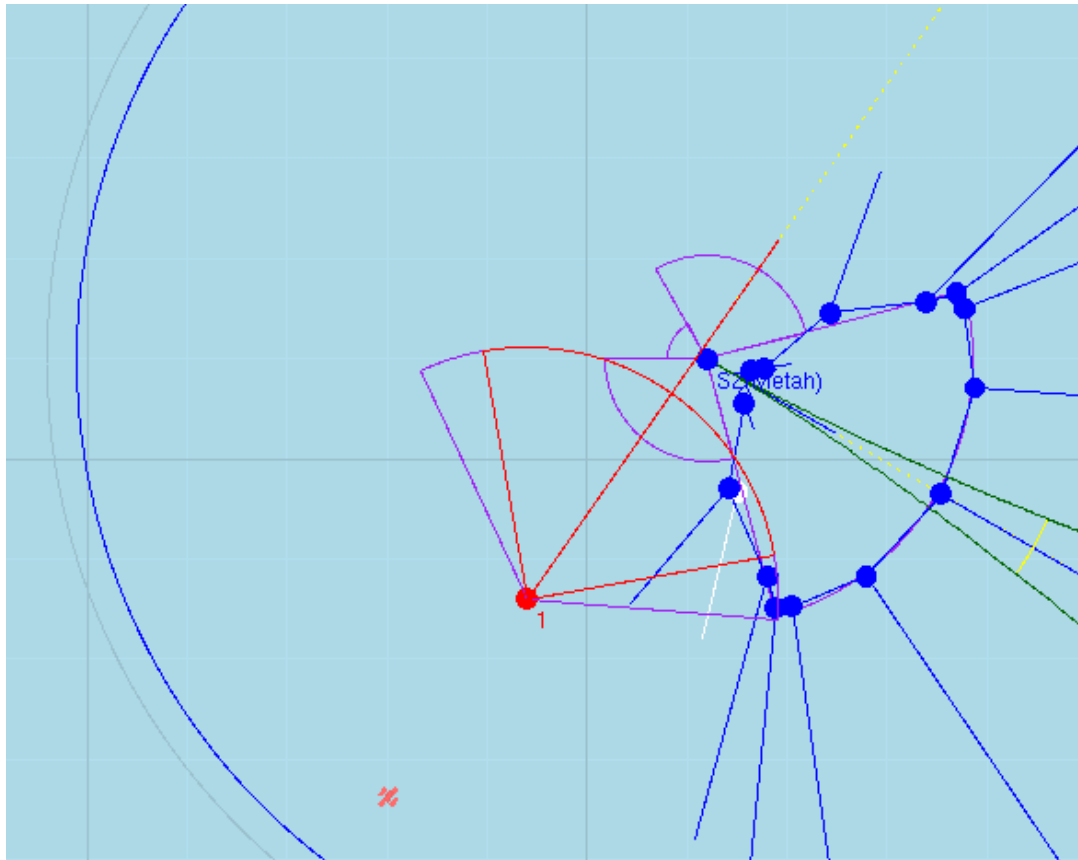
Déplacement de bras robot

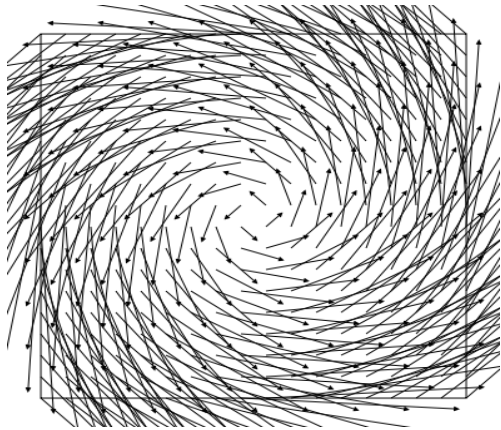
- Minimiser l'énergie dépensée



IA jeu vidéo

- ▶ Maximiser le gain





Optimisation continue

- ▶ Variables réelles

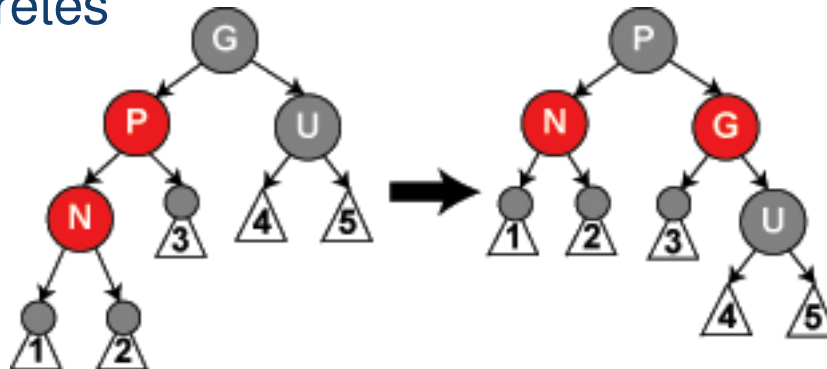
Voisinage

- ▶ Implicite
 - ▶ Fonctions

Optimisation combinatoire

- ▶ Variables discrètes

- ▶ Explicite
 - ▶ Permutation



http://commons.wikimedia.org/wiki/Image:Red-black_tree_insert_case_5.png



Trouver le problème



Observation



Modèle



But

Modélisation

- ▶ Expression mathématique
- ▶ Contraintes
- ▶ Qualification des solutions
- ▶ Objectifs contradictoires



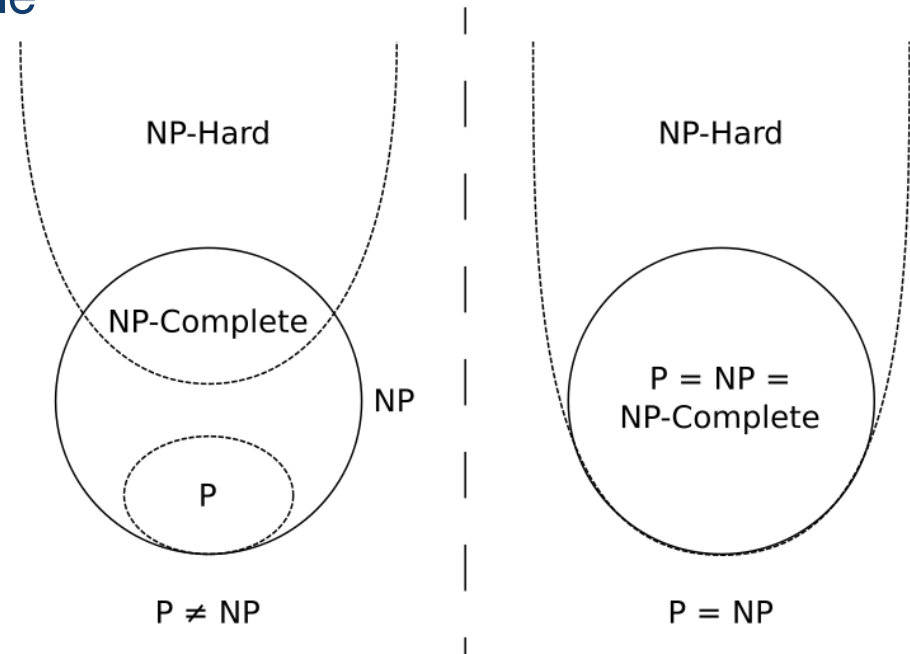
Résolution

- ▶ Taille de l'espace des solutions
 - ▶ NP-...
 - ▶ Nombres de variables
- ▶ Complexité
 - ▶ Irrégularités
 - ▶ Différentiabilité
 - ▶ Continuité
- ▶ Temps de calcul
 - ▶ Fonction objectif par simulation
- ▶ Conception/Production



NP

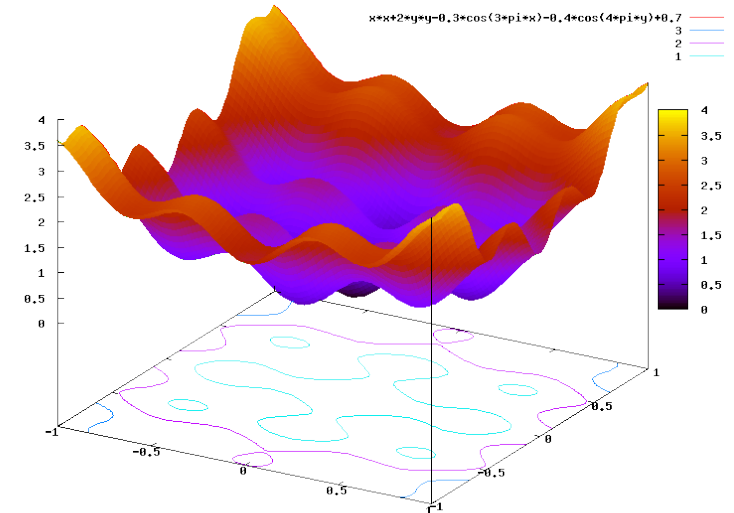
- ▶ Algorithme **N**on déterministe
- ▶ Résolution **P**olynomialie
- ▶ Oracle
- ▶ Estimation valeur polynomiale



http://commons.wikimedia.org/wiki/Image:P_np_np-complete_np-hard.svg

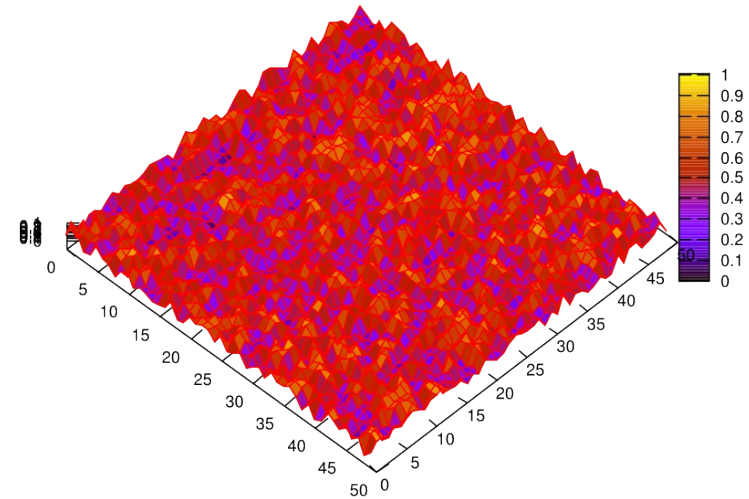
Problème structuré

- ▶ Voisinage
- ▶ Structure générale cohérente



Problème impossible :

- ▶ Fonction aléatoire





Problèmes

- ▶ Exemples
- ▶ Caractéristiques

Applications

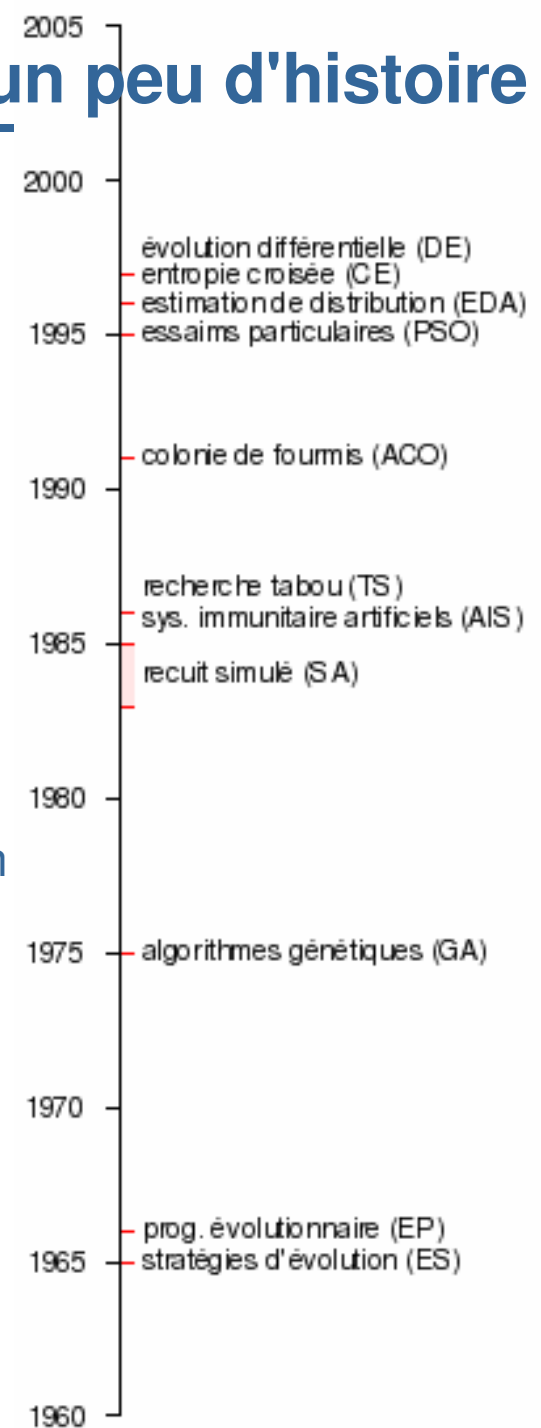
- ▶ Validation
- ▶ Exemples

Métaheuristiques

- ▶ Exemples
 - ▶ Recherche tabou
 - ▶ Algorithmes évolutionnistes
 - ▶ Recuit simulé
 - ▶ Colonies de fourmis
 - ▶ Estimation de distribution
- ▶ Synthèse

Métaheuristiques : un peu d'histoire

- ▶ 1954 : Barricelli, simulation évolution, optimisation
- ▶ 1965 : Rechenberg, stratégies d'évolution
- ▶ 1986 : terme méta-heuristique, Fred Glover :
 - ▶ « La recherche avec tabou peut être vue comme une "méta-heuristique", **superposée** à une autre heuristique. L'approche vise à éviter les **optimums locaux** par une stratégie d'interdiction (ou, plus généralement, de pénalisation) de certains mouvements. »
- ▶ 1988 : conférence, algorithmes génétiques
- ▶ 1989 : premier logiciel A.G. Commercial
- ▶ 1996 : algorithmes à estimation de distribution





Fonction objectif

- ▶ Minimisation
- ▶ Mono-objectif

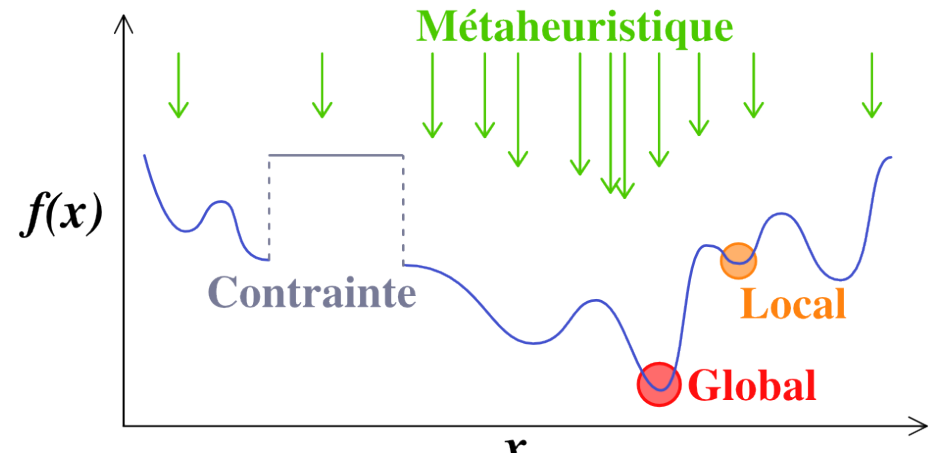
Solution

- ▶ Représentation
- ▶ Valeur

Échantillon

- ▶ Ensemble de solution

Voisinage



Stochastique

- ▶ Probabiliste + temps

Itératif

- ▶ Critère d'arrêt



Problèmes

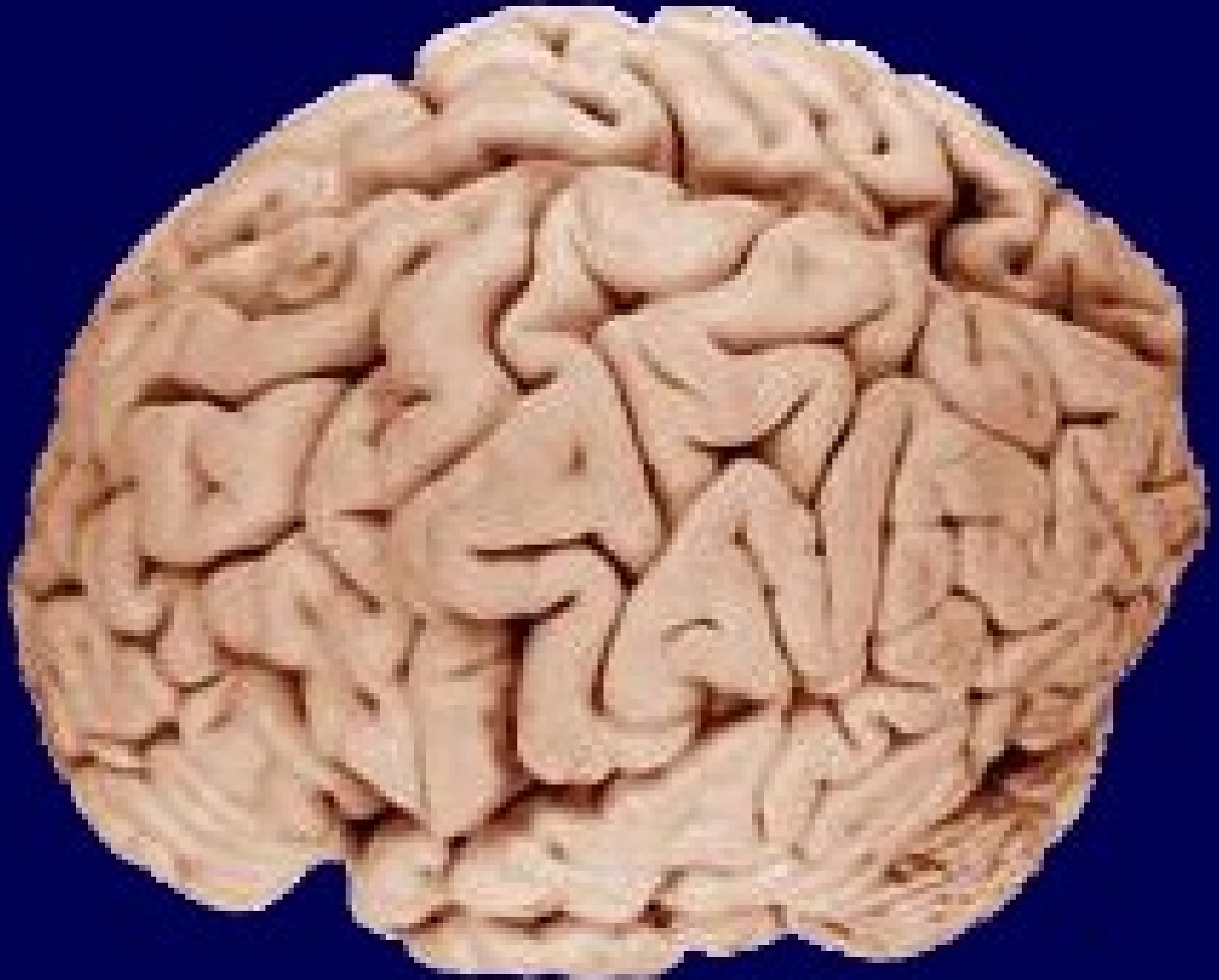
- ▶ Exemples
- ▶ Caractéristiques

Applications

- ▶ Validation
- ▶ Exemples

Métaheuristiques

- ▶ Exemples
 - ▶ Recherche tabou
 - ▶ Algorithmes évolutionnistes
 - ▶ Recuit simulé
 - ▶ Colonies de fourmis
 - ▶ Estimation de distribution
- ▶ Synthèse





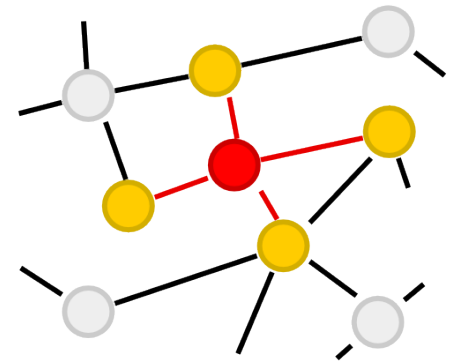
Recherche « tabou » ou « avec tabous »

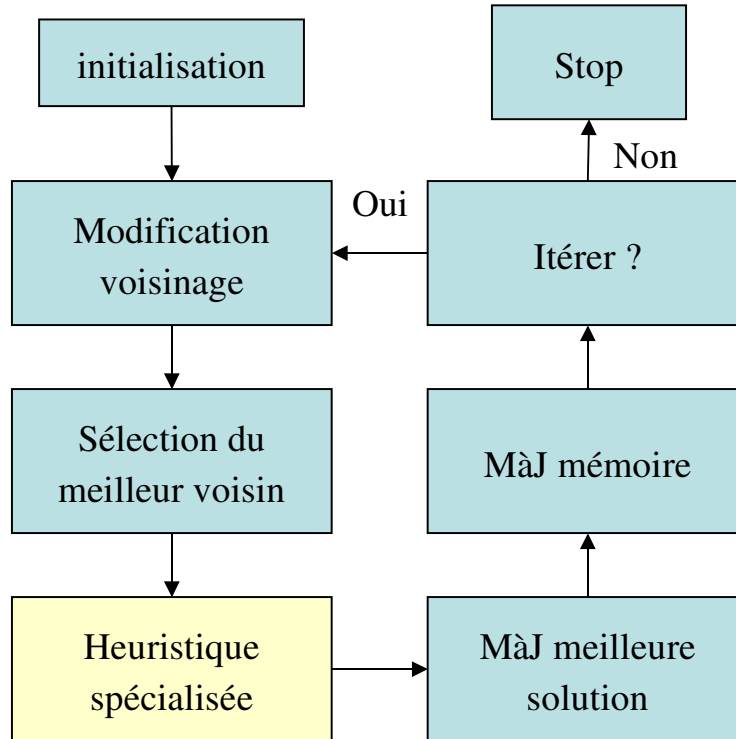
Liste « tabou » de mouvements interdits

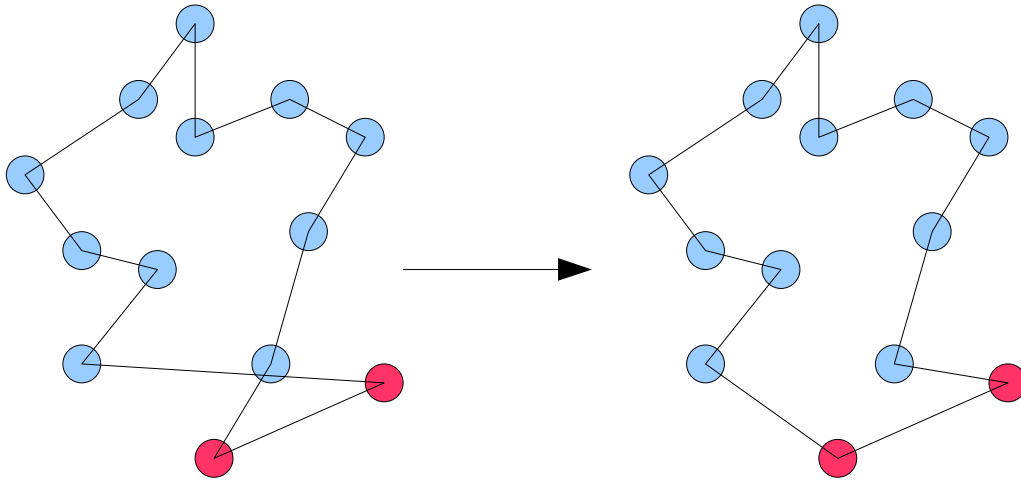
- ▶ Minimums locaux

Voisinage

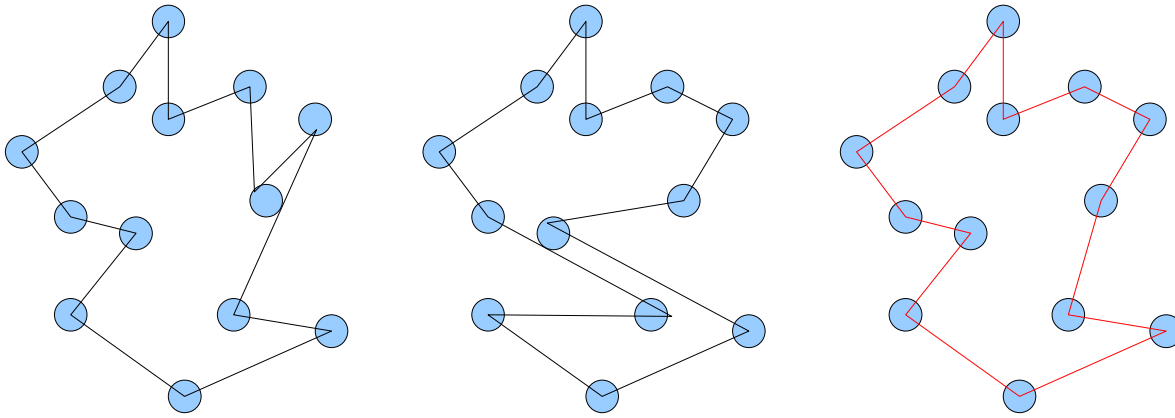
- ▶ Initialisation
- ▶ Modification de solution existante



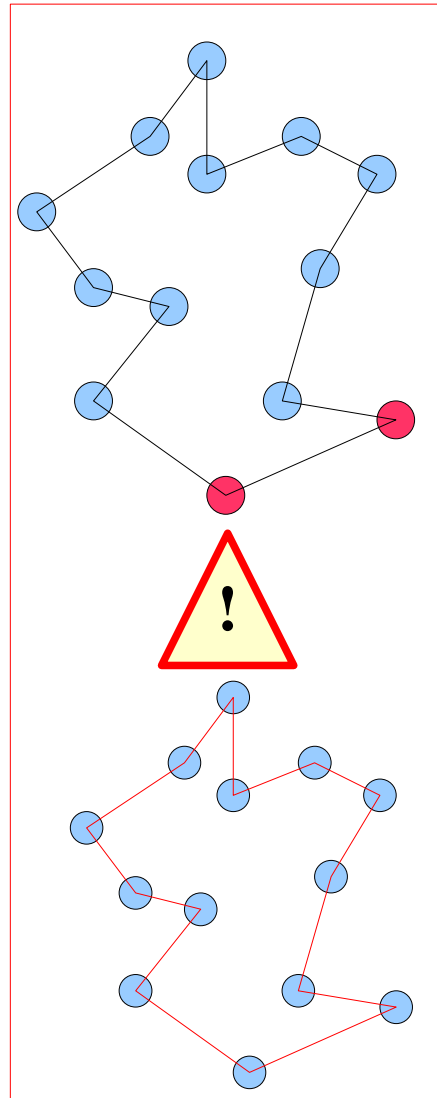
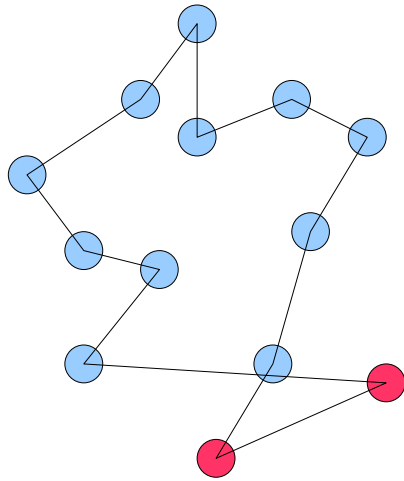




Voisinage



Liste tabou



Voisinage

Liste tabou



Problèmes

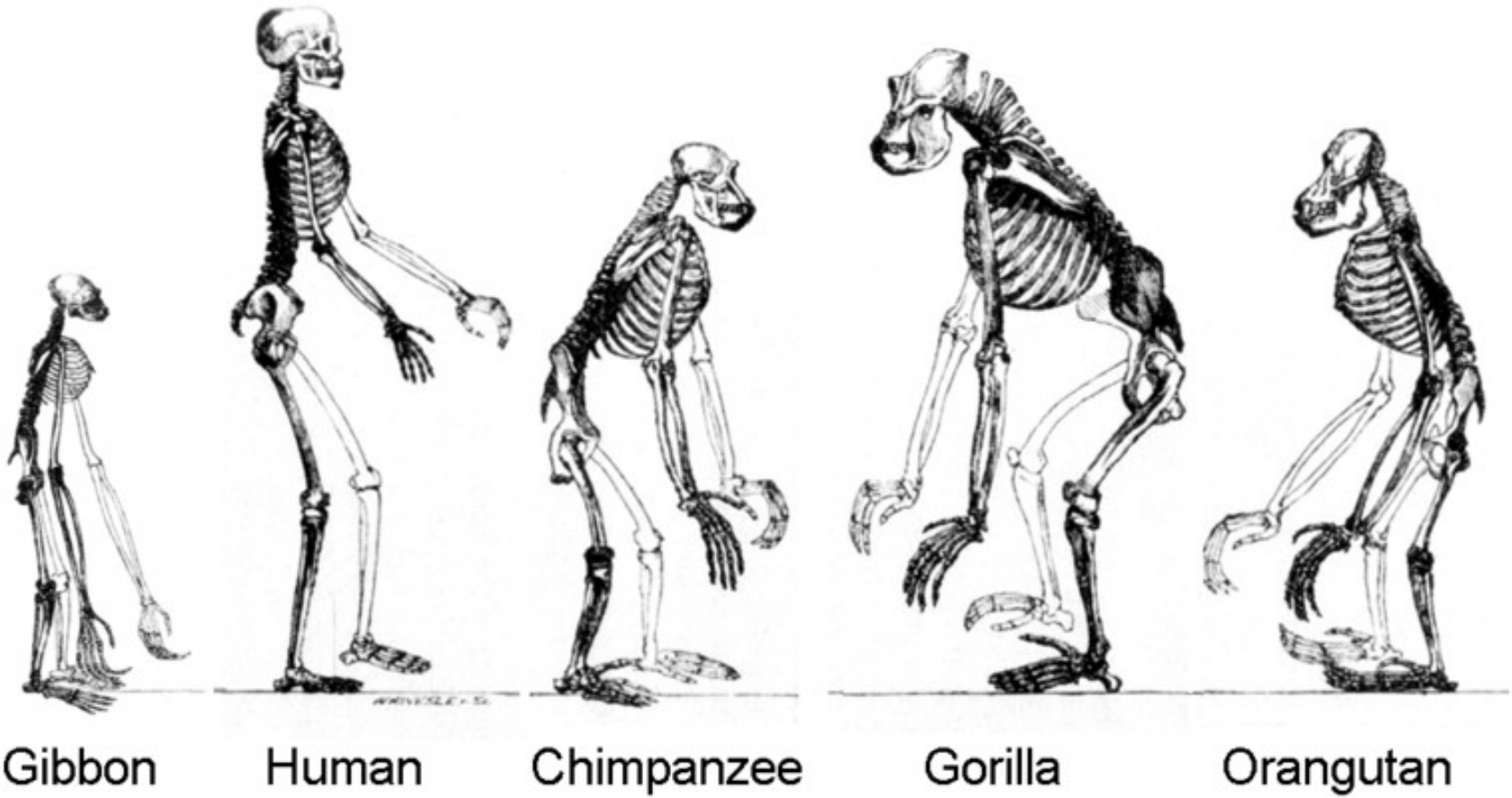
- ▶ Exemples
- ▶ Caractéristiques

Applications

- ▶ Validation
- ▶ Exemples

Métaheuristiques

- ▶ Exemples
 - ▶ Recherche tabou
 - ▶ Algorithmes évolutionnistes
 - ▶ Recuit simulé
 - ▶ Colonies de fourmis
 - ▶ Estimation de distribution
- ▶ Synthèse





1965 : stratégies d'évolution

1966 : programmation évolutionnaire

1975 : algorithmes génétiques

1980 : programmation génétique

1986 : systèmes immunitaire artificiels



Métaheuristiques les plus connus

Différences historiques entre algorithmes

Concepts équivalents

- ▶ Algorithmes génétiques
- ▶ Algorithmes évolutionnaires
- ▶ Algorithmes évolutionnistes

Algorithmes stochastiques

Enchaînement d'opérateurs



Solution	= individu
Échantillon	= population
Valeur	= fitness
Représentation	= codage



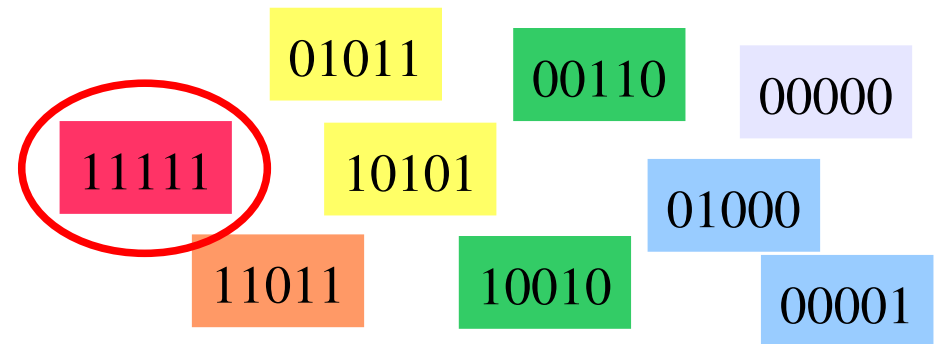


« Opérateurs »

- ▶ Sélection
- ▶ Croisement
- ▶ Mutation
- ▶ Évaluation
- ▶ Remplacement

One Max

$$\sum_{i=0}^4 x_i$$





« Opérateurs »

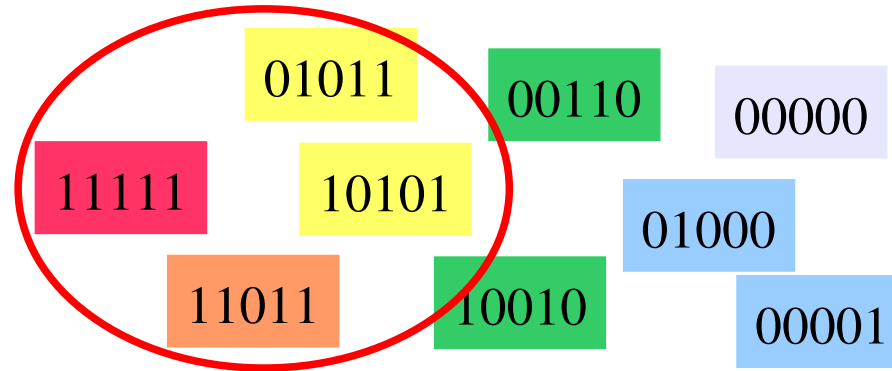
▶ *Sélection*

▶ Croisement

▶ Mutation

▶ Évaluation

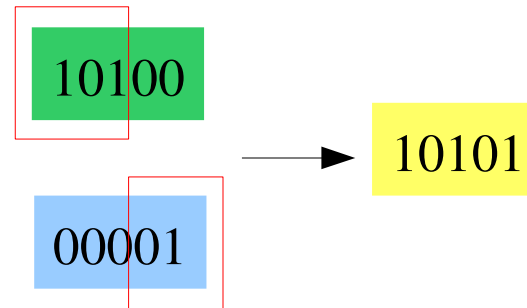
▶ Remplacement





« Opérateurs »

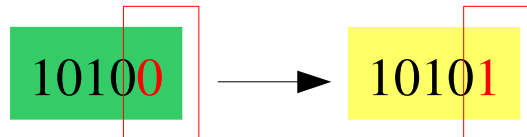
- ▶ Sélection
- ▶ ***Croisement***
- ▶ Mutation
- ▶ Évaluation
- ▶ Remplacement





« Opérateurs »

- ▶ Sélection
- ▶ Croisement
- ▶ ***Mutation***
- ▶ Évaluation
- ▶ Remplacement





« Opérateurs »

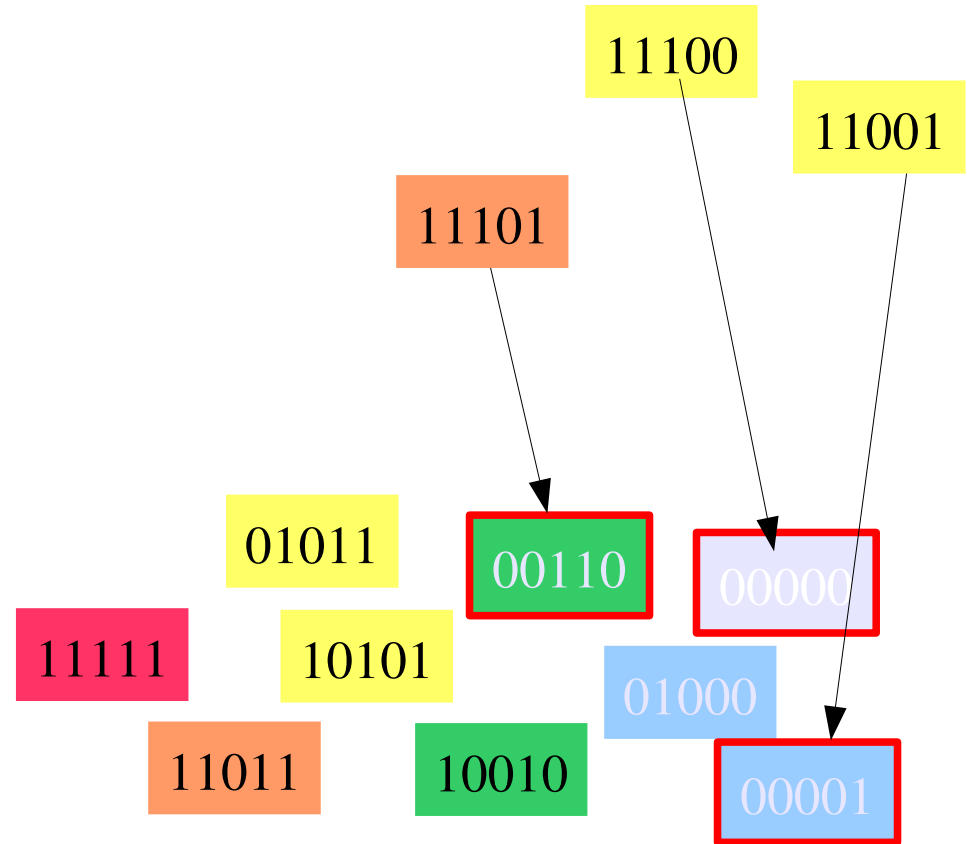
- ▶ Sélection
- ▶ Croisement
- ▶ Mutation
- ▶ ***Évaluation***
- ▶ Remplacement

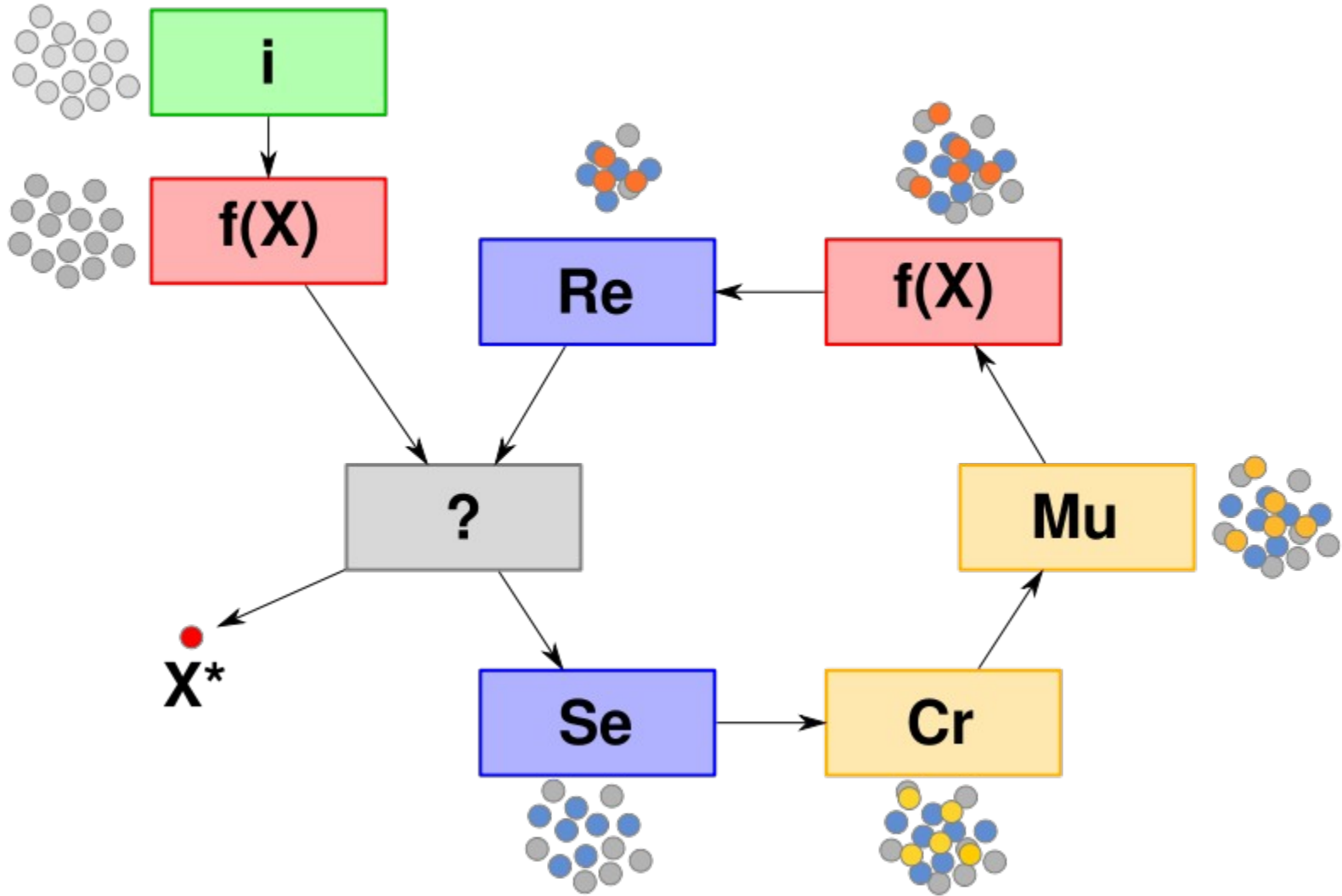
10101 → 10101



« Opérateurs »

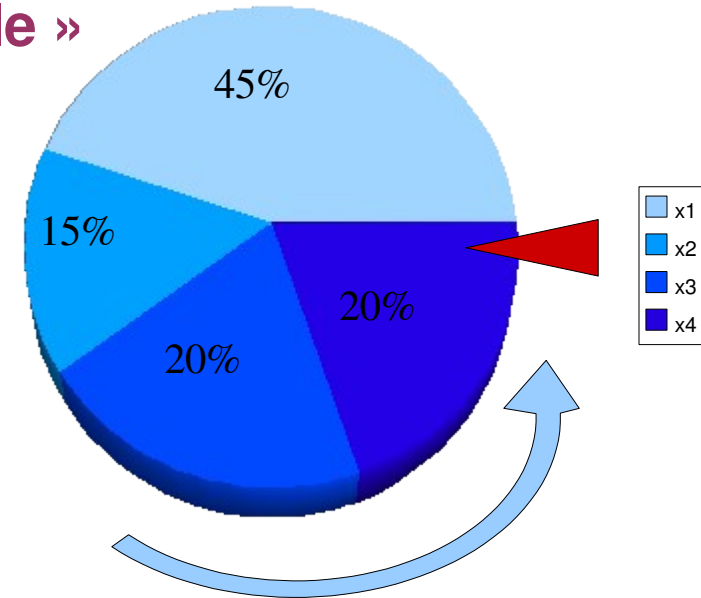
- ▶ Sélection
- ▶ Croisement
- ▶ Mutation
- ▶ Évaluation
- ▶ *Remplacement*





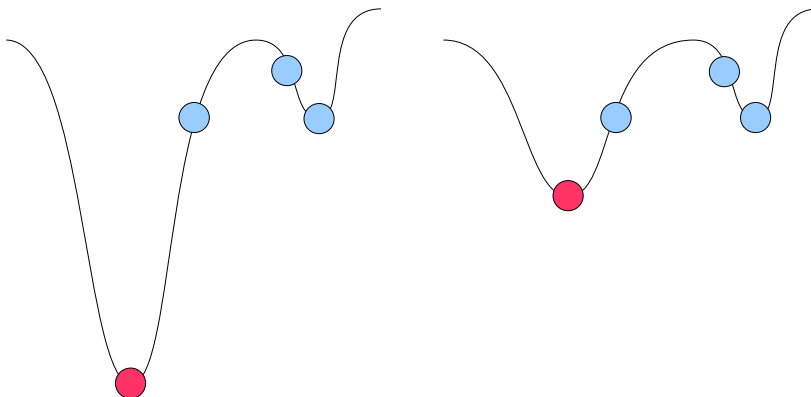
Sélection par « roulette proportionnelle »

- ▶ surface proportionnelle à sa valeur



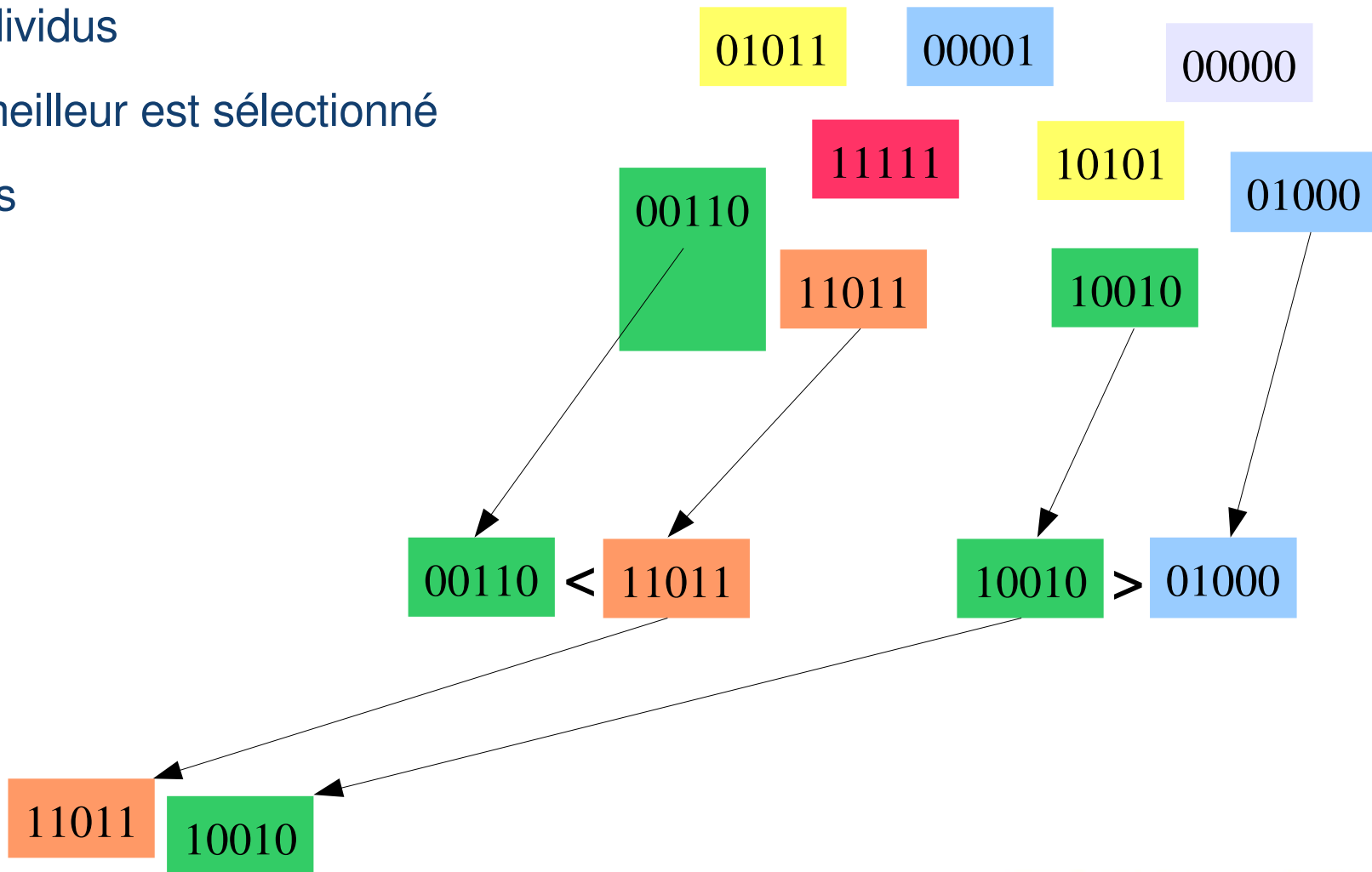
Problème

- ▶ Biais selon échelle de la fonction objectif



Sélection par « tournoi stochastique »

- ▶ k individus
- ▶ Le meilleur est sélectionné
- ▶ n fois

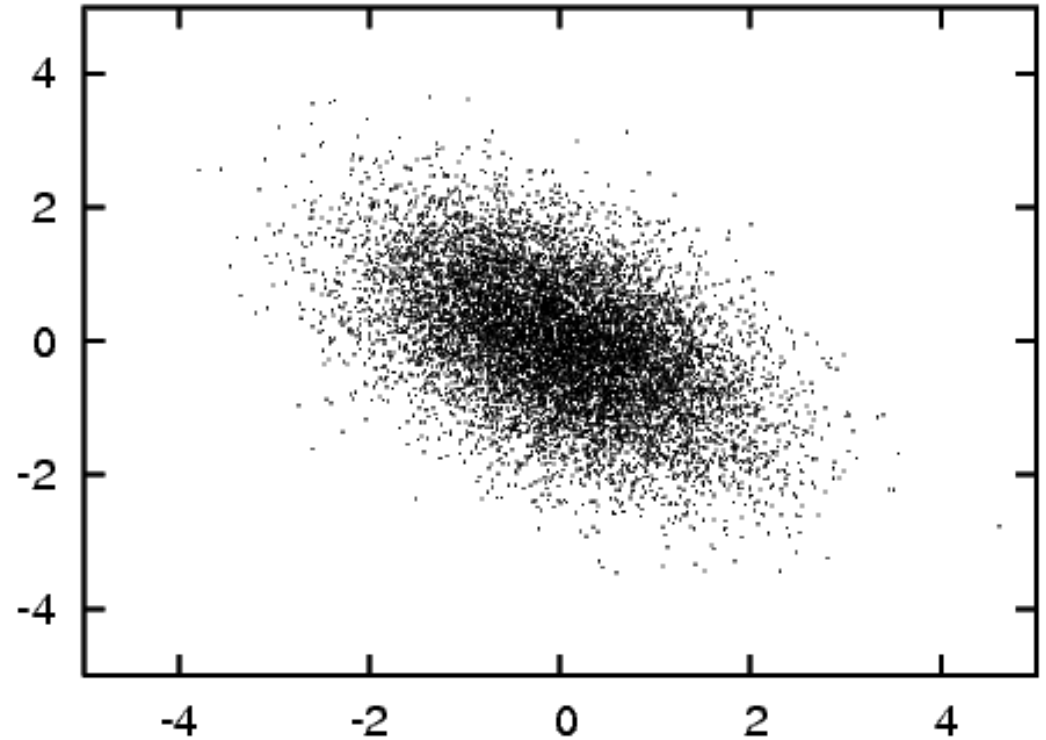




Mutation gaussienne

► Pour tout x :

► $x_i + \mathcal{N}_{m,s}^i$



Simulated Binary Crossover

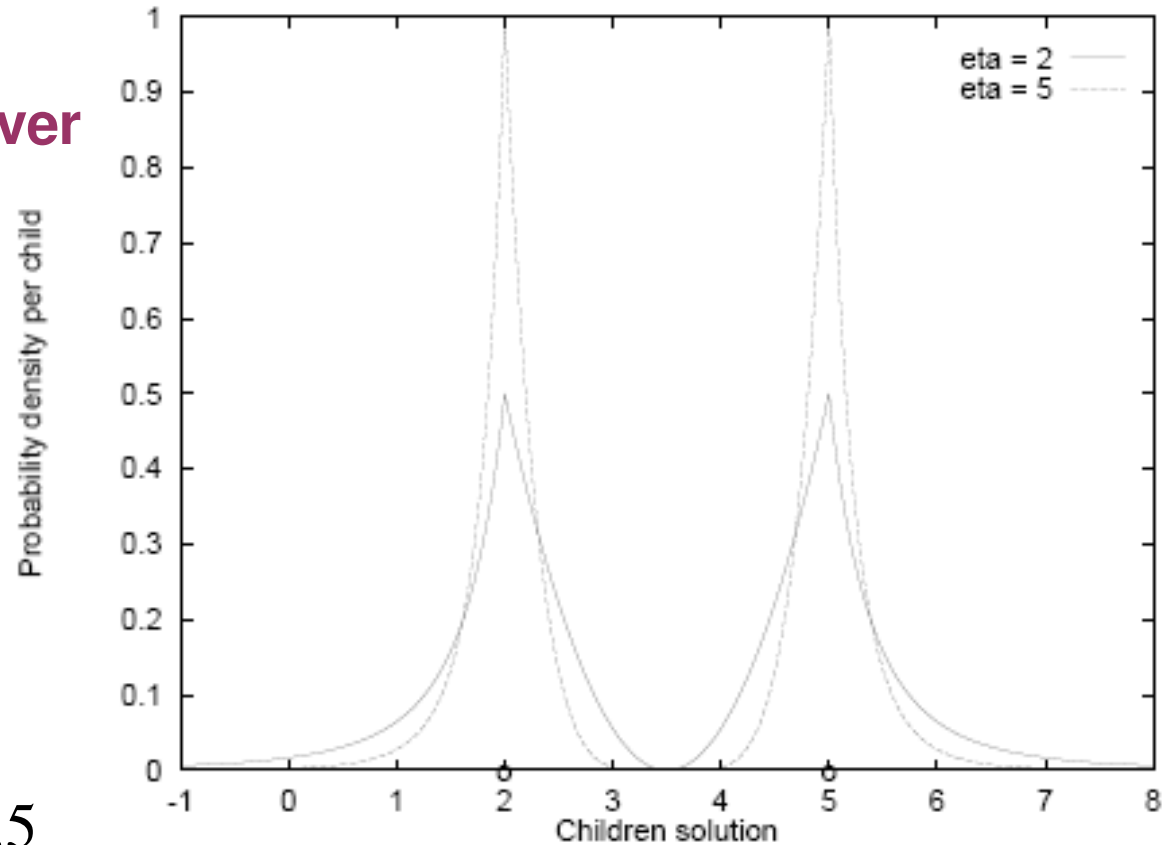
$$u_i \in U_{0,1}$$

$$b_i = (2 \cdot u_i)^{1/(1+\eta)} \text{ si } u_i \leq 0.5$$

$$= (1/(1-u_i))^{1/(1+\eta)} \text{ sinon}$$

$$2 \cdot x_i^1(t+1) = (1-b_i) \cdot x_i^1(t) + (1+b_i) \cdot x_i^2(t)$$

$$2 \cdot x_i^2(t+1) = (1+b_i) \cdot x_i^1(t) + (1-b_i) \cdot x_i^2(t)$$





Croisement multipoint

11001010

01011100

11 | 0010 | 10

01 | 0111 | 00

01 | 0010 | 00

11 | 0111 | 10

Croisement uniforme

11001010

01011100

01110100 Masque

11001010

01011100

11011110

01001000



Problèmes

- ▶ Exemples
- ▶ Caractéristiques

Applications

- ▶ Validation
- ▶ Exemples

Métaheuristiques

- ▶ Exemples
 - ▶ Recherche tabou
 - ▶ Algorithmes évolutionnistes
 - ▶ Recuit simulé
 - ▶ Colonies de fourmis
 - ▶ Estimation de distribution
- ▶ Synthèse



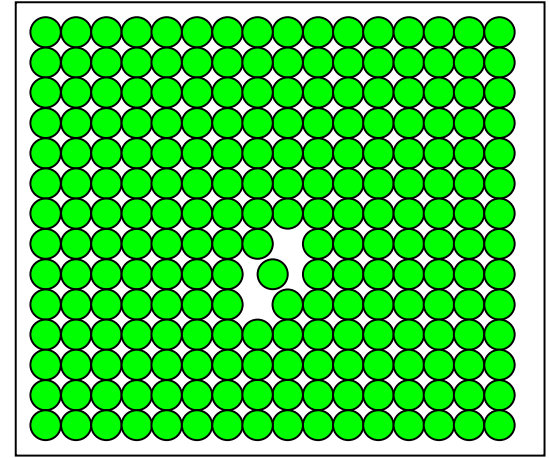
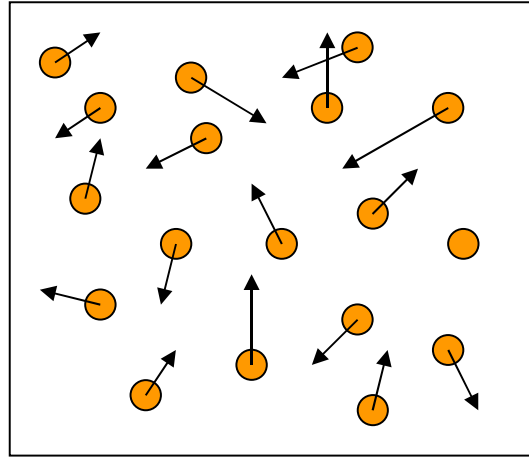
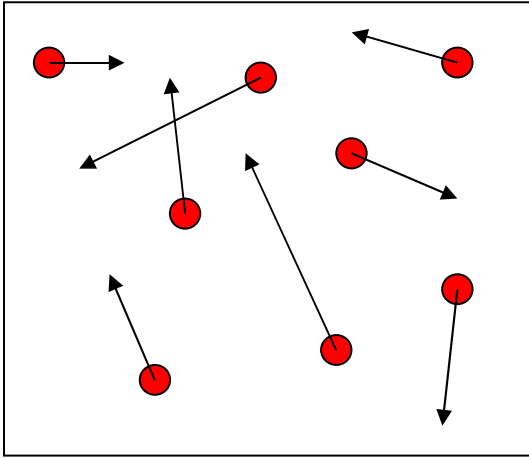
http://commons.wikimedia.org/wiki/Image:Hot_metalwork.jpg

Historique

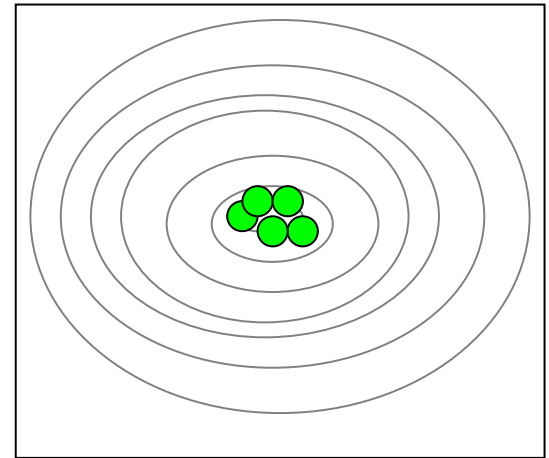
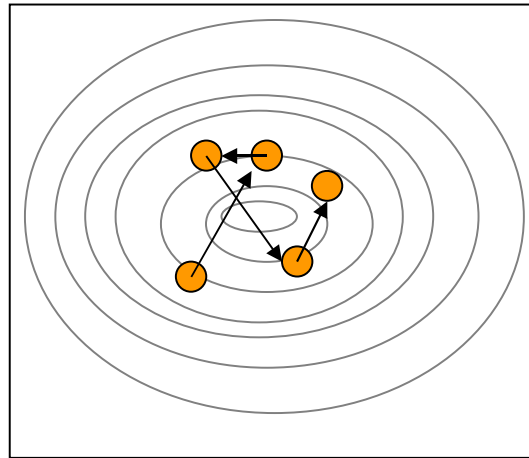
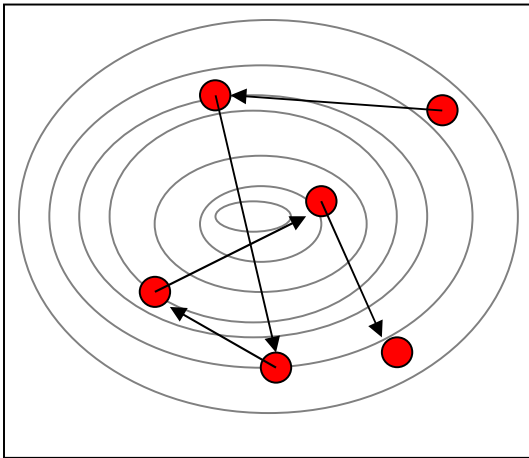
- ▶ 1970 : Hastings propose l'algorithme de Metropolis-Hastings,
- ▶ 1983 : Kirkpatrick, Gelatt et Vecchi conçoivent le recuit simulé
- ▶ 1985 : indépendamment de ceux-ci, Černý propose le même algorithme



Coté physique



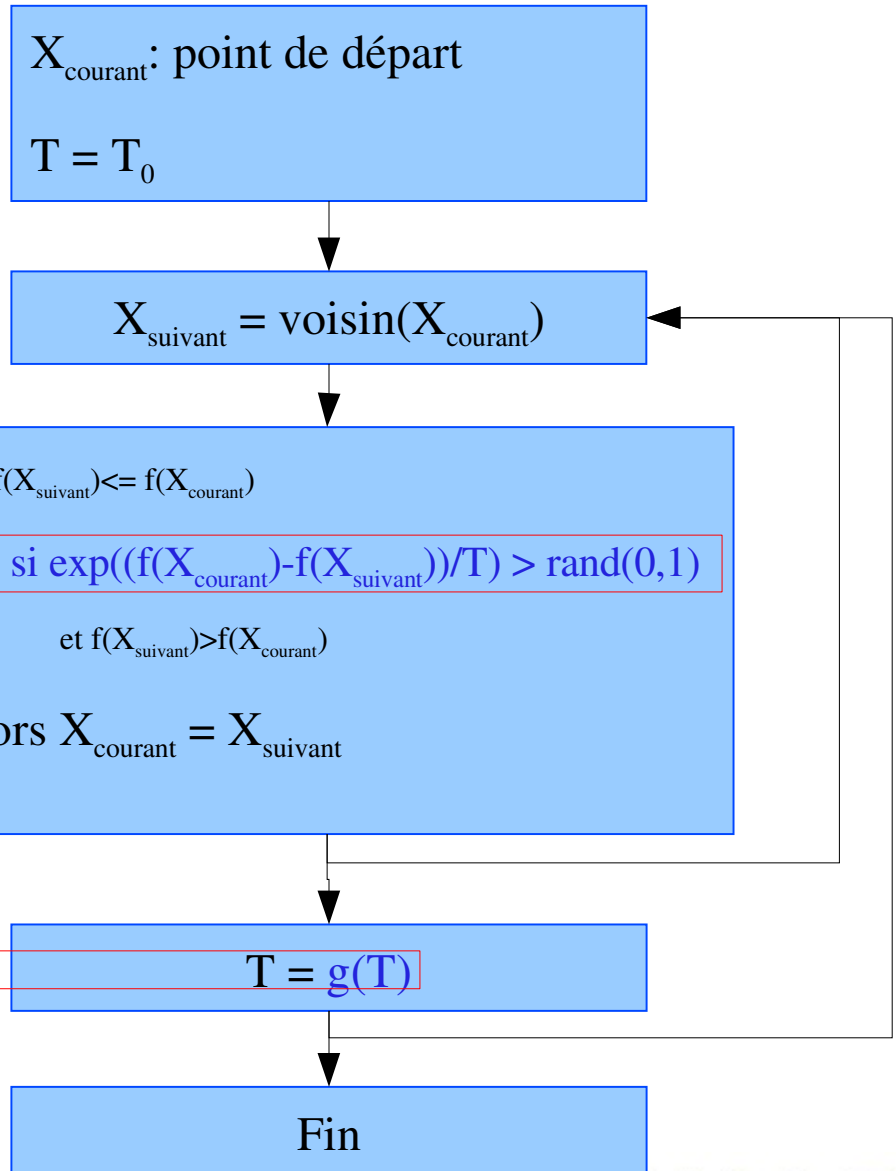
Coté optimisation





Méthode de descente

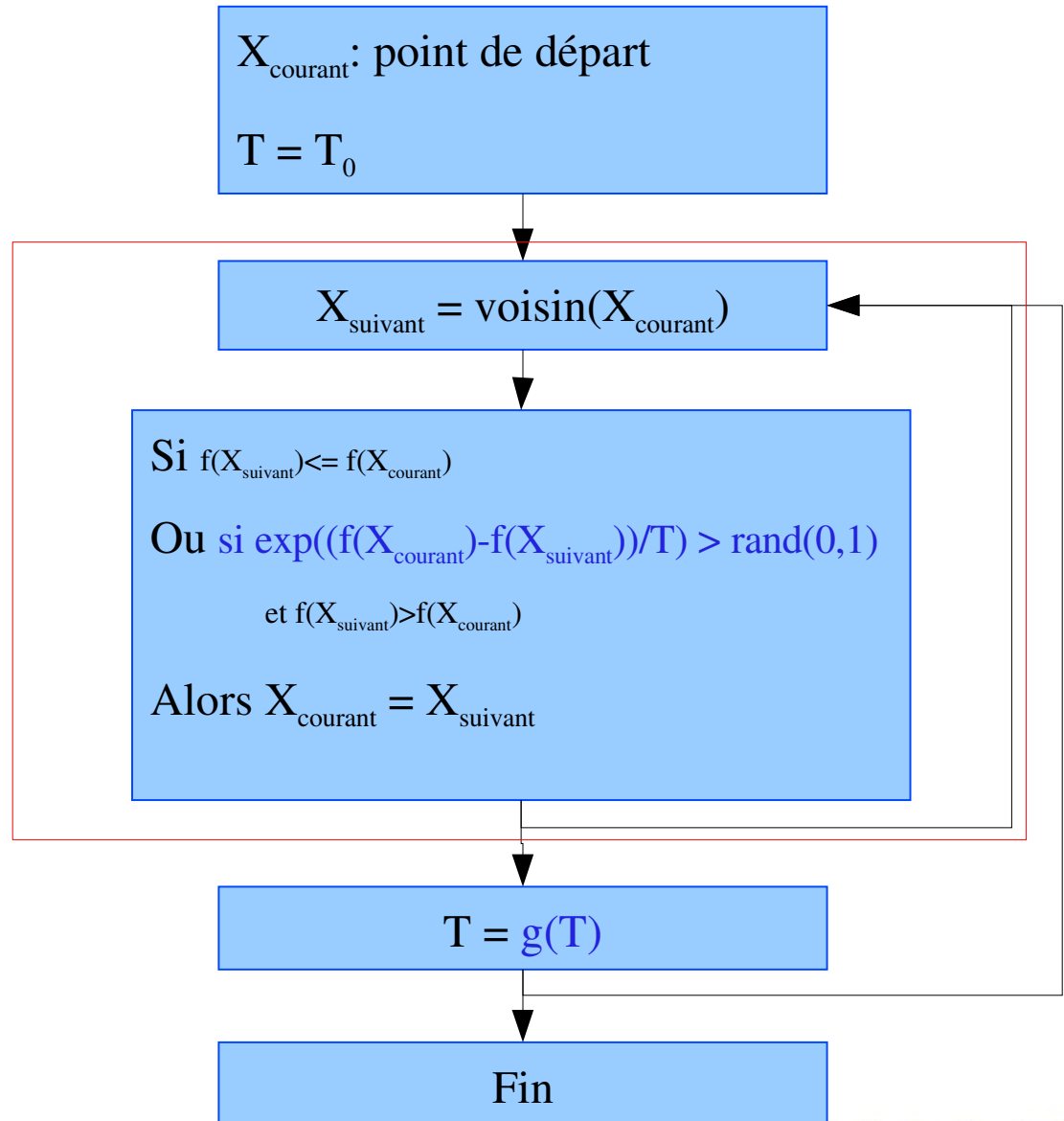
- ▶ Si voisin meilleur, gardé
- ▶ Sinon, accepte aléatoirement un voisin moins bon
- ▶ Probabilité commandé par un paramètre T (« température »)
- ▶ T décroît avec le temps



Contrôle de la proba
d'accepter une mauvaise
solution

Loi de décroissance
de la température

Méthode de Metropolis





Échantillonnage de distribution de probabilité

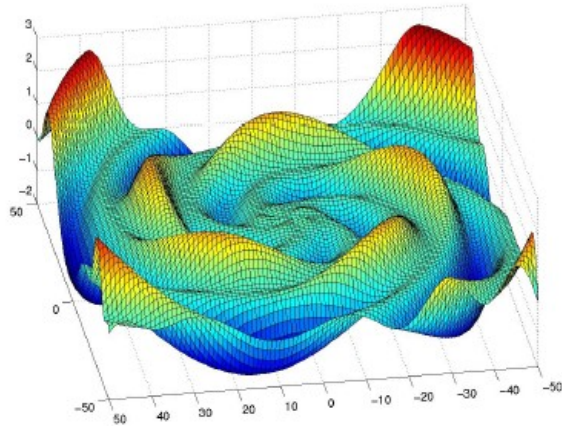
- ▶ Non intégrable, sans expression analytique, non normalisée
- ▶ Fonction objectif → distribution de probabilité
- ▶ Via distribution de Boltzman

$$p(x) = \frac{\exp(-f(x)/T)}{\sum_y \exp(-f(y)/T)}$$

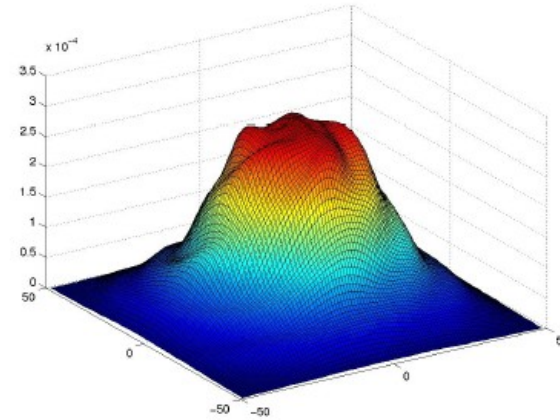
- ▶ Recuit simulé → échantillonner une distribution paramétrique (T)

Algorithme de Metropolis-Hastings

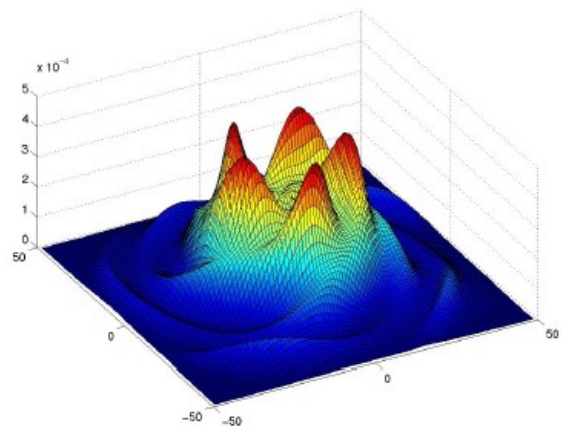
- ▶ n'importe quelle distribution



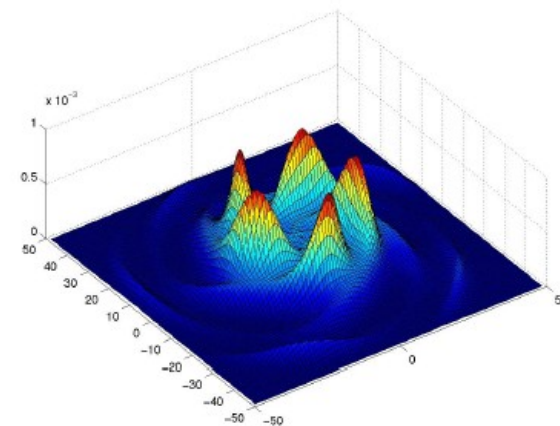
(a) Cost function



(b) $t=5$



(c) $t=1$



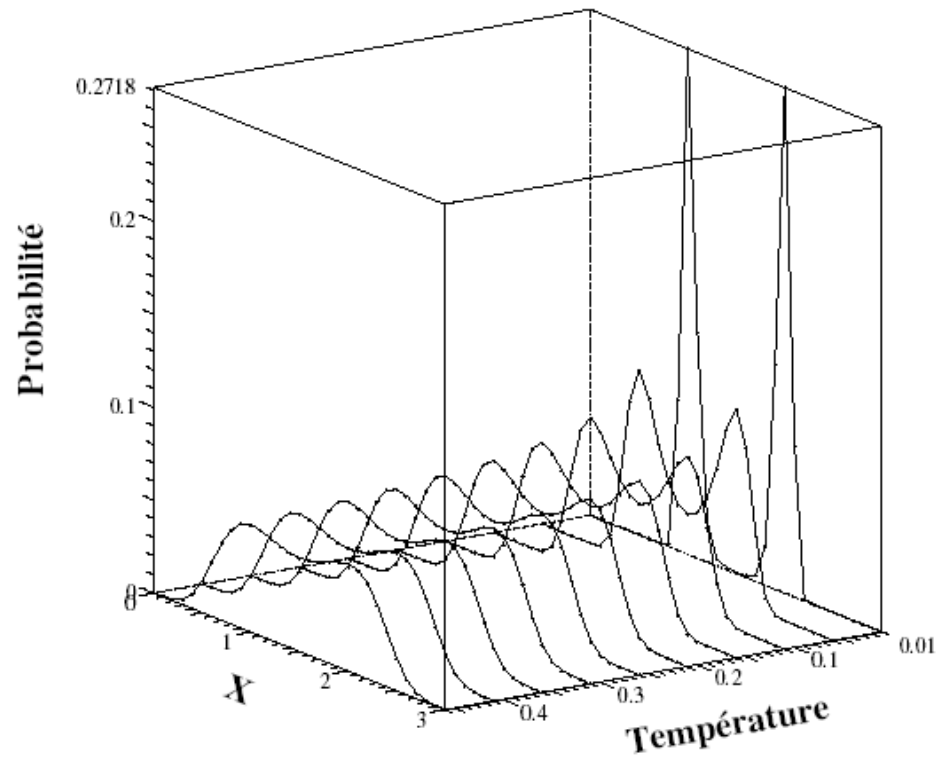
(d) $t=0.5$

Garantie d'atteindre l'optimum en un temps fini.

En pratique :

► $\lim_{t \rightarrow \infty} p_t(\hat{x}) = 1$

Densité de probabilité

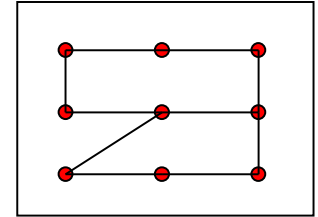




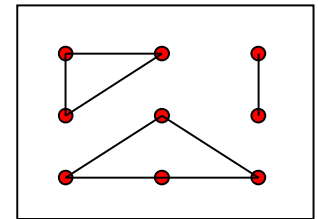
Ergodicité

- ▶ N'importe quelle solution atteignable
- ▶ Quasi-ergodicité : nombre fini de changements
- ▶ Voisinage

ergodique

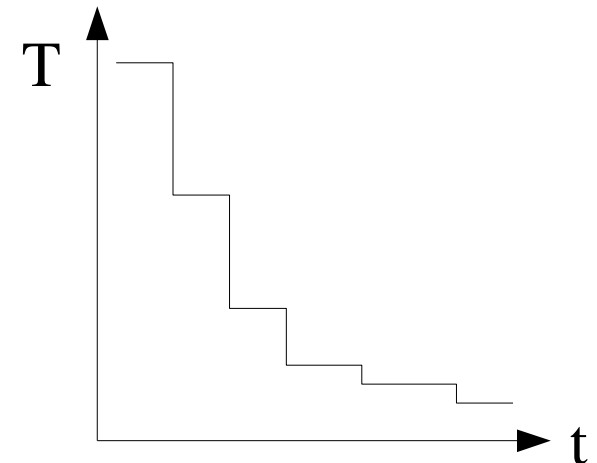


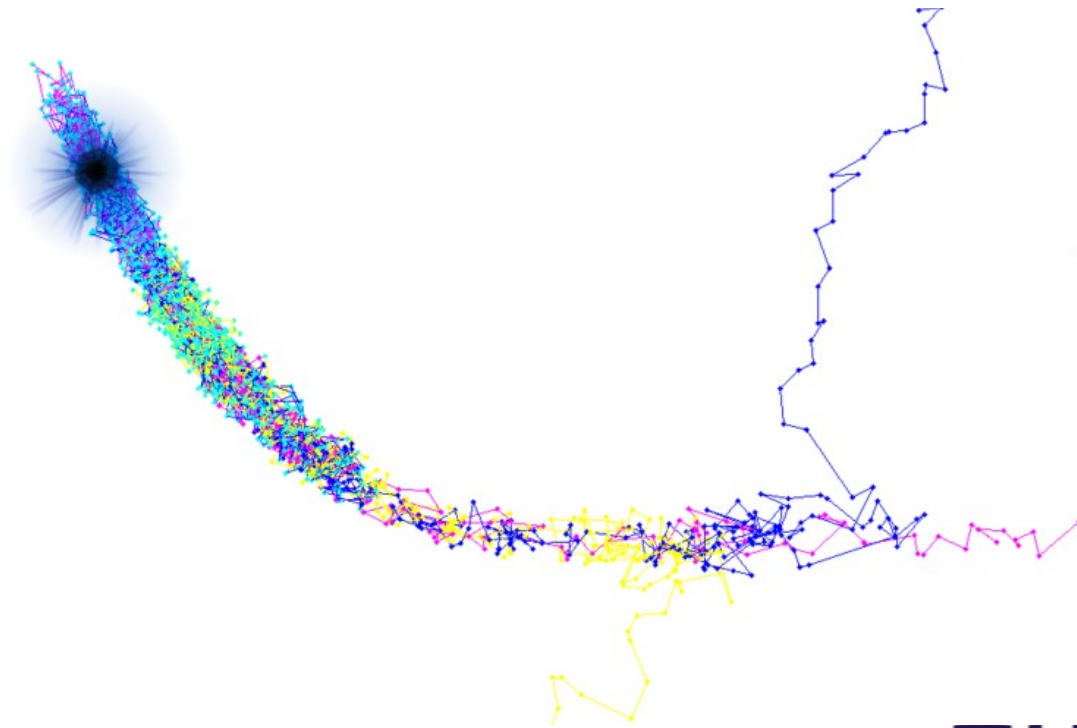
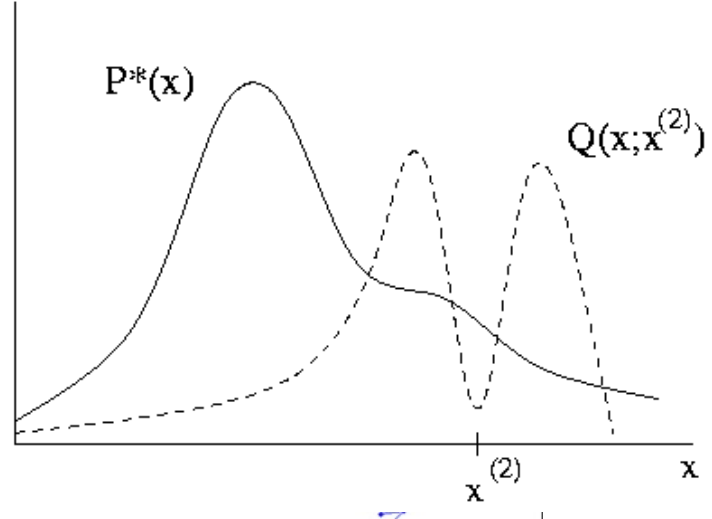
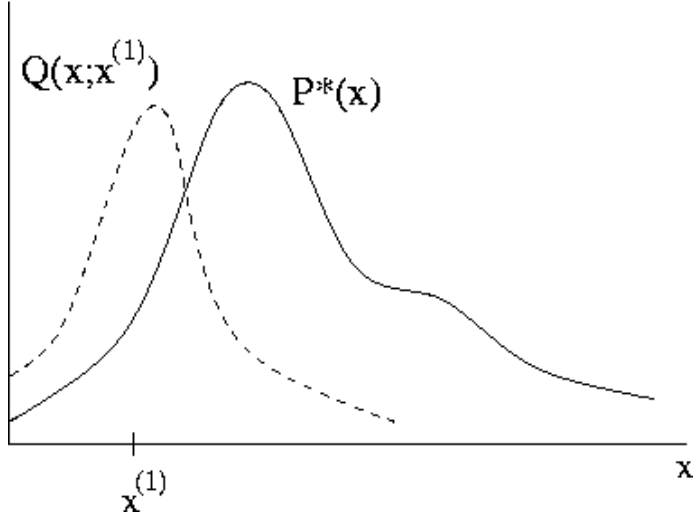
non ergodique



Convergence

- ▶ Quasi-ergodicité
- ▶ Décroissance par palier
- ▶ Décroissance graduelle
 - ▶ T_t ne décroît pas plus vite que : $C/\log(t)$
 - ▶ C , constante liée à échelle fonction objectif





La loi classique:

$$T_k = \alpha \cdot T_{k-1}$$

Les lois adaptatives:

Van Laarhoven:

$$T_k = T_{k-1} \frac{1}{1 + \frac{\ln(1 + \delta)}{3\sigma(T_{k-1})} T_{k-1}}$$

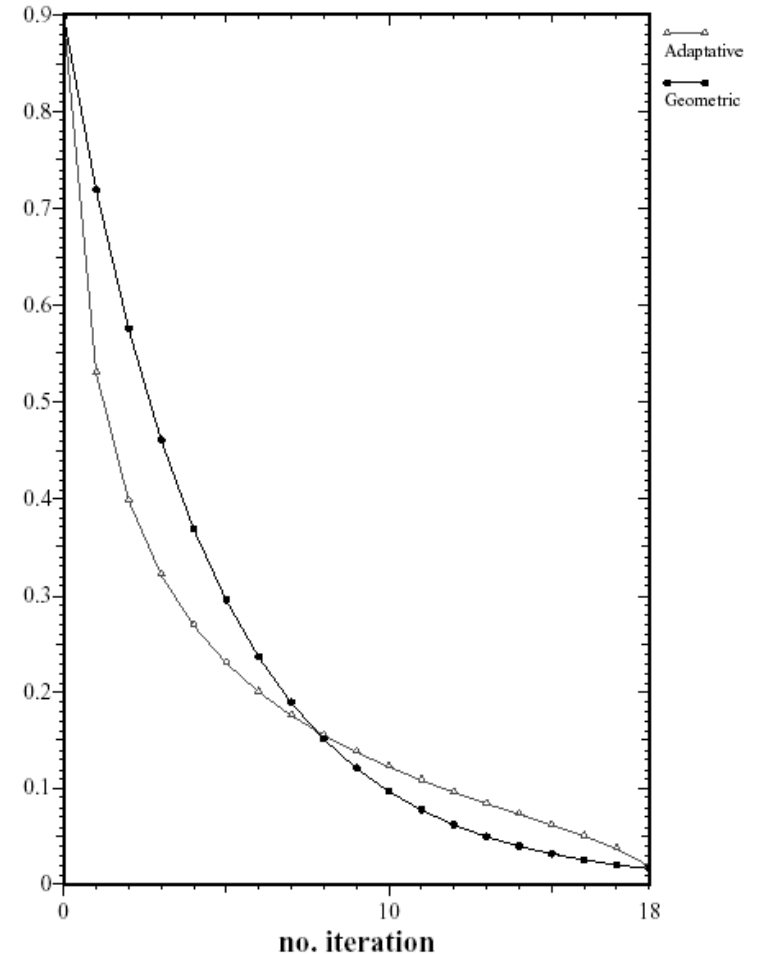
Huang:

$$T_k = T_{k-1} \exp\left(\frac{-\lambda T_{k-1}}{\sigma(T_{k-1})}\right)$$

Triki:

$$T_k = T_{k-1} \left(1 - T_{k-1} \frac{\Delta}{\sigma^2(T_{k-1})}\right)$$

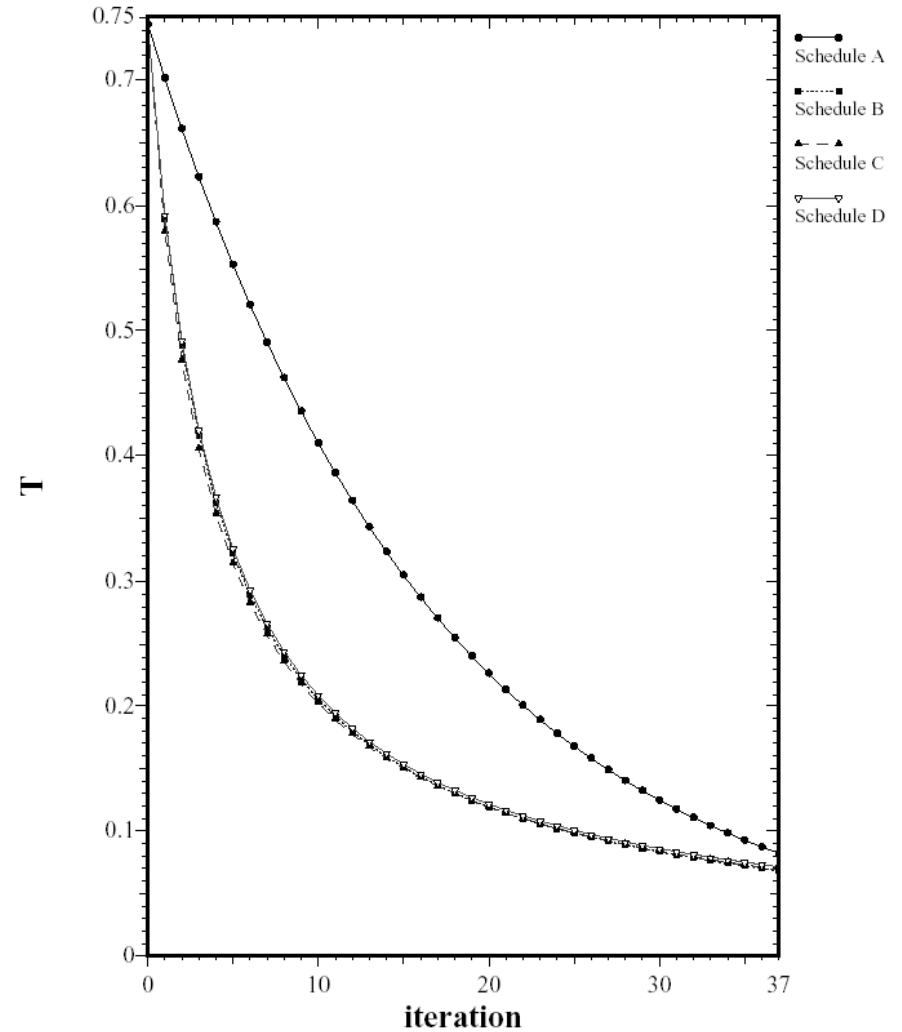
Cooling schedules

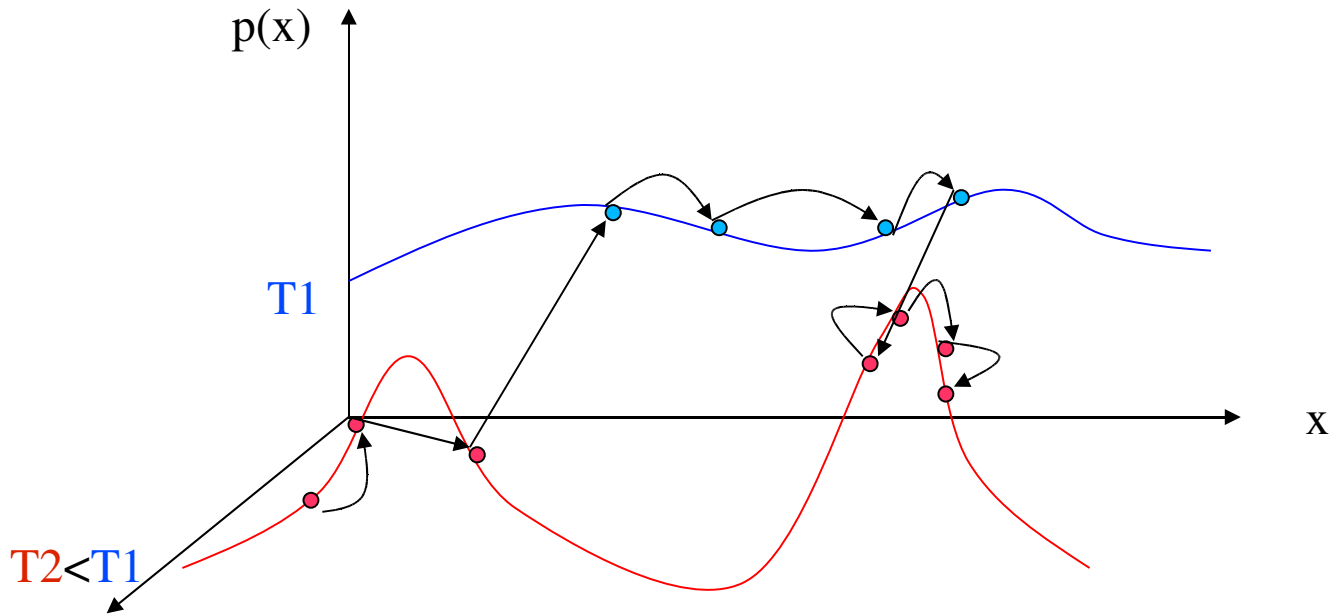
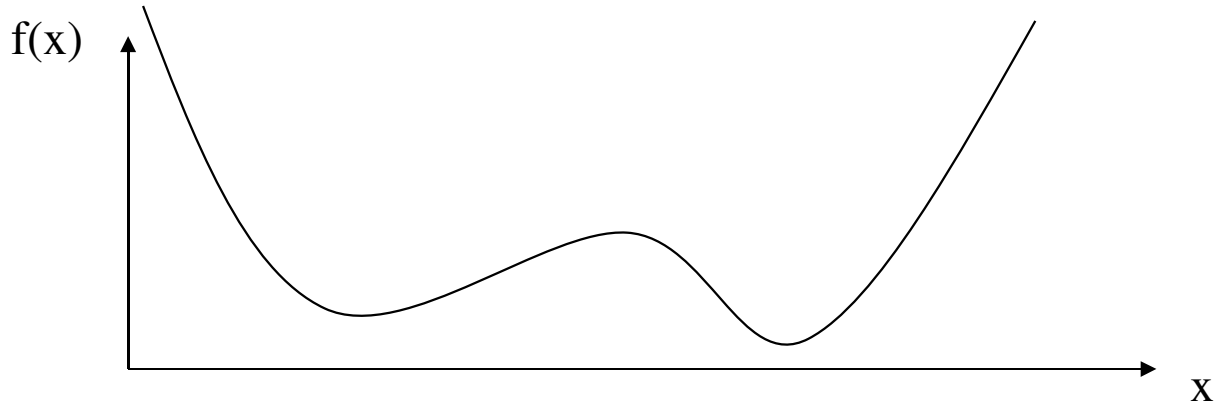


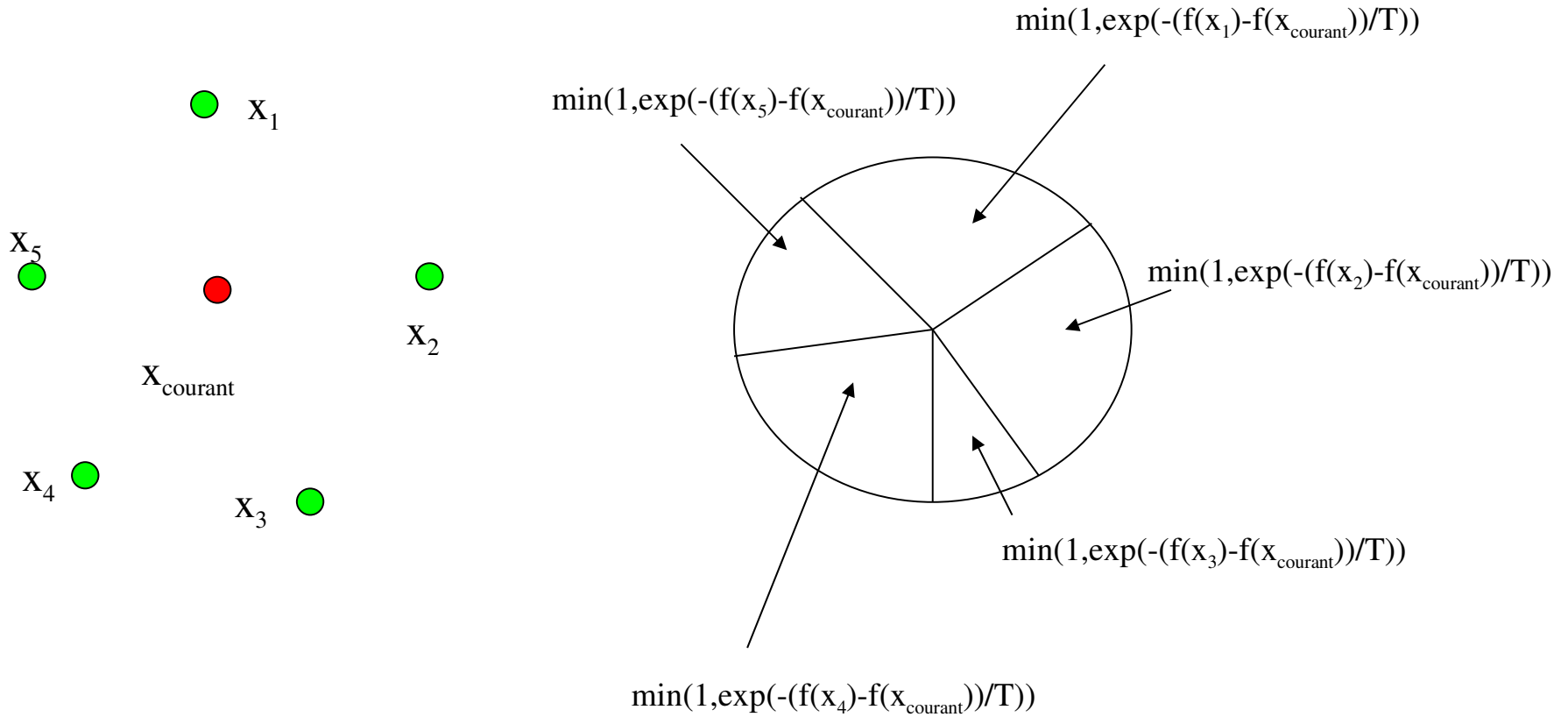


Temperature schedules

Loi de originale	$\Delta(T_k)$
Van Laarhoven	$\ln(1+\delta)/3.\sigma(T_k)$
Otten	$\ln(1+\delta)/C_{max}.\sigma^2(T_k)$
Huang	$\lambda.\sigma(T_k)$









Problèmes

- ▶ Exemples
 - ▶ Recherche opérationnelle
 - ▶ Ingénierie
 - ▶ Intelligence artificielle
- ▶ Caractéristiques

Applications

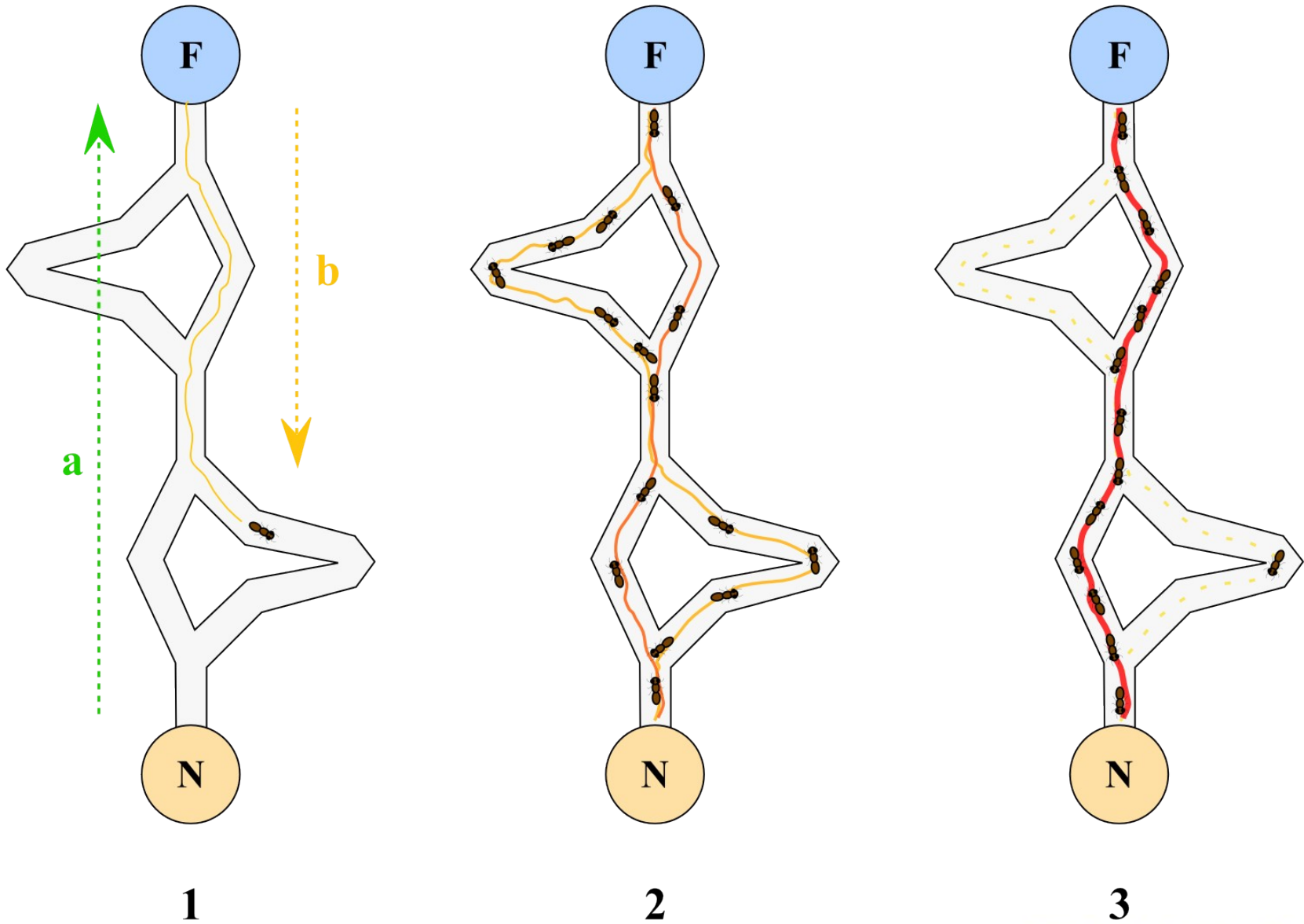
- ▶ Validation
 - ▶ Performances ?
 - ▶ Paramétrage
 - ▶ Validation
- ▶ Exemples

Métaheuristiques

- ▶ Exemples
 - ▶ Recherche tabou
 - ▶ Algorithmes évolutionnistes
 - ▶ Recuit simulé
 - ▶ Colonies de fourmis
 - ▶ Estimation de distribution
- ▶ Synthèse



- ▶ 1959, Pierre-Paul Grassé : stigmergie.
- ▶ 1983, Deneubourg, comportement collectif
- ▶ 1988, Moyson et Manderick, auto-organisation
- ▶ 1989, travaux de Goss, Aron, Deneubourg et Pasteels
- ▶ 1989, modèle de comportement de recherche de nourriture
- ▶ 1991, M. Dorigo propose le Ant System
- ▶ 1997, première application aux réseaux de télécommunications
- ▶ 2000, première preuve de convergence
- ▶ 2004, preuve d'équivalence avec d'autres métaheuristiques



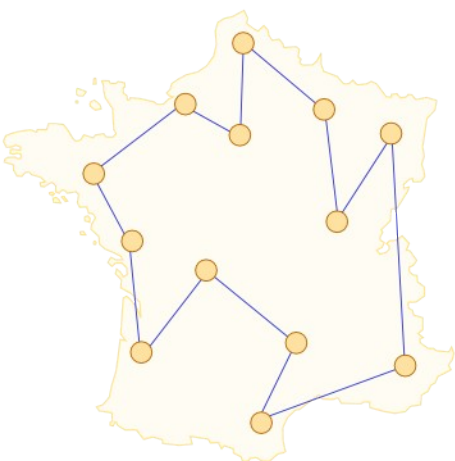


Initialisation

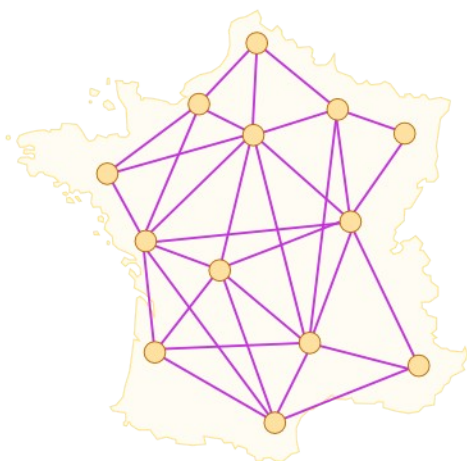
Jusqu'à critère d'arrêt

- ▶ Pour chaque fourmis
 - ▶ Parcours d'un trajet
 - ▶ Laisser piste (quantité fixe, étalée sur chaque arrêt)
- ▶ Évaporation

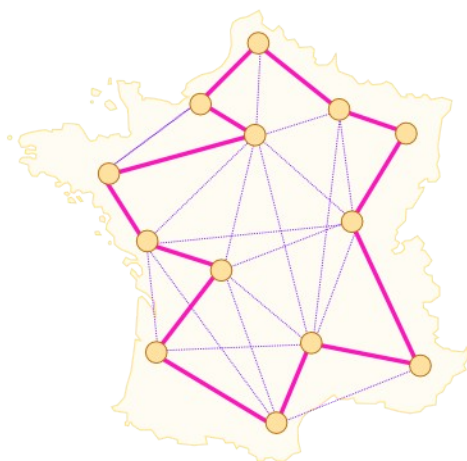
Fin



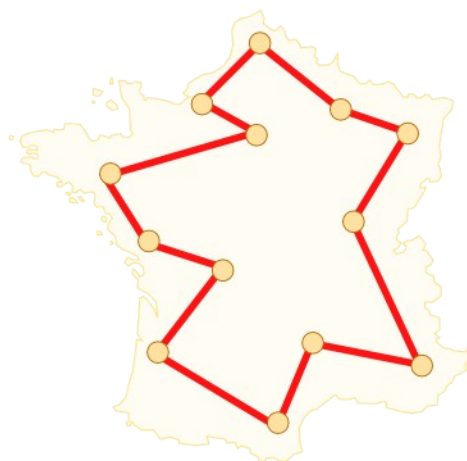
1



2



3



4



- ▶ t itérations, k fourmis, n villes, J_i^k villes à visiter

Choix d'une ville : visibilité

- ▶ $\eta_{ij} = 1/d_{ij}$

Quantité de phéromone

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{L^k(t)} & \text{si } (i, j) \in T^k(t) \\ 0 & \text{sinon} \end{cases}$$

Choix ville

$$p_{ij}^k(t) = \begin{cases} \frac{\tau_{ij}(t)^\alpha (\mu_{ij})^\beta}{\sum_{l \in J_i^k} \tau_{il}(t)^\alpha (\mu_{il})^\beta} & \text{si } j \in J_i^k \\ 0 & \text{si } j \notin J_i^k \end{cases}$$

Évaporation

$$\tau_{ij}(t+1) = (1 - \rho) \tau_{ij}(t) + \Delta \tau_{ij}(t)$$



MMAS

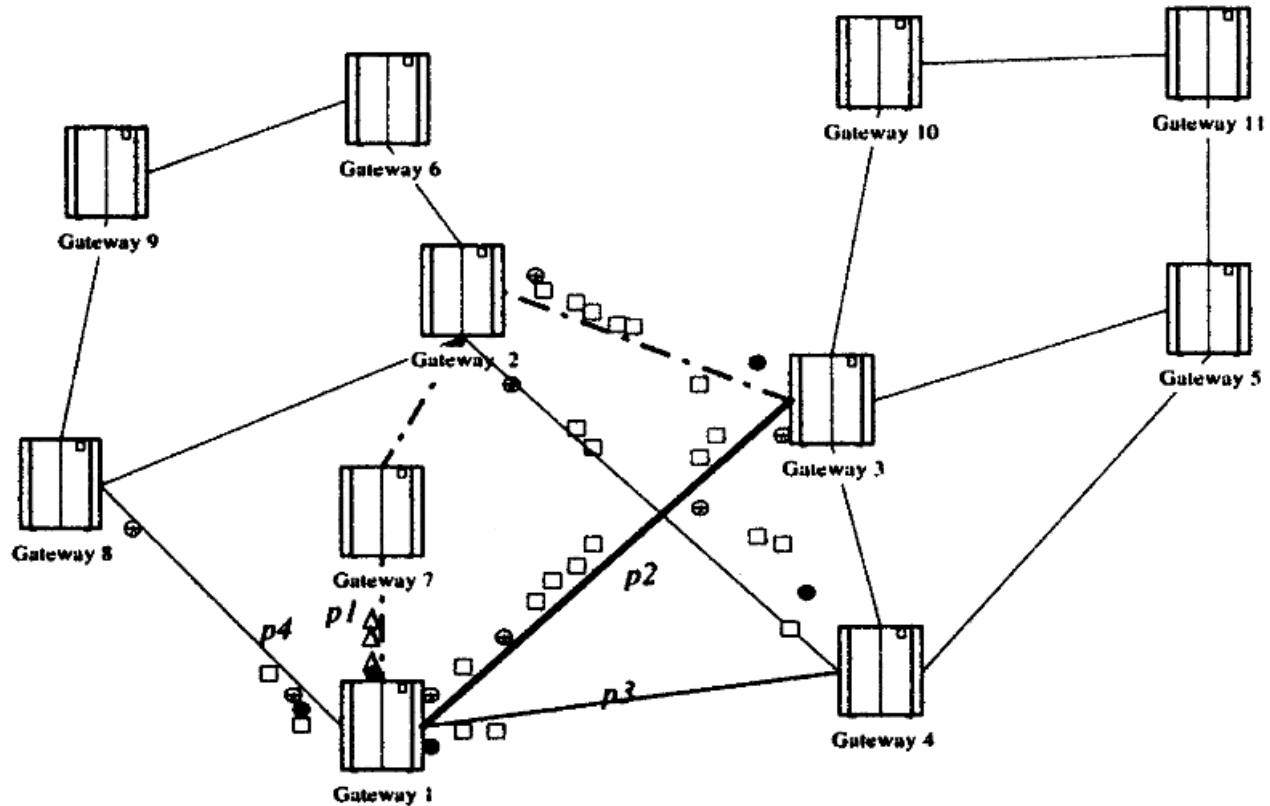
- ▶ $\tau_{\min} < \tau_{ij} < \tau_{\max}$
- ▶ Ergodicité







HCAS

- ▶ Problème codage binaire
- ▶ Normalisation « fonction de qualité »

- ▶
$$\tau_i = (1 - \rho)\tau_i + \rho \frac{\sum_{s \in S_t} Q_f(s) s_i}{\sum_{s \in S_t} Q_f(s)}$$

- ▶
$$Q_f(s | S_1, \dots, S_t) = \tau_0 \left(1 - \frac{f(s) - LB}{\bar{f} - LB} \right)$$



MAG1	
MAG2	
Data packets after call set up by MAG1	
Data packets after call set up by MAG2	
Optimal path p_2 to Gateway 3	
'Near optimal' path p_1 to Gateway 3	



Problèmes

- ▶ Exemples
 - ▶ Recherche opérationnelle
 - ▶ Ingénierie
 - ▶ Intelligence artificielle
- ▶ Caractéristiques

Applications

- ▶ Validation
 - ▶ Performances ?
 - ▶ Paramétrage
 - ▶ Validation
- ▶ Exemples

Métaheuristiques

- ▶ Exemples
 - ▶ Recherche tabou
 - ▶ Algorithmes évolutionnistes
 - ▶ Recuit simulé
 - ▶ Colonies de fourmis
 - ▶ Estimation de distribution
- ▶ Synthèse





- ▶ 1994 : apprentissage incrémental à population (PBIL)
- ▶ 1996 : algorithmes à estimation de distribution
- ▶ 1999 : compact genetic algorithm
- ▶ 2004 : équivalence avec d'autres métaheuristiques



Idée originale

- ▶ Simplifier les algorithmes génétiques
- ▶ Un seul opérateur

Population Base Incremental Learning Compact Genetic Algorithm



Idées de base

- ▶ Distribution de probabilité choisie *a priori*
- ▶ Échantillonnage
- ▶ Opérateurs de réduction de variance (→ sélection)

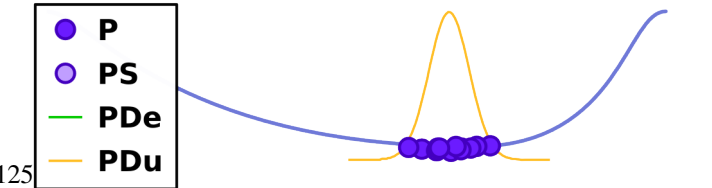
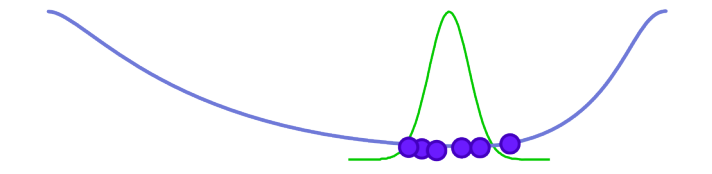
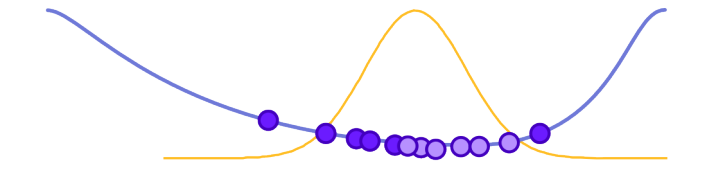
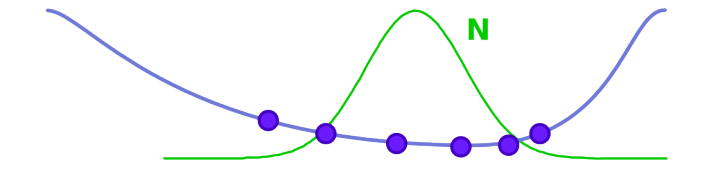
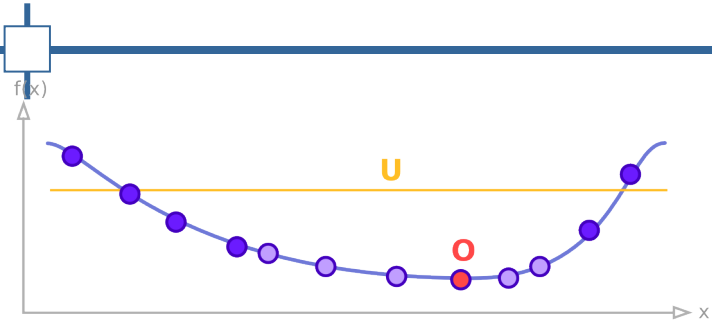


Initialisation

Jusqu'à critère d'arrêt

- ▶ Estimation des paramètres
- ▶ Tirage échantillon selon distribution
- ▶ Sélection

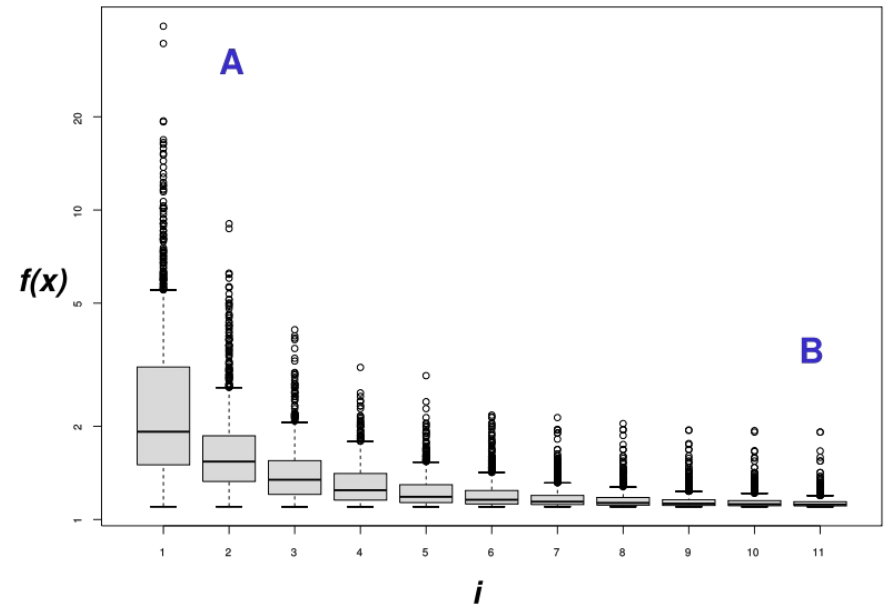
Fin



- P
- PS
- PDe
- PDu

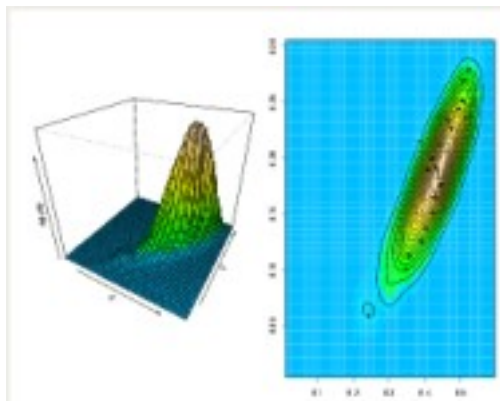
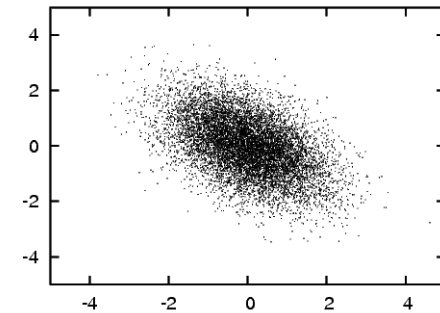
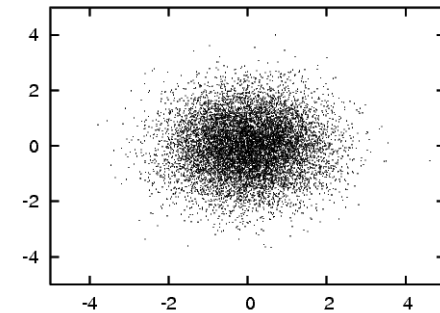
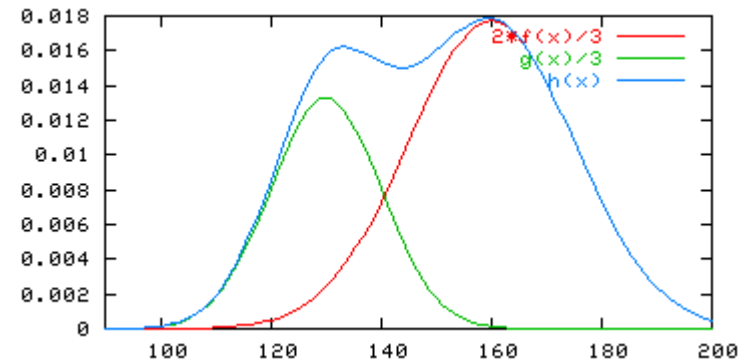
Distributions :

- ▶ Uniforme
- ▶ Gaussienne univariante



Modèles classiques

- ▶ Gaussienne univariante
- ▶ Mélange de gaussiennes univariantes
- ▶ Gaussienne multivariante
- ▶ Mélange de gaussiennes multivariantes



i	x1	x2	x3	f(x)
1	0	1	0	1
2	0	1	0	1
3	1	0	1	2
4	1	0	1	2
5	0	1	1	2
6	1	0	0	1
p(x)	0.5	0.5	0.5	



i	x1	x2	x3	f(x)
1	0	1	0	1
2	0	1	0	1
3	1	0	1	2
4	1	0	1	2
5	0	1	1	2
6	1	0	0	1
p(x)	0.5	0.5	0.5	

i	x1	x2	x3	f(x)
3	1	0	1	2
4	1	0	1	2
5	0	1	1	2
p(x)	0.7	0.3	1	



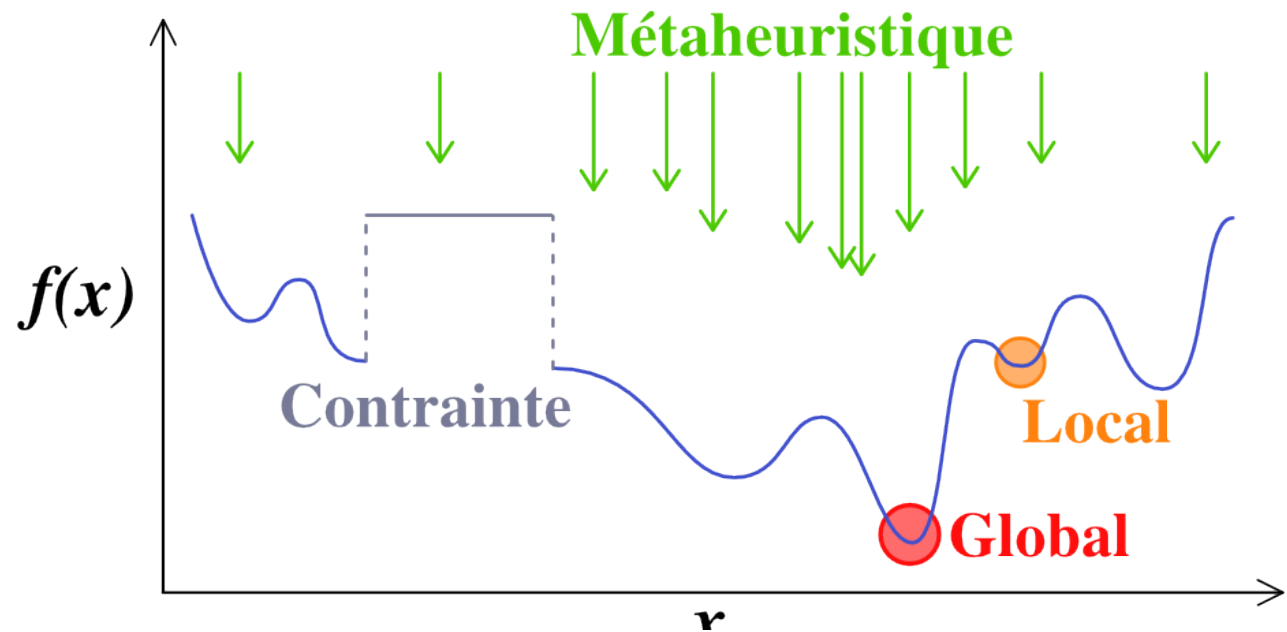
i	x1	x2	x3	f(x)
1	0	1	0	1
2	0	1	0	1
3	1	0	1	2
4	1	0	1	2
5	0	1	1	2
6	1	0	0	1
p(x)	0.5	0.5	0.5	

i	x1	x2	x3	f(x)
3	1	0	1	2
4	1	0	1	2
5	0	1	1	2
p(x)	0.7	0.3	1	

i	x1	x2	x3	f(x)
1	1	1	1	3
2	0	1	1	2
3	1	0	1	2
4	1	0	1	2
5	1	0	1	2
6	0	0	1	1
p(x)	0.7	0.3	1	

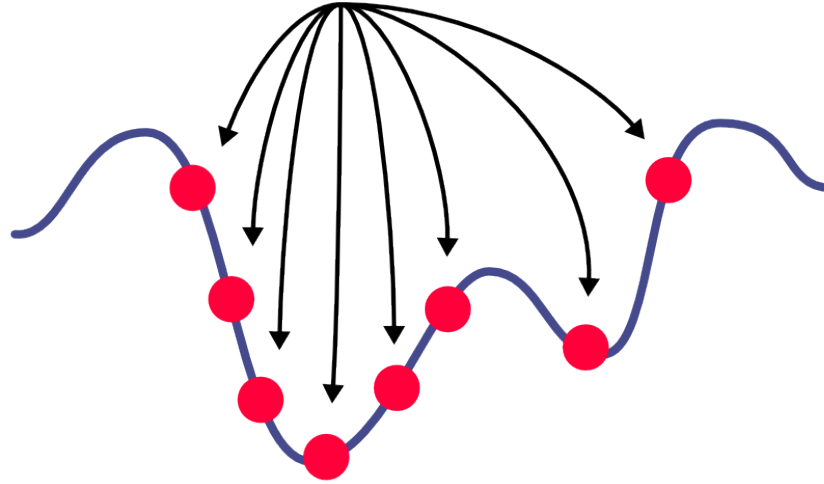


- ▶ Global
- ▶ Généraliste
- ▶ Stochastiques
- ▶ Facile à implémenter

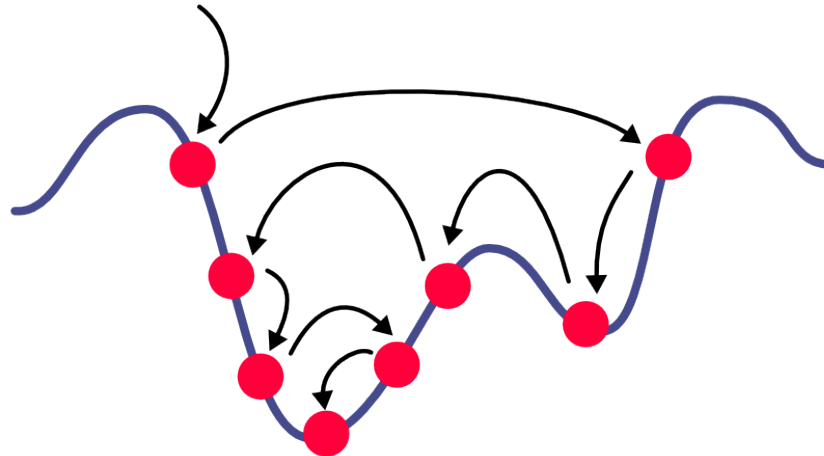


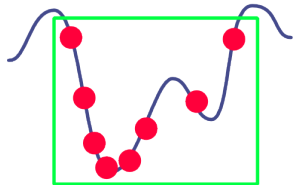


(a)



(b)



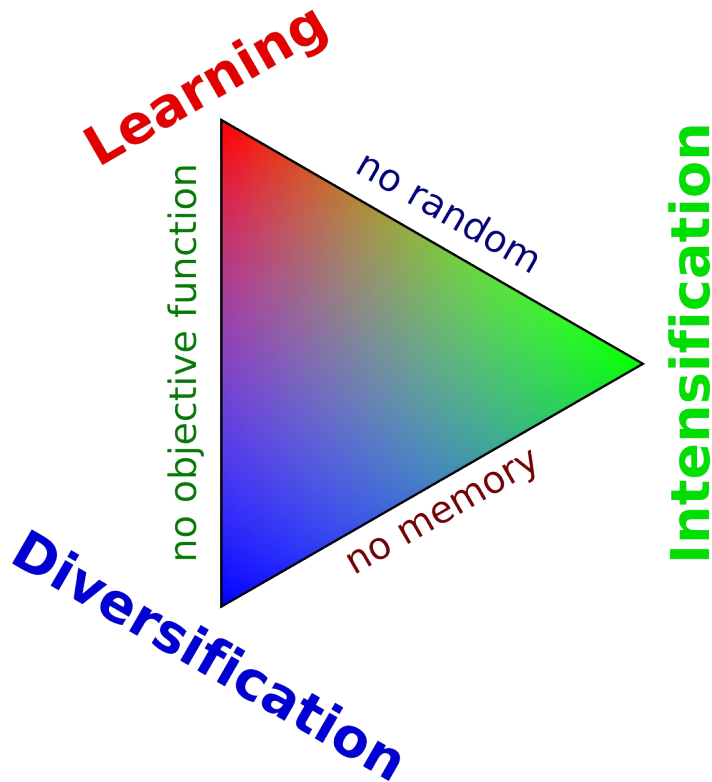
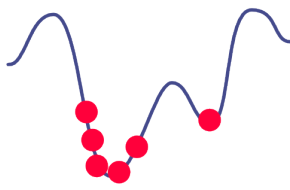
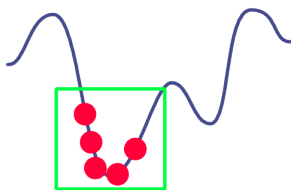


Diversification

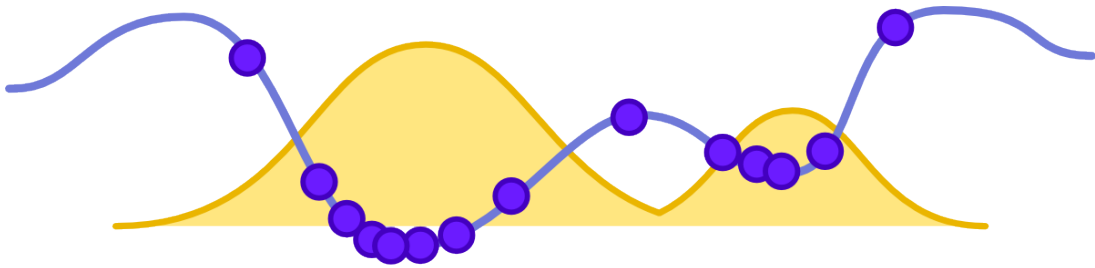


Apprentissage

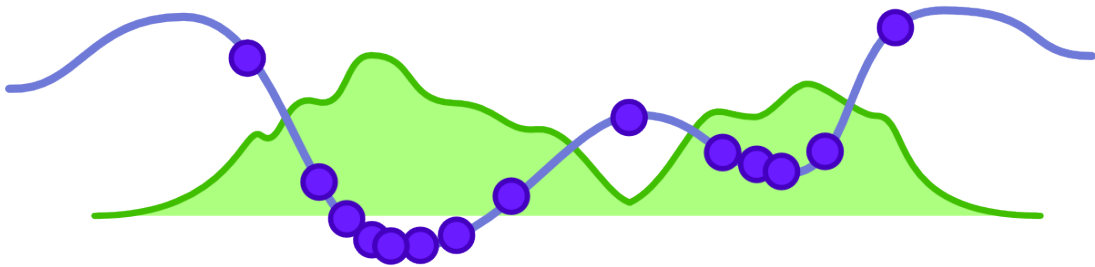
Intensification



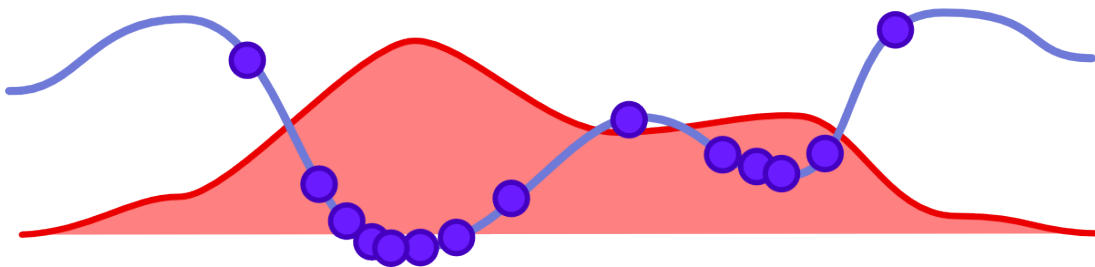
Diversification = exploration
Intensification = exploitation



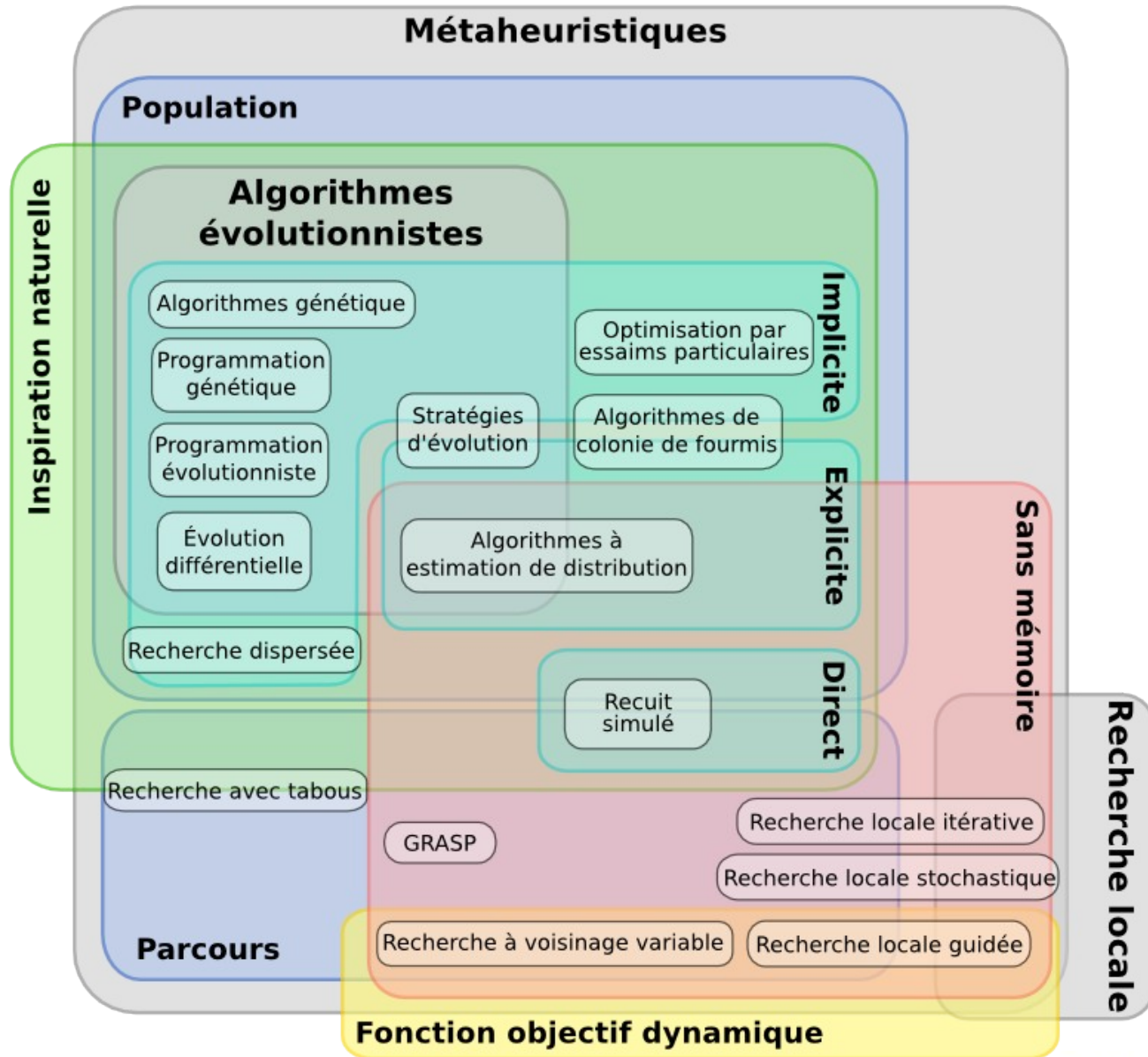
E.



I.



D.





Problèmes

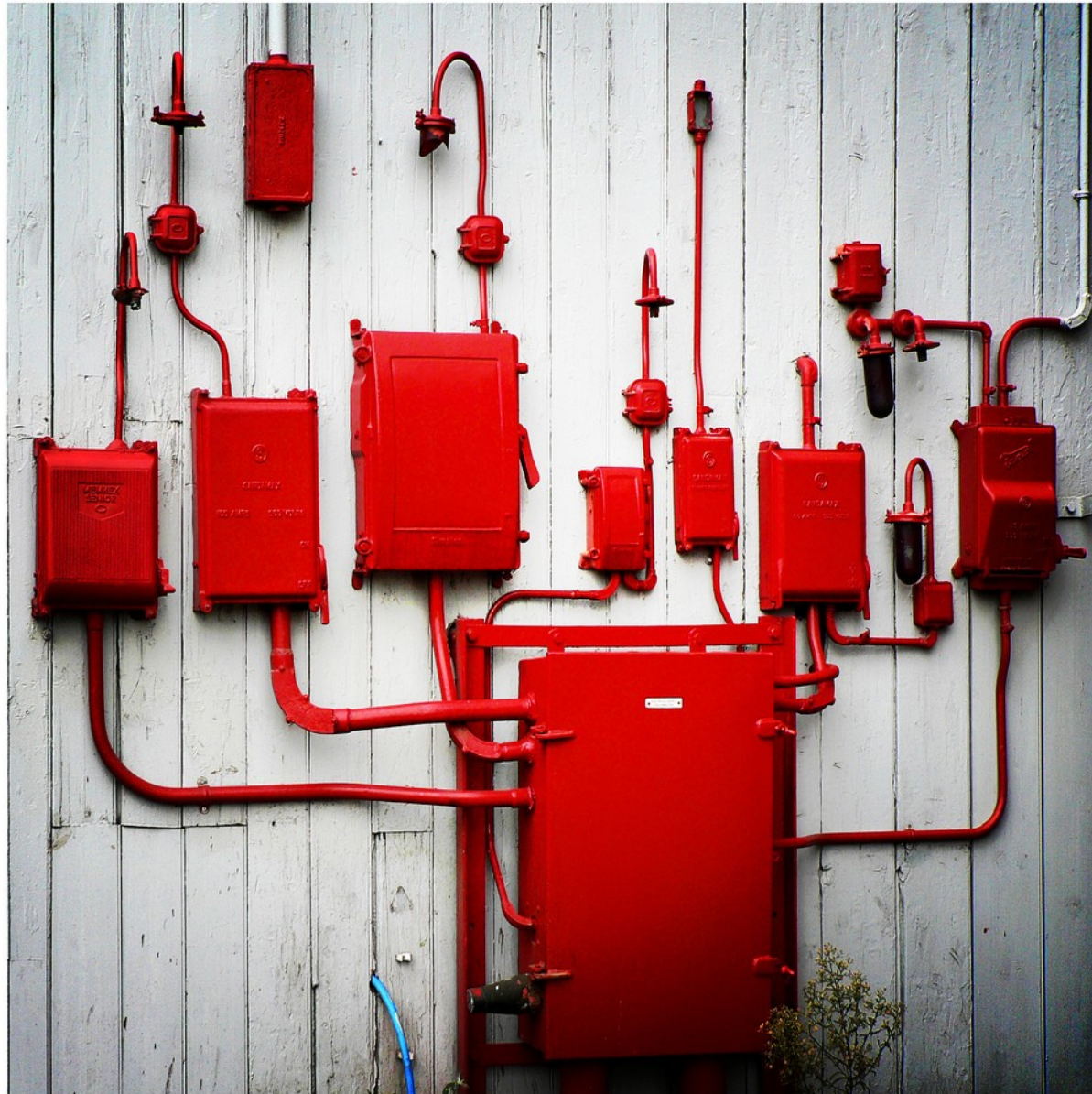
- ▶ Exemples
- ▶ Caractéristiques

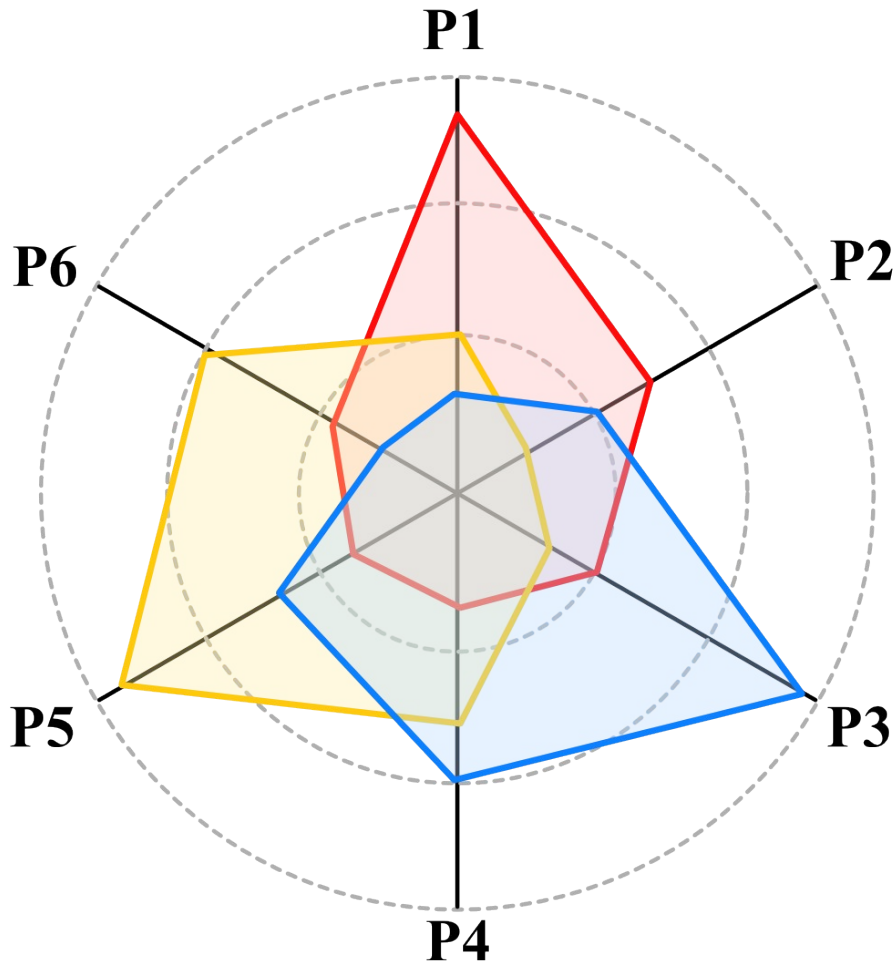
Métaheuristiques

- ▶ Exemples
- ▶ Synthèse

Applications

- ▶ Validation
 - ▶ Performances ?
 - ▶ Paramétrage
 - ▶ Validation
- ▶ Exemples





M1 

M2 

M3 

Instance de méthode

- ▶ Algorithme
- ▶ Paramétrage

Instance de problème

- ▶ Modèle
- ▶ Caractéristiques



Instance métaheuristique

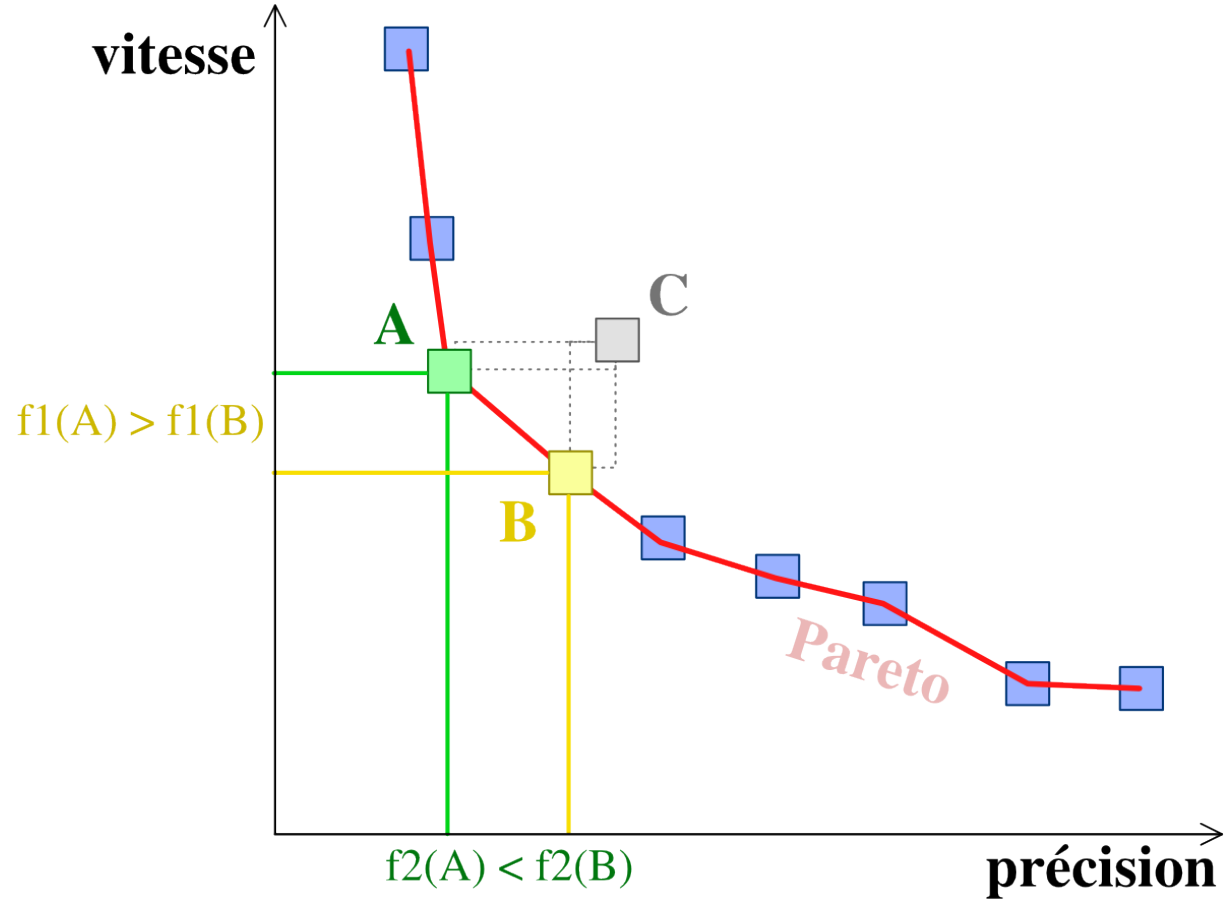
- ▶ 1 paramétrage optimal pour 1 instance de problème
 - ▶ Initialisation
 - ▶ Critère d'arrêt
 - ▶ Valeurs des paramètres

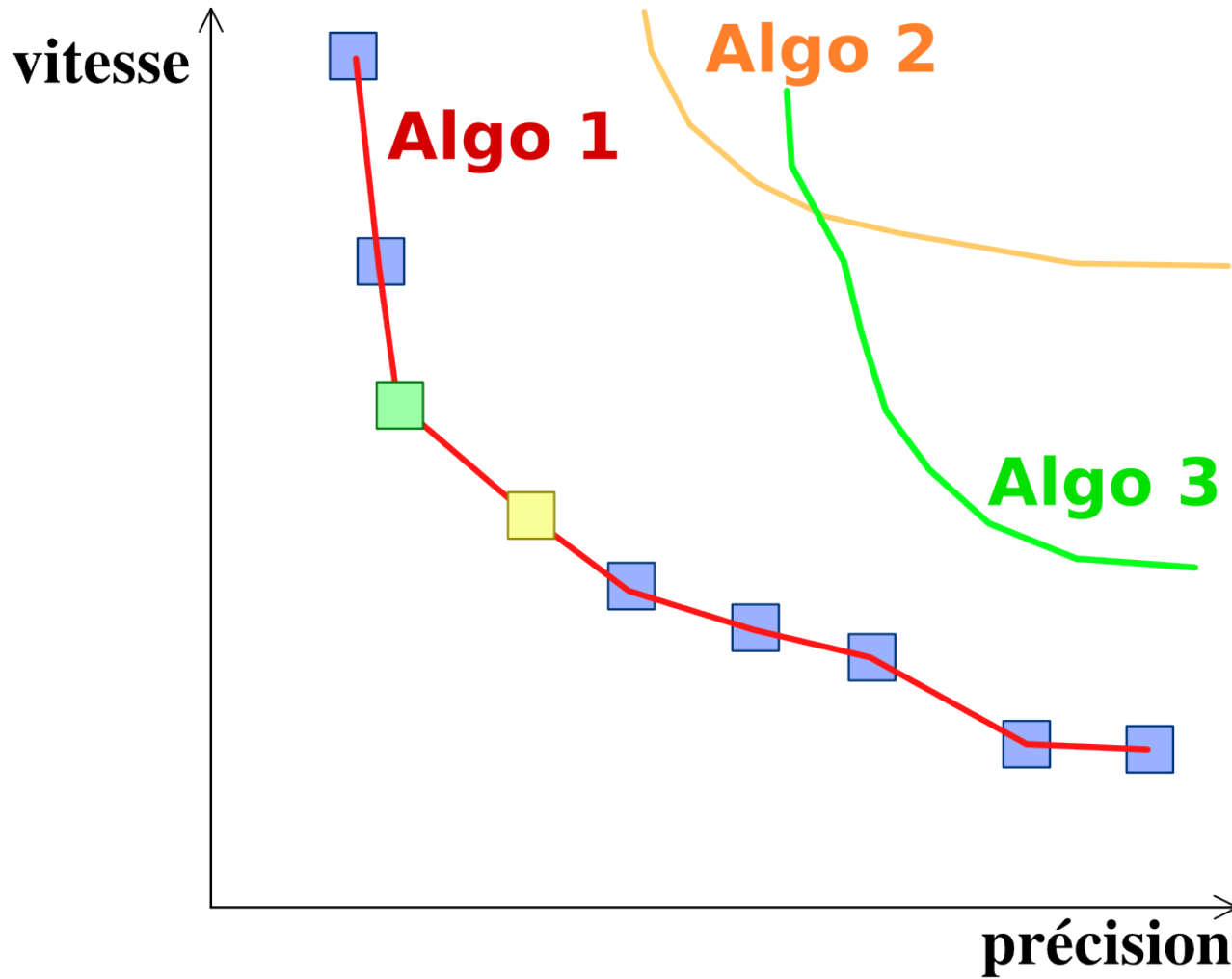
Quel paramétrage ?

- ▶ Critères de performance
 - ▶ Vitesse ou précision ?
 - ▶ Production ou conception ?



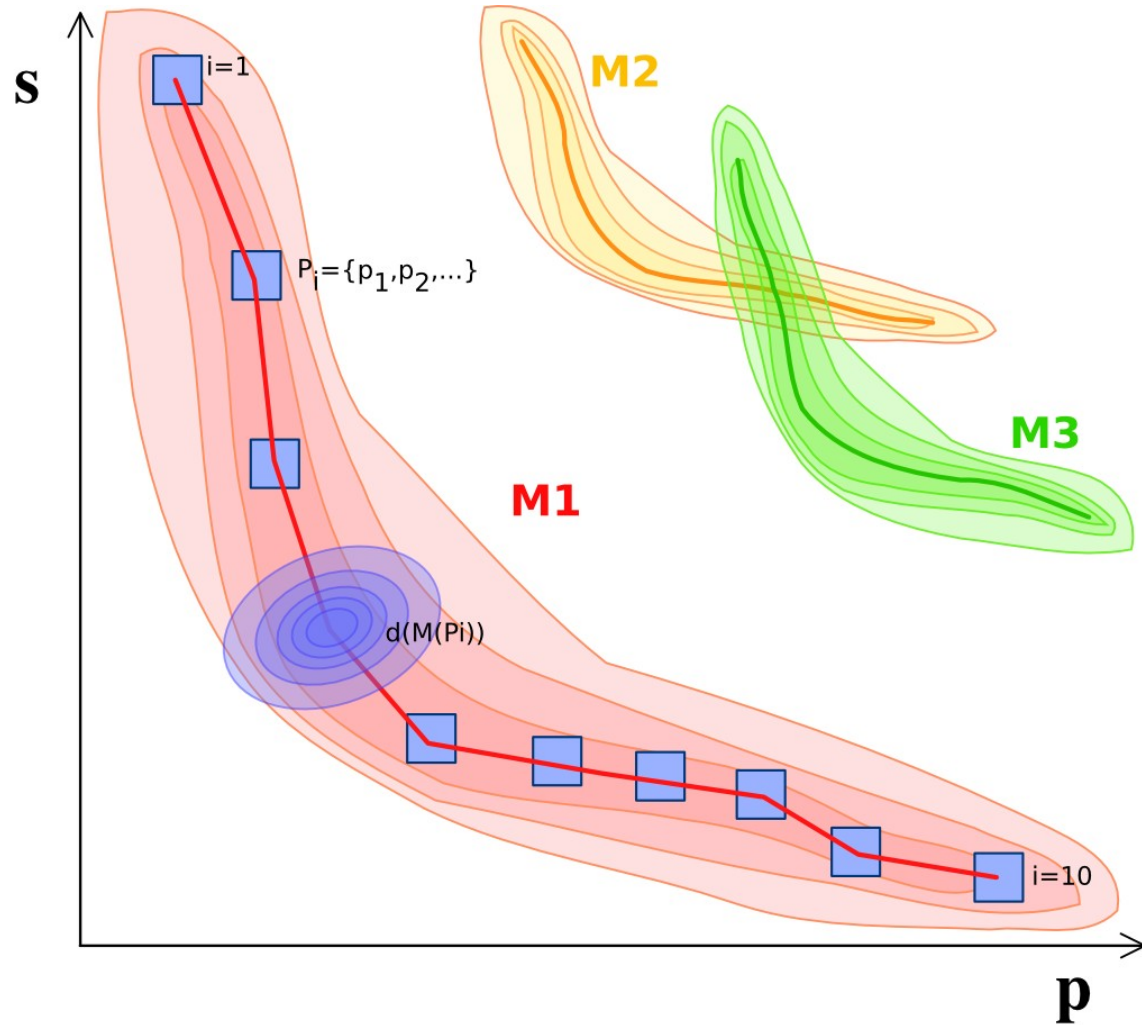
Vitesse
(x)OU
Précision





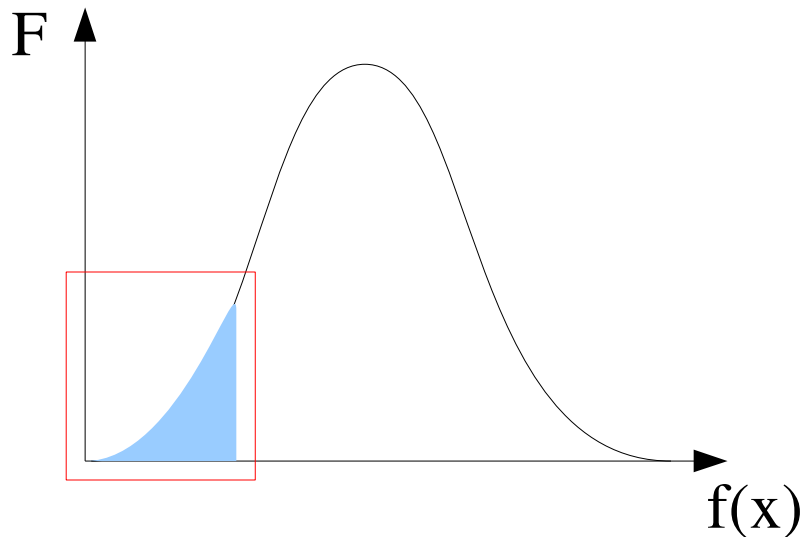


+ Stochastique



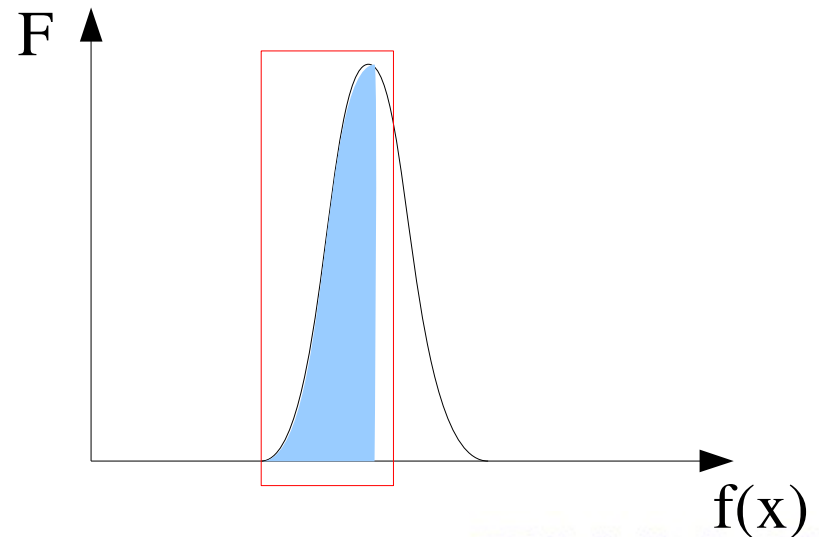
Conception

- ▶ Vitesse négligeable
- ▶ Précision cruciale
- ▶ Répétitions possibles
 - ▶ Recherche
 - ▶ Diversification



Production

- ▶ Vitesse cruciale
- ▶ Précision négligeable
- ▶ 1 seul essai
 - ▶ Robustesse
 - ▶ Intensification





Probabiliste → STATISTIQUE

Plan d'expérience

- ▶ Paramétrage
- ▶ Répétitions
- ▶ Tests statistiques
- ▶ Analyse de données



Simple

- ▶ Combinaisons de paramètres
- ▶ Répétitions
- ▶ Paramétrage optimal

Moins simple

- ▶ Problème d'optimisation
- ▶ Problème d'estimation d'erreur

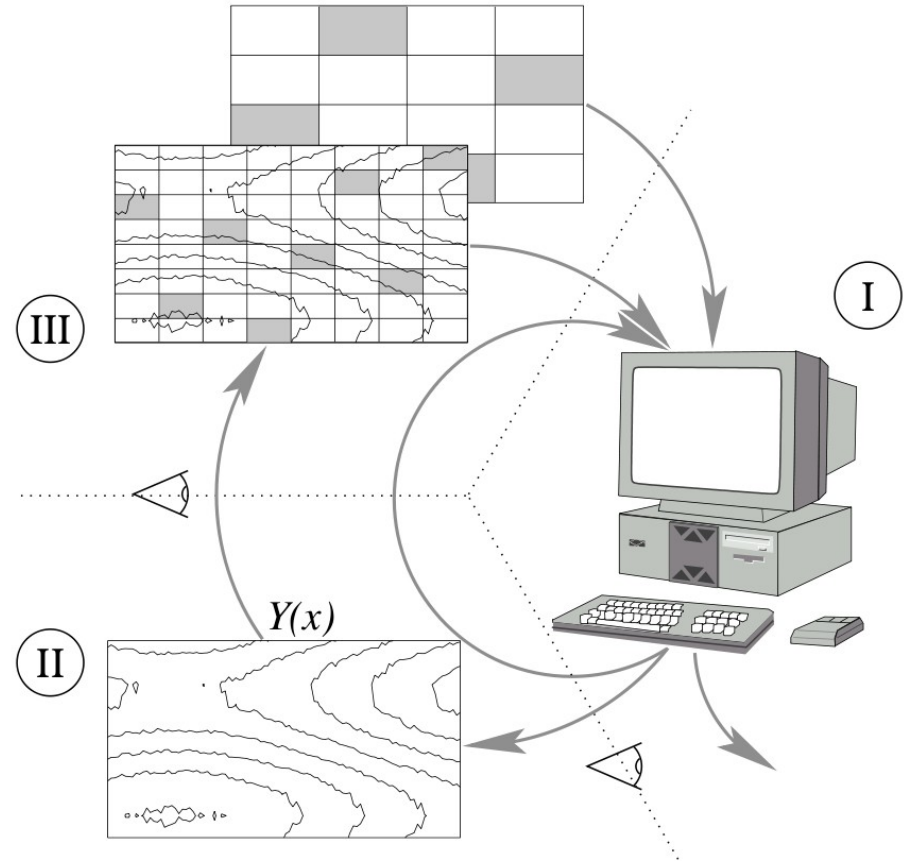
(I) analyse expérimentale

- ▶ De plusieurs jeux de paramètres

(II) estimation des performance

- ▶ Modèle de processus stochastique

(III) détermination de jeux supplémentaires à tester





Évaluation expérimentale

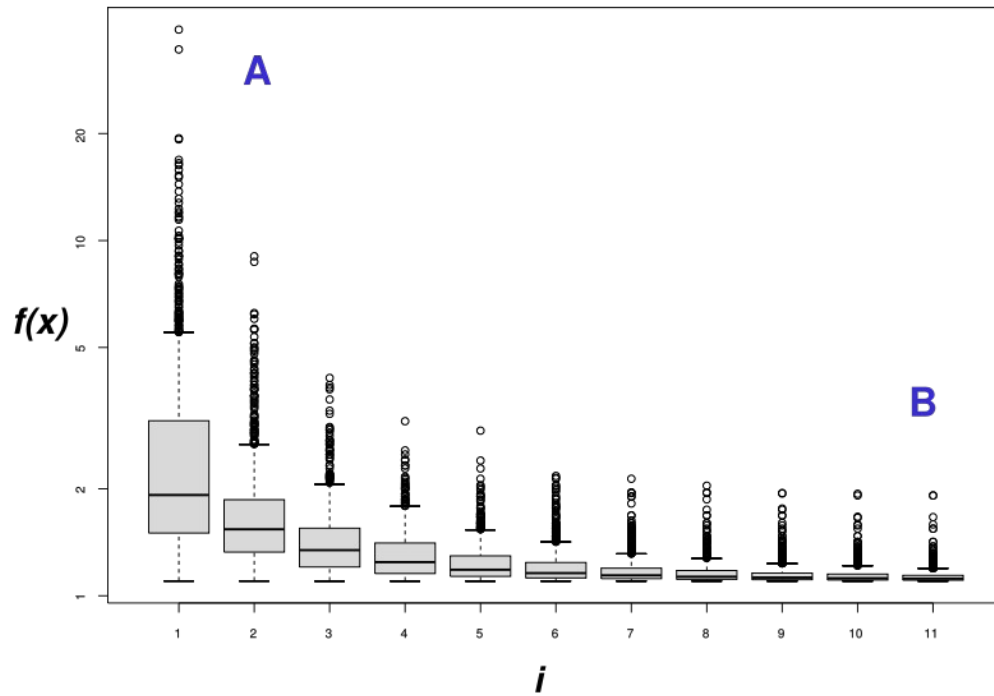
- ▶ Plusieurs évaluations par jeu
- ▶ Meilleur jeu précédent re-évalué
- ▶ Nouveaux jeux évalués autant

Modélisation

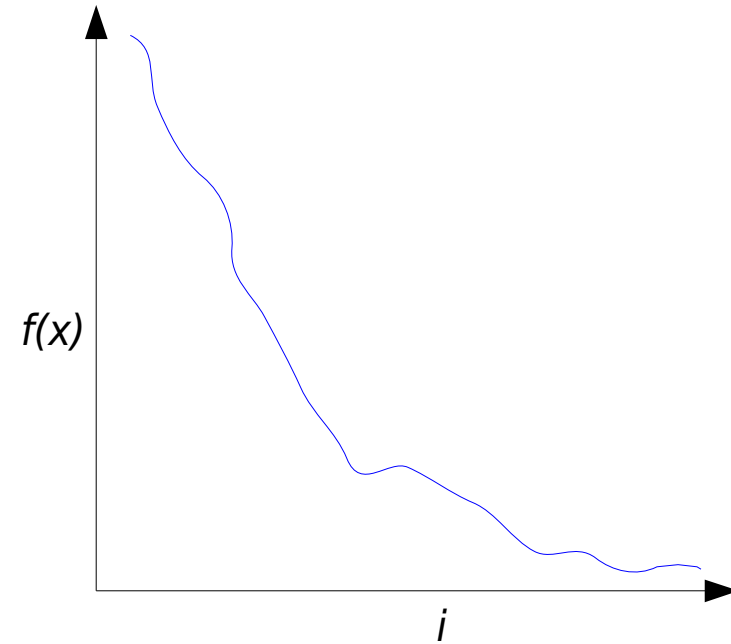
- ▶ Corrélacion gaussienne
- ▶ Régression polynomiale ordre 2
- ▶ Estimation des performances pour jeux non testés

Nouveaux jeux à tester

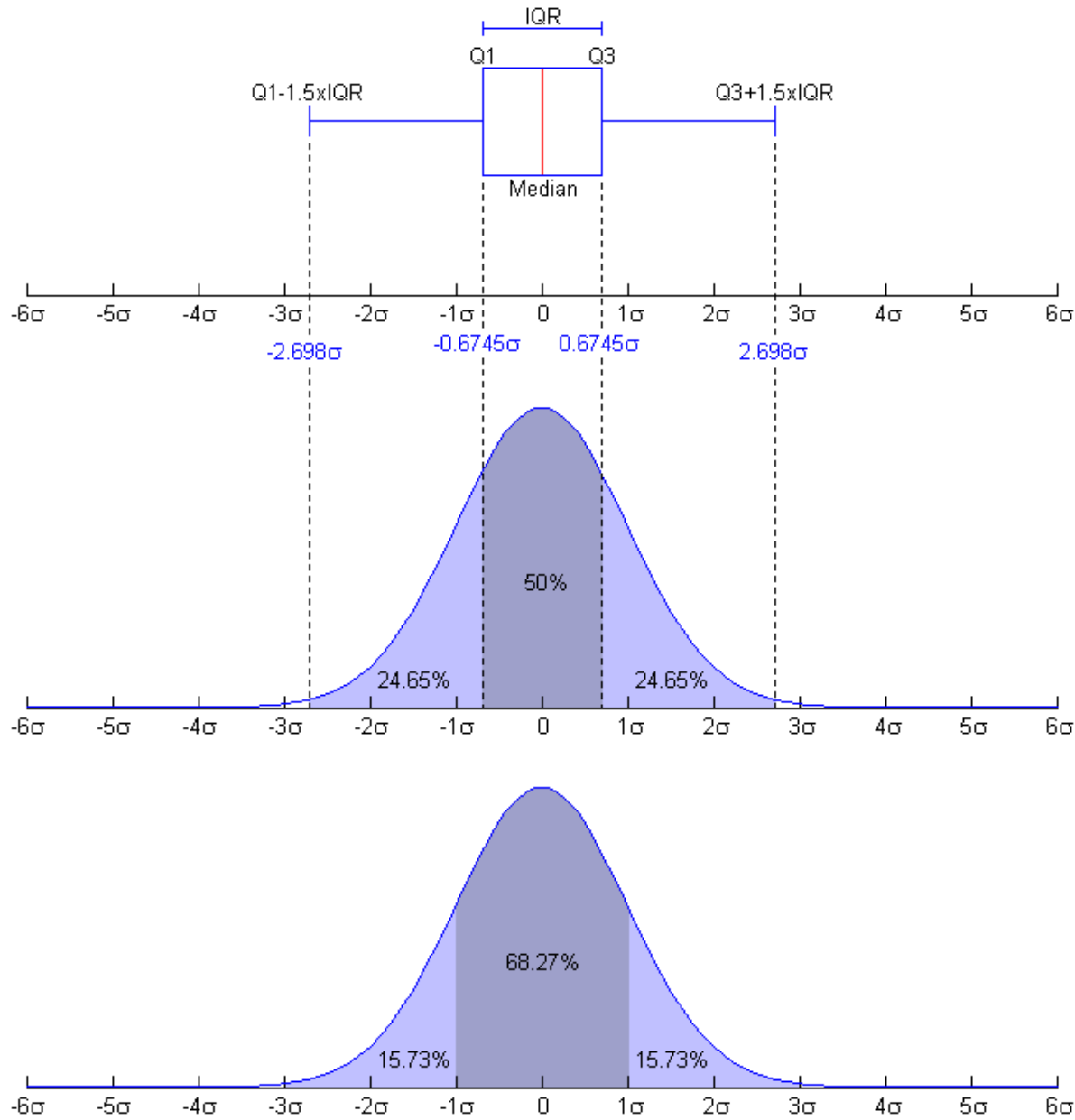
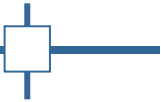
- ▶ Échantillonnage « Latin » de l'hypercube
 - ▶ Un seul essai par ligne/colonne



Bien



Pas bien





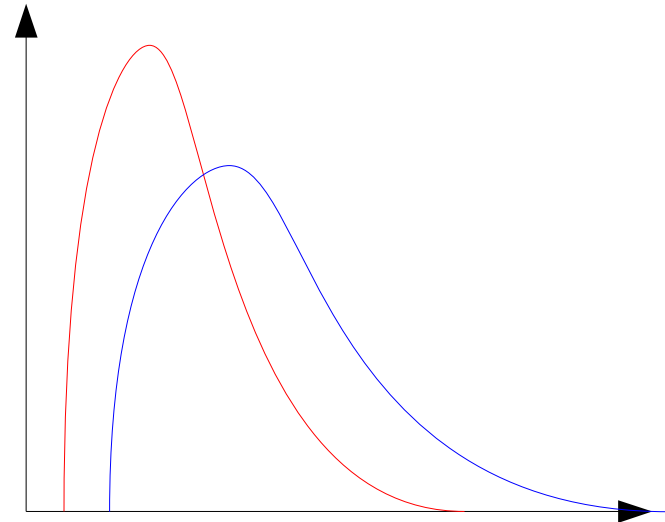
Test statistique

- ▶ Déterminer si deux échantillons proviennent de la même distribution
- ▶ Déterminer si deux méthodes se comportent différemment
 - ▶ Et donc, si l'un est meilleur que l'autre
- ▶ Hypothèse « nulle » : les deux sont identiques
 - ▶ Acceptée : on ne sait pas faire la différence
 - ▶ Rejetée : ils sont différents
 - ▶ **Avec une certaine probabilité de se tromper**



Caractéristiques

- ▶ Distributions non normales, mais similaires
- ▶ Échantillons indépendants
- ▶ Nombre de points faible
- ▶ Variables continues



Procédure

- ▶ Calcul du test
- ▶ Hypothèse nulle rejetée avec $p > P_{seuil}$ → différents
- ▶ Au mieux, $P_{seuil} = 0.95$







**Trouver la forme d'un morceau de papier
qui met le plus de temps possible à tomber.**

- ▶ Peter Bentley - <http://www.cs.ucl.ac.uk/staff/P.Bentley/>



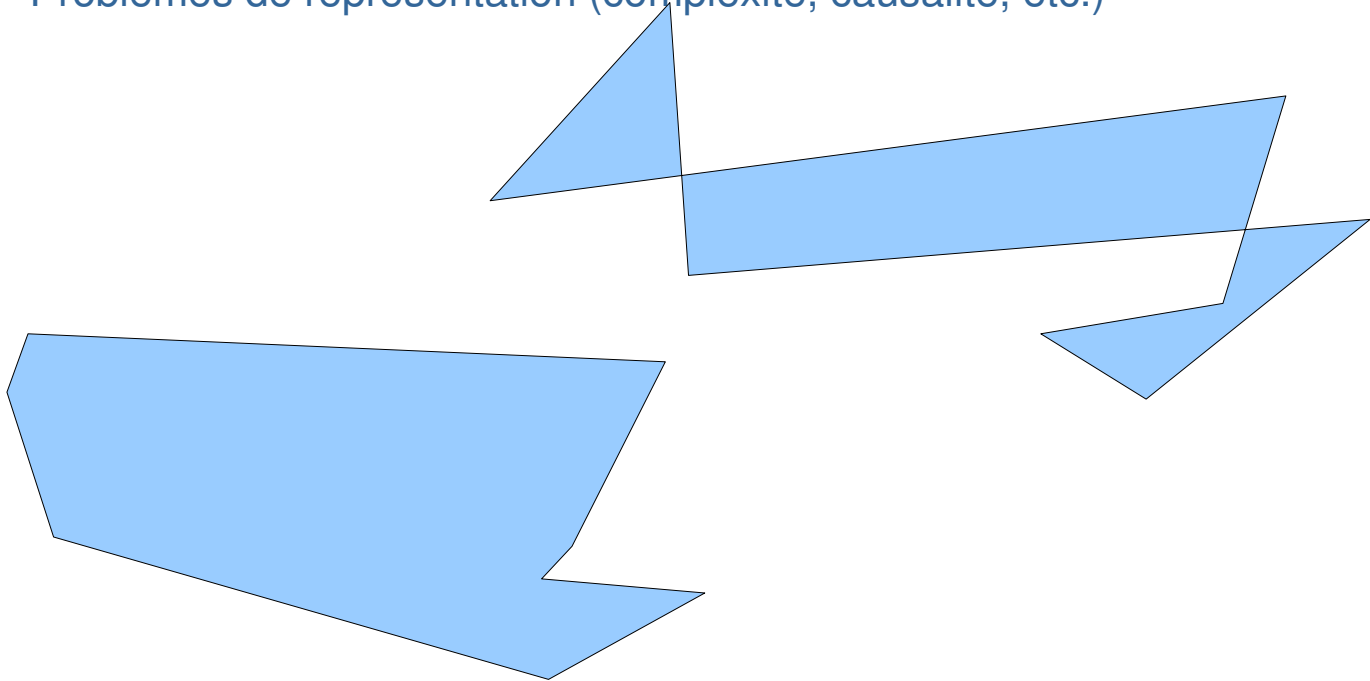
Matériel :

- ▶ Ramette A4
- ▶ Ciseaux
- ▶ Bac à sable A4
- ▶ ~ 10 caillous numérotés
- ▶ Chronomètre
- ▶ Une fléchette



Algorithme

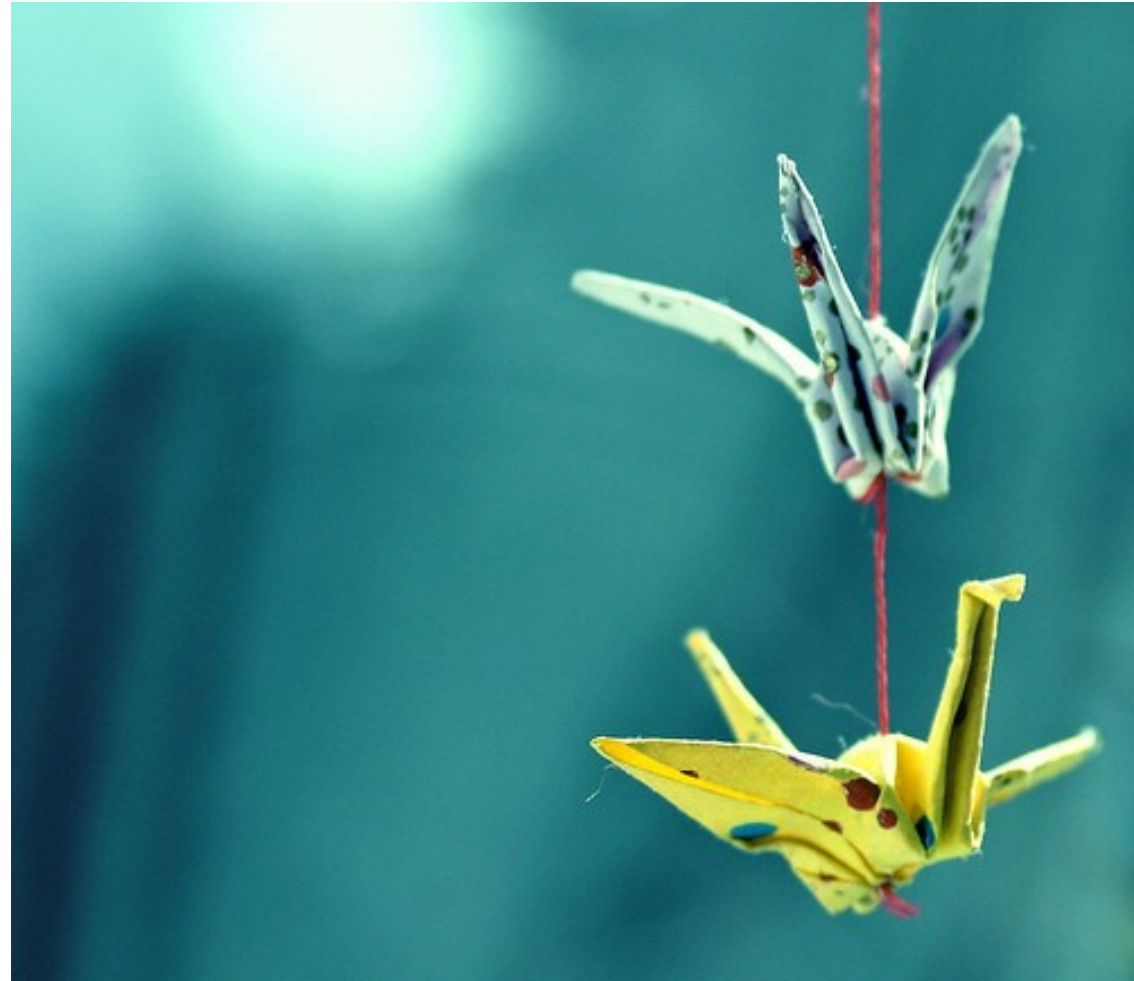
- ▶ Initialisation
 - ▶ Lancer les cailloux dans le bac
 - ▶ Reporter leur position sur la feuille
 - Problèmes de représentation (complexité, causalité, etc.)





Évaluation

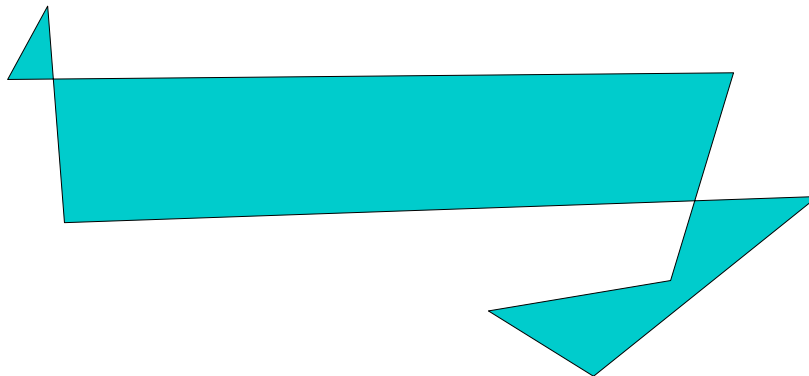
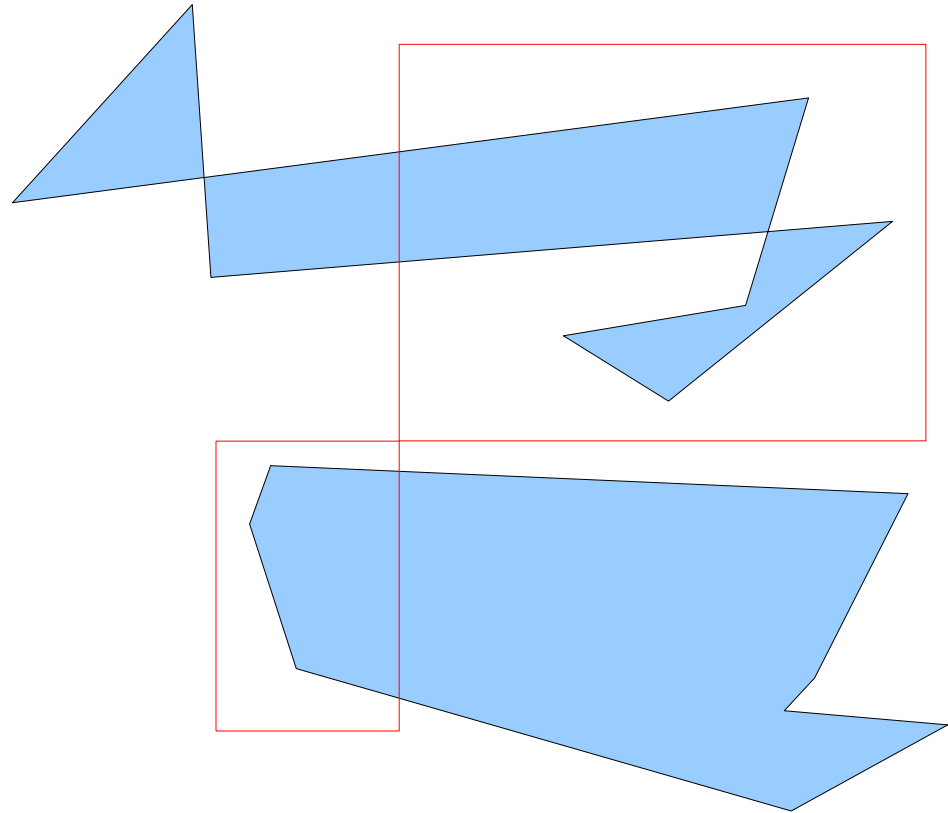
- ▶ Découper
- ▶ Chronométrer 5 lancés





Croisement

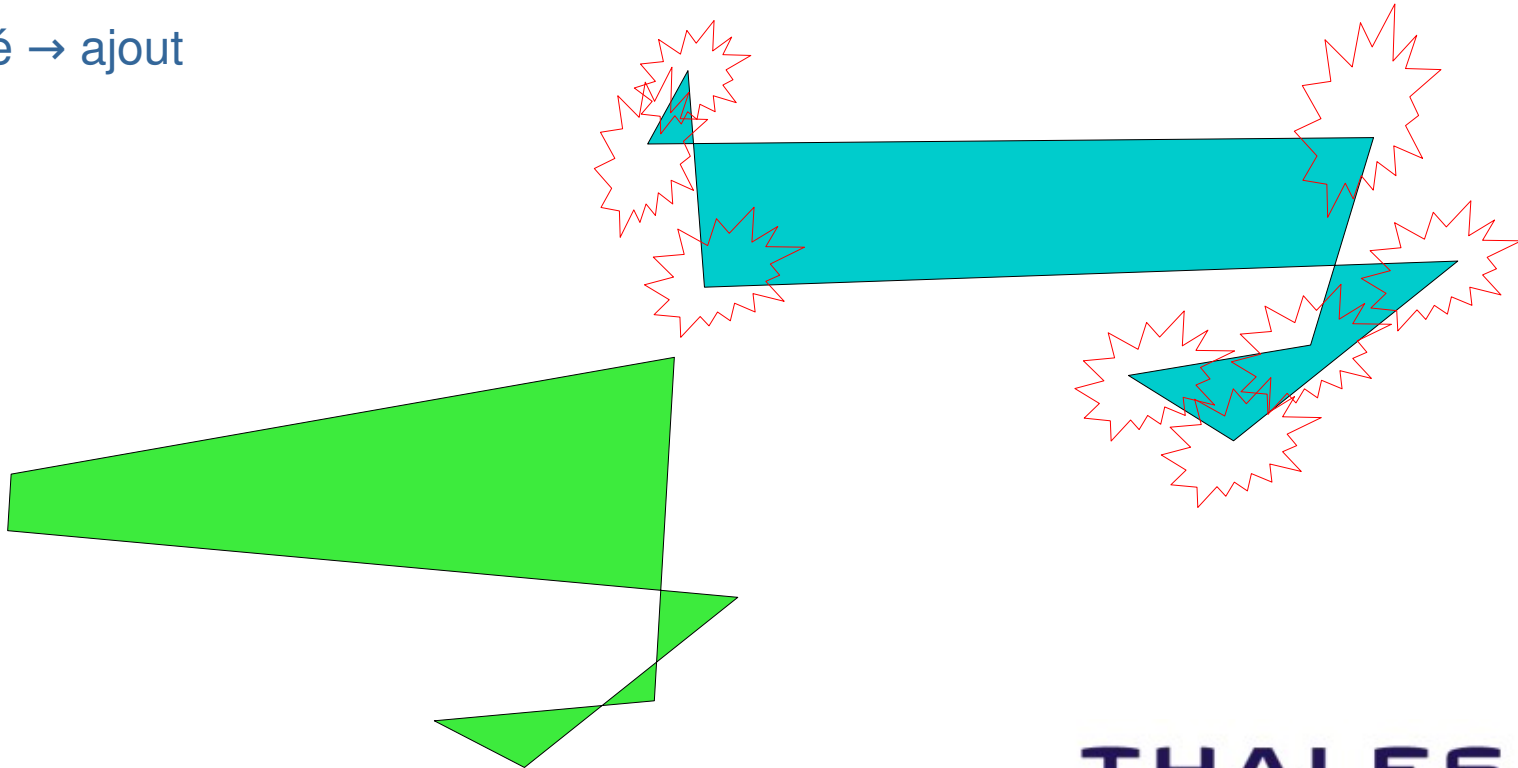
- ▶ Intervertir des points





Mutation

- ▶ Lancé de fléchette sur chaque point
- ▶ Déplacer le point
 - ▶ Moins d'un centimètre → suppression
 - ▶ Raté → ajout



Remplacement

- ▶ Déchirer les formes les plus rapides

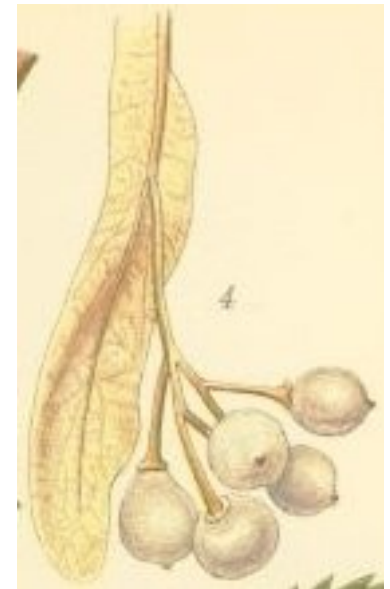
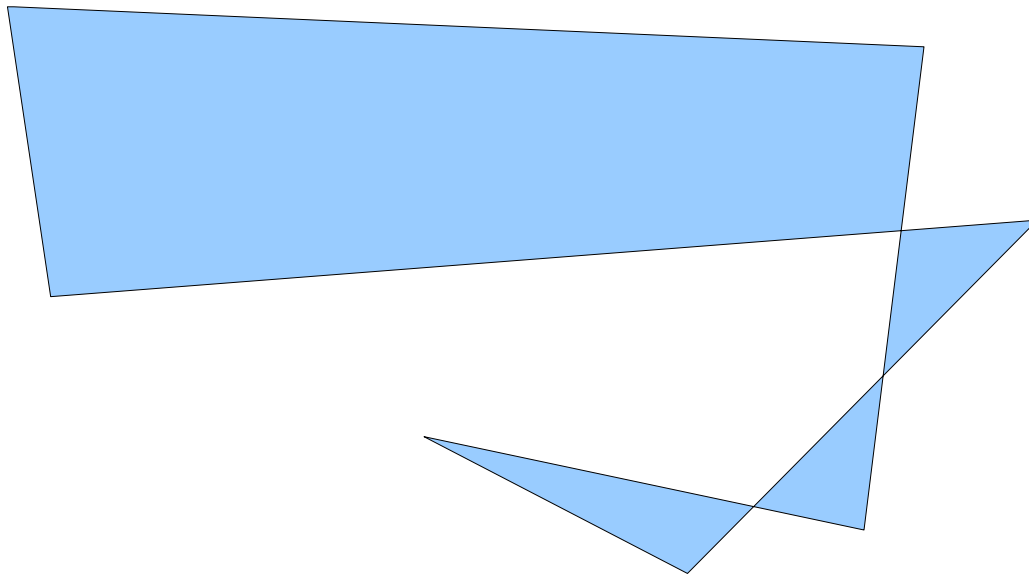
<http://www.flickr.com/photos/vorty/1409444890/>





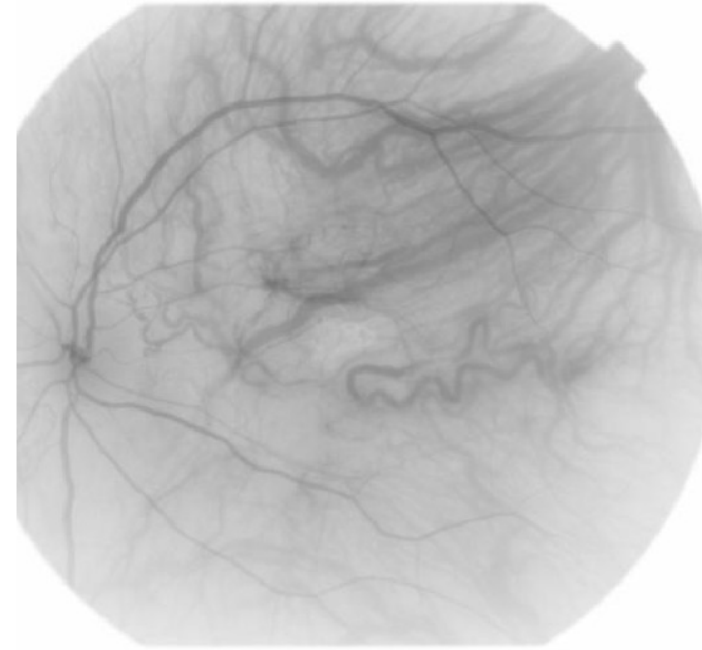
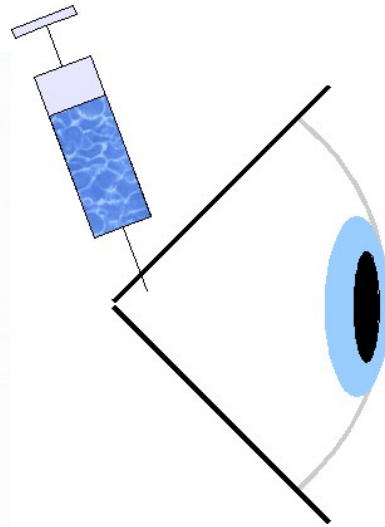
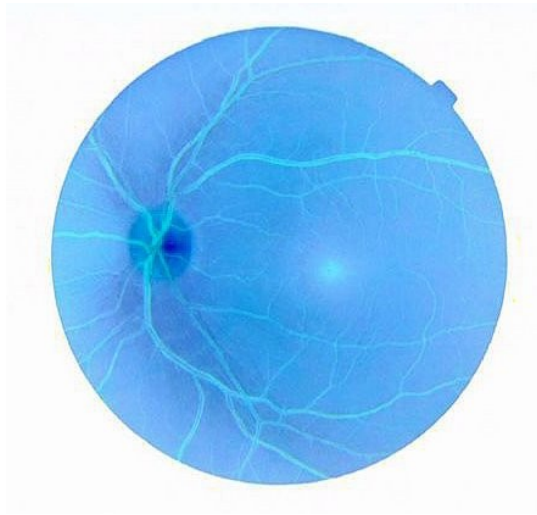
Optimum

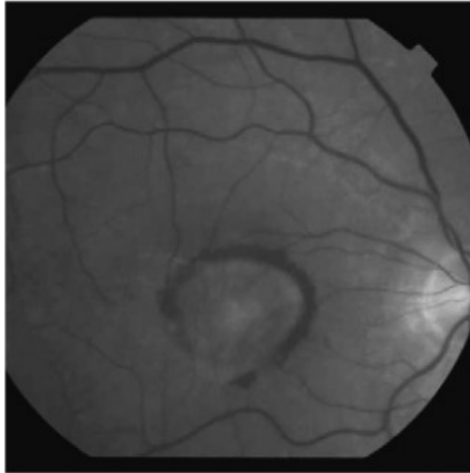
- ▶ 5 formes
- ▶ 10 générations



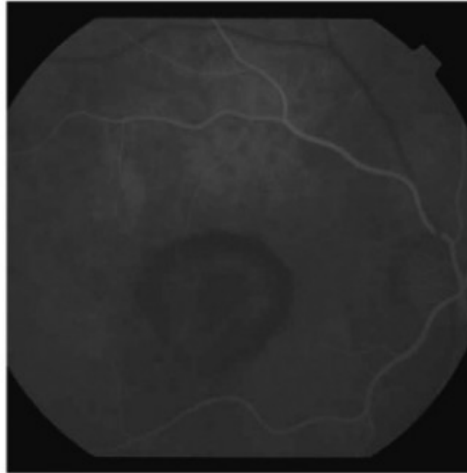


Angiographie du fond de l'oeil

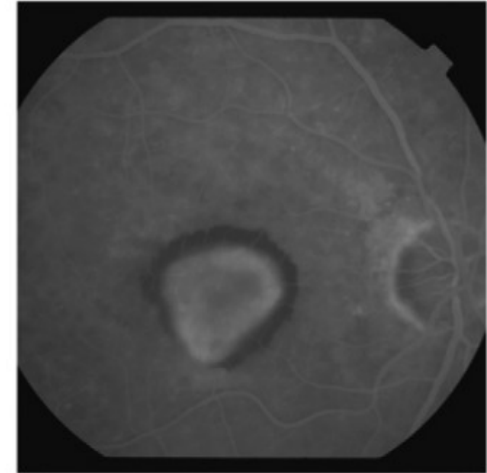




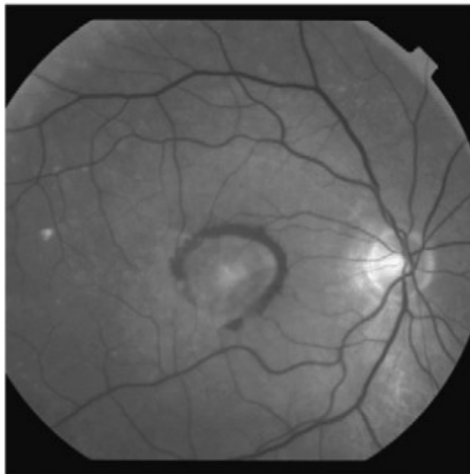
(a) original fluorescein image before injection



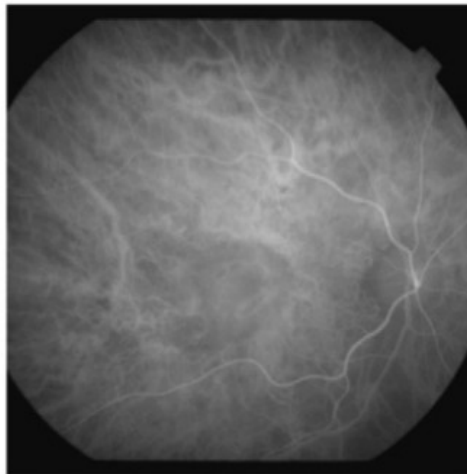
(b) original fluorescein image in the arterial phase



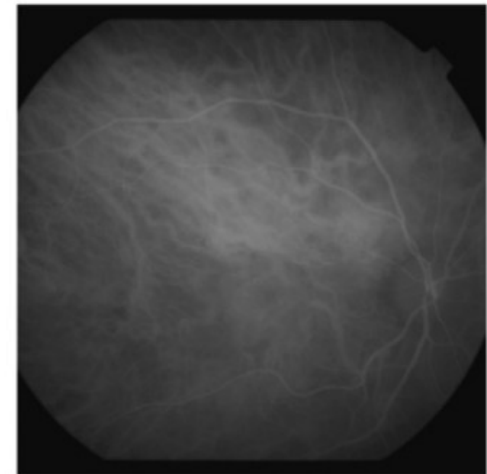
(c) original fluorescein image in the venous phase



(d) original ICG image before injection



(e) original ICG image in the arterial phase



(f) original ICG image in the venous phase



Variables :

- ▶ vertical
- ▶ horizontal
- ▶ Rotation
- ▶ zoom

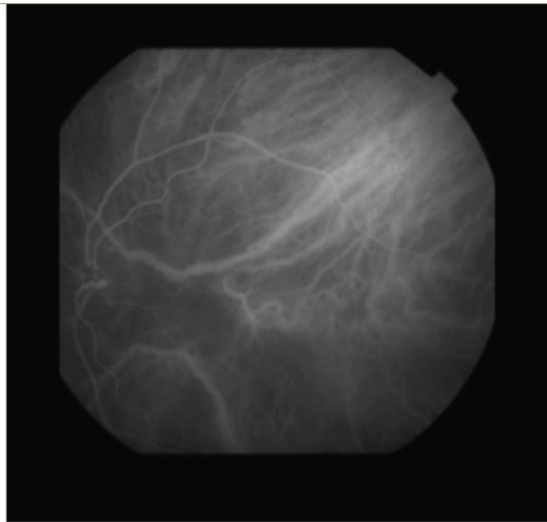
Objectif :

- ▶ Minimiser différence

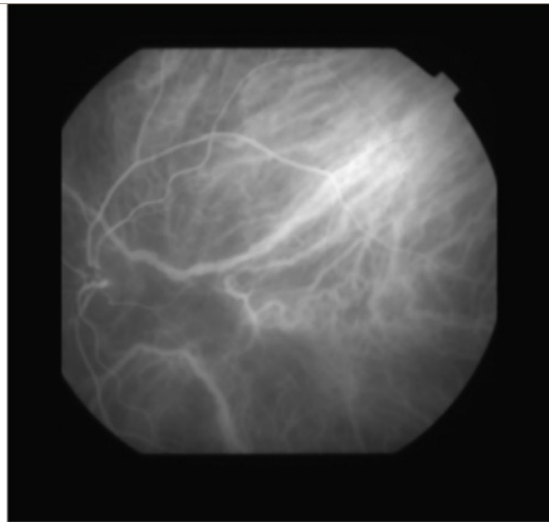
$$\frac{\sum_{(i,j)=(1,1)}^{(l,h)} |I_1(i,j) - I_2(i,j)|}{l \cdot h}$$

Difficulté :

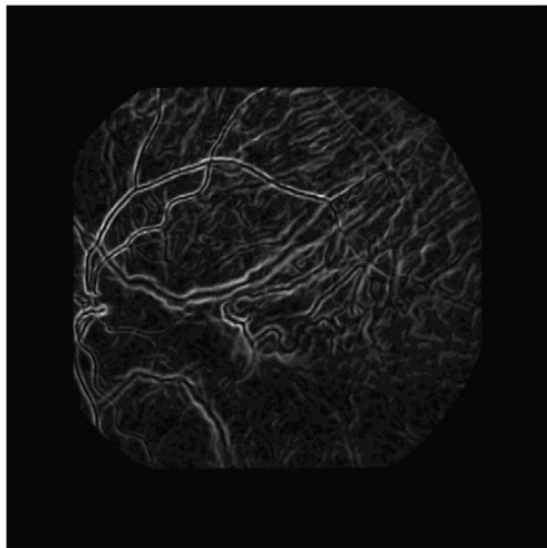
- ▶ Information pertinente
- ▶ Résolution



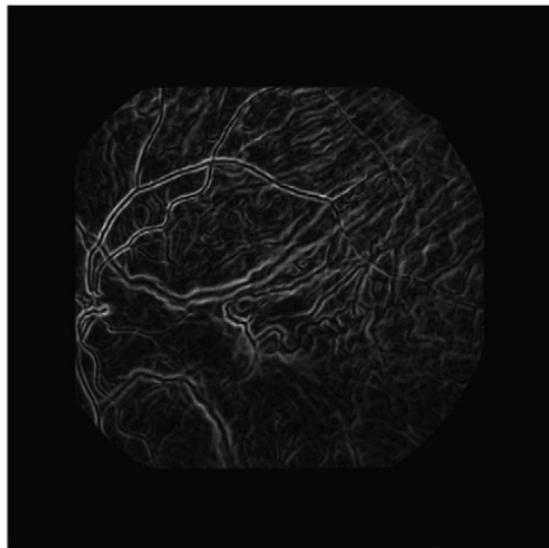
(a) Original image



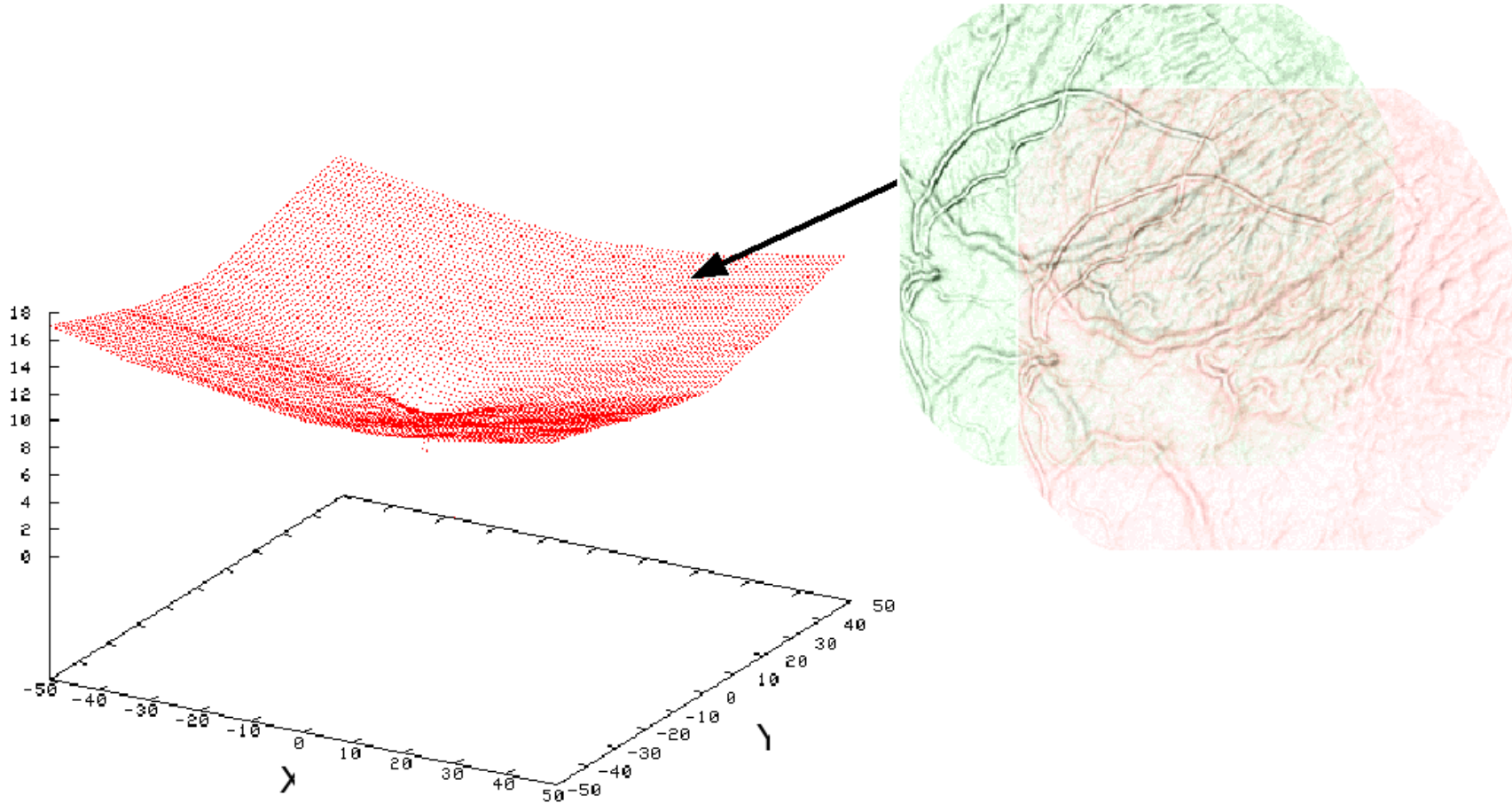
(b) Wiener filtered image

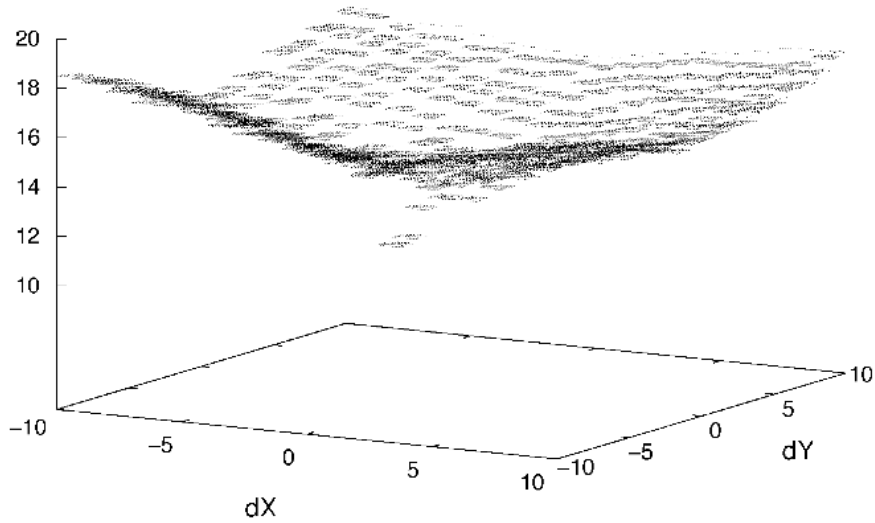


(c) Gradient image

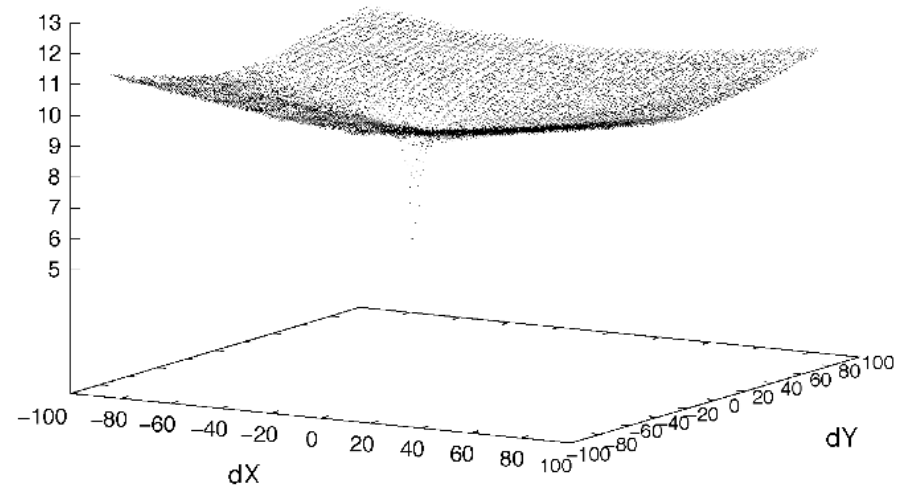


(d) Filtered gradient image

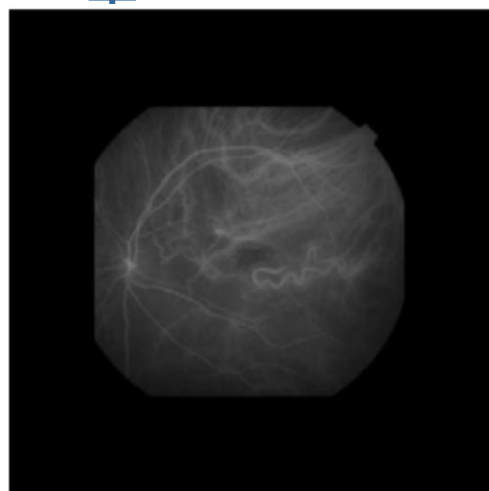




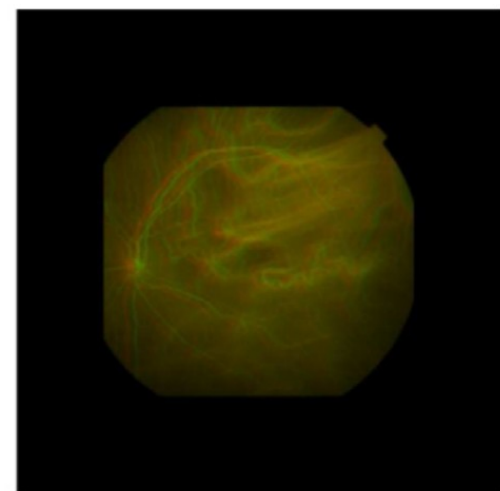
(a) low resolution



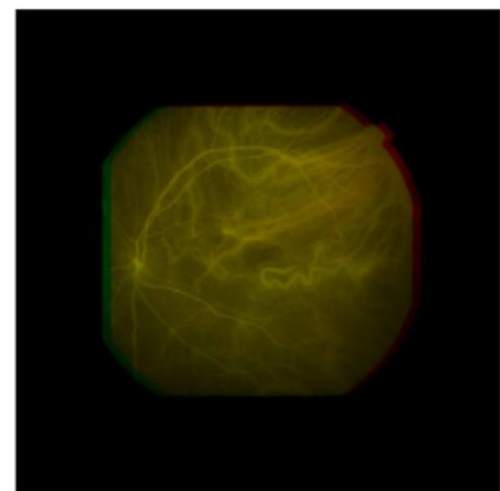
(b) high resolution



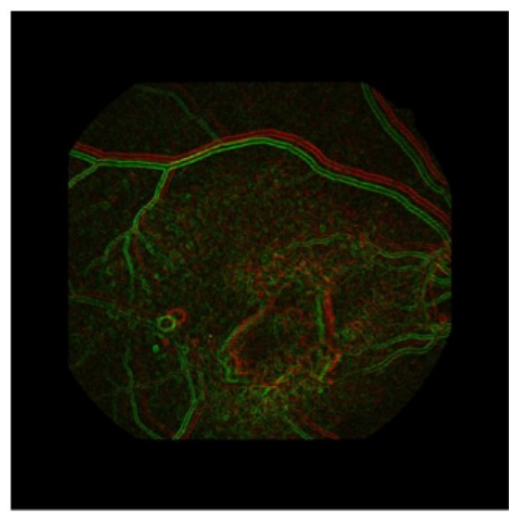
(a) First ICG image



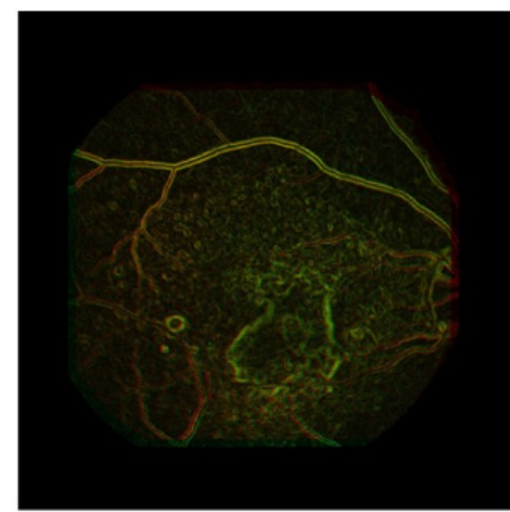
(b) Two not registered ICG images



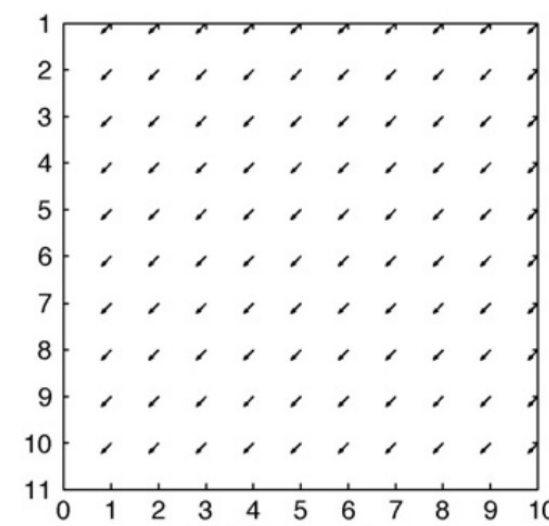
(c) Registered ICG images



(a) not registered



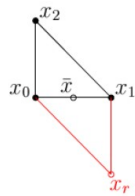
(b) registered



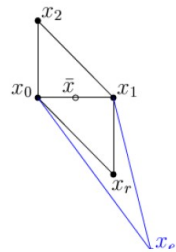
(c) rigid transform

Colonies de fourmis

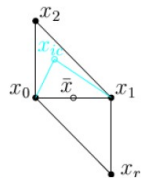
- ▶ Type multi-agents
- ▶ Hybridation Nelder-Mead



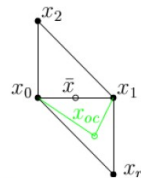
a) Reflexion



b) Expansion



c) Inside contraction

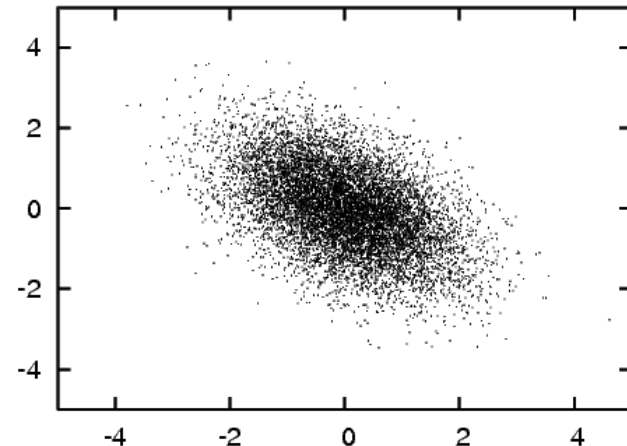


d) Outside contraction

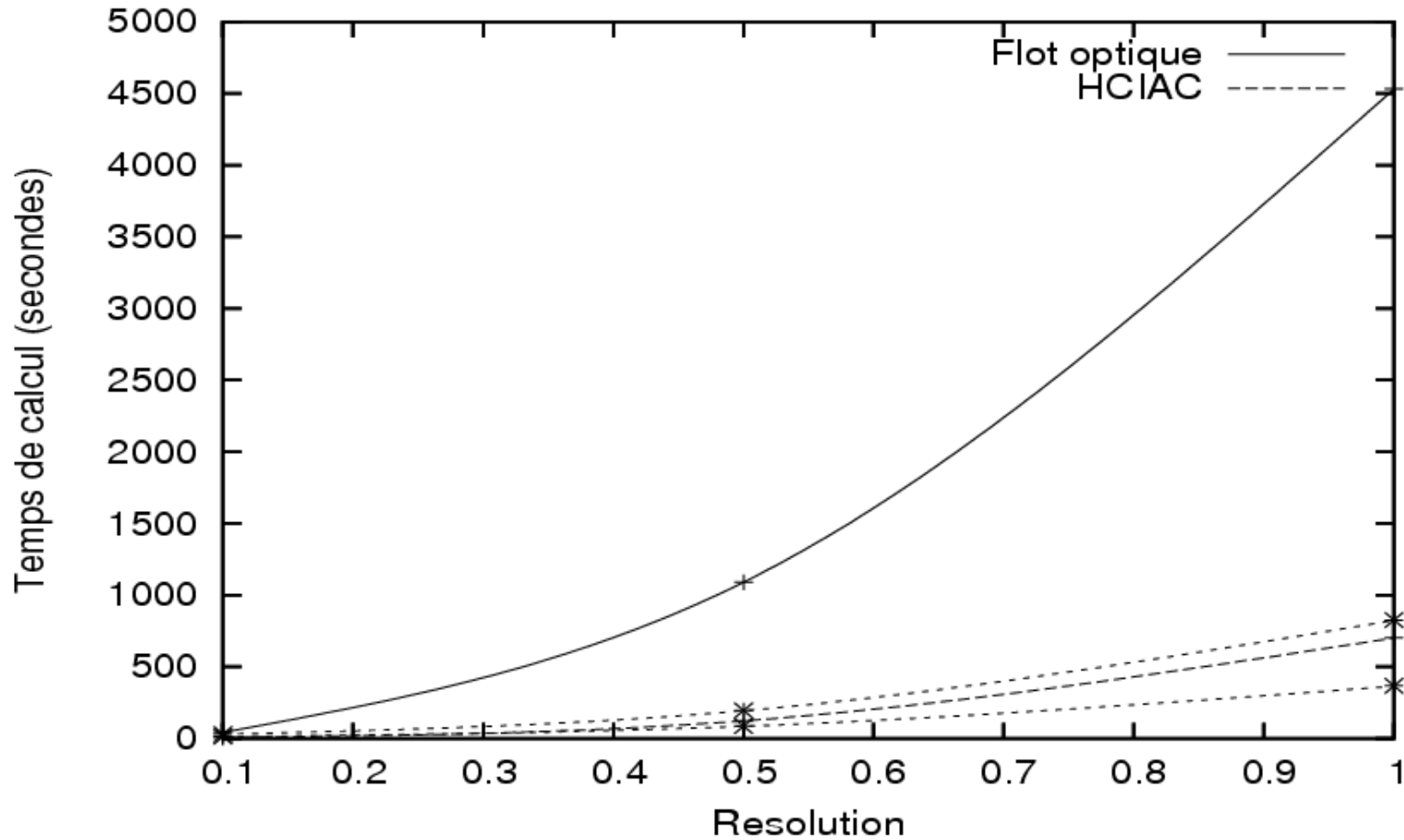
- ▶ Modélisation simplifiée
- ▶ Paramétrage difficile

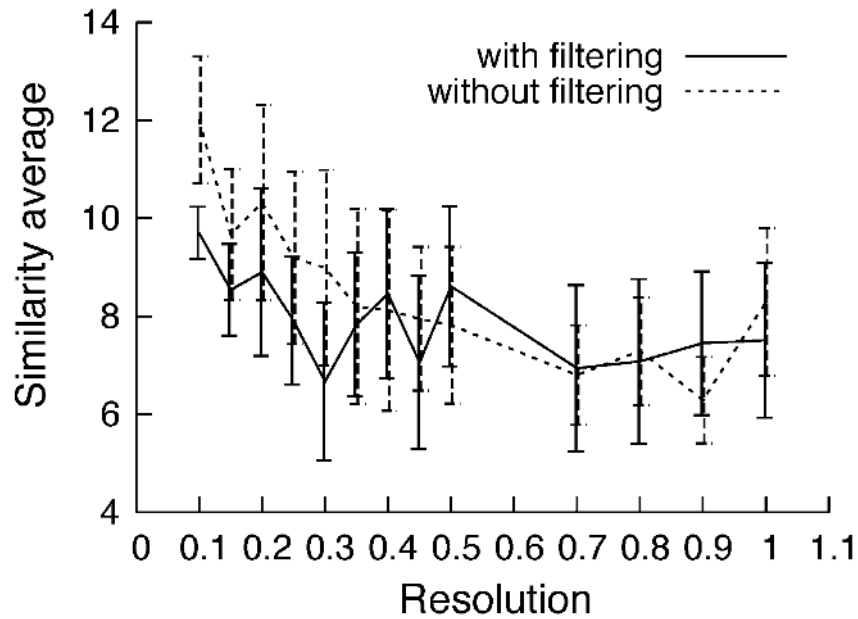
Estimation de distribution

- ▶ Gaussienne multi-variante + NMS
- ▶ Initialisation ad-hoc

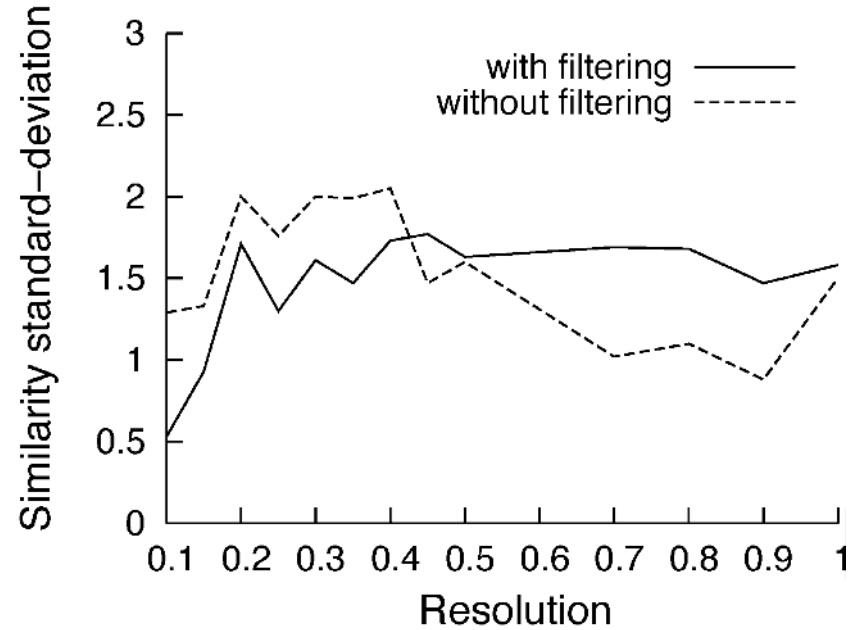


- ▶ Choix distribution délicat
- ▶ Paramétrage intuitif

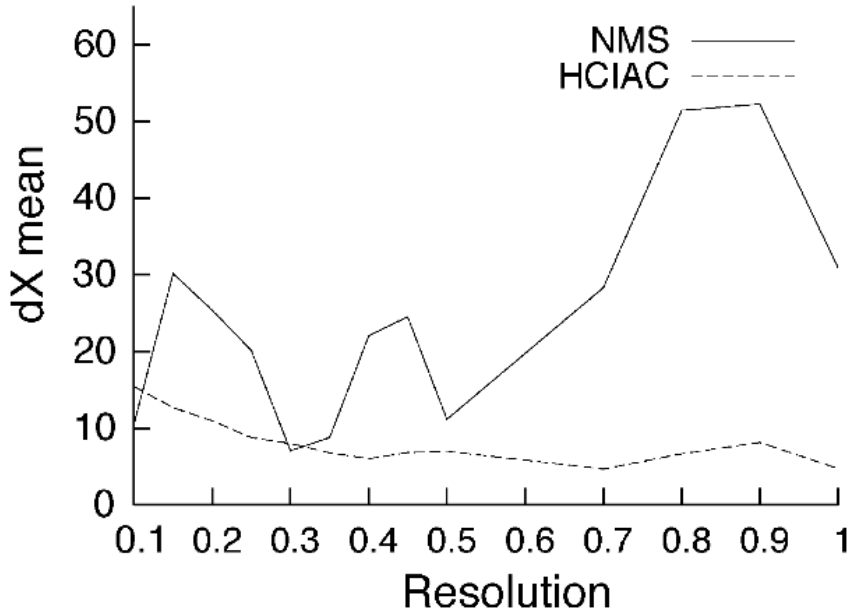




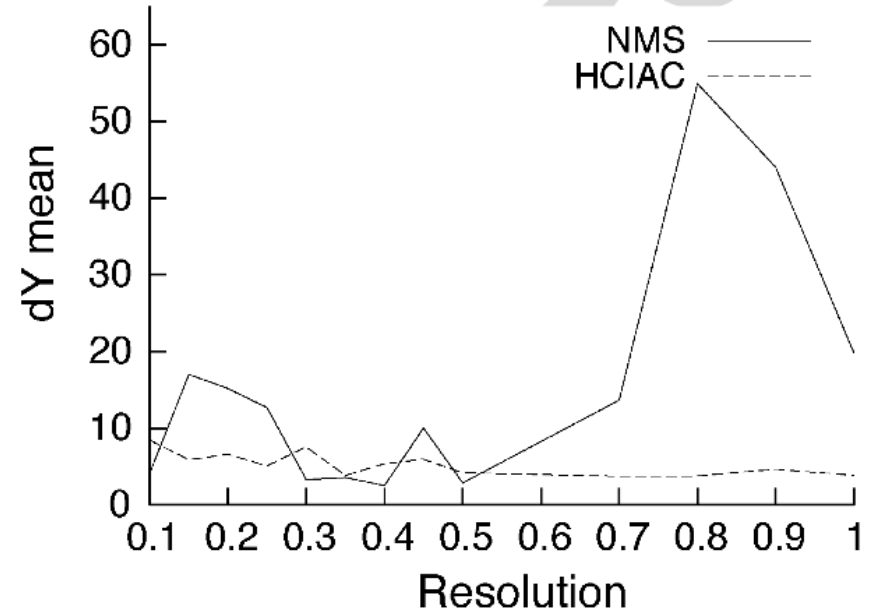
(a) average and standard deviation



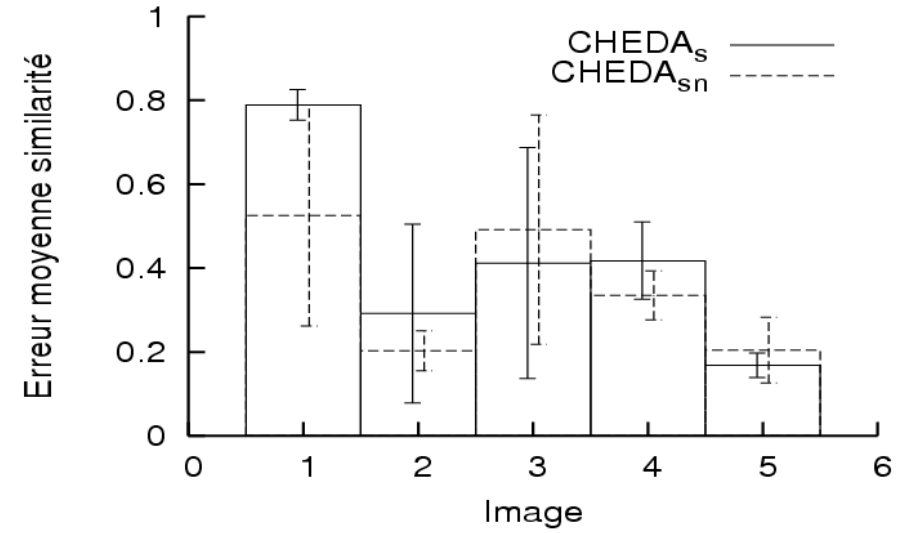
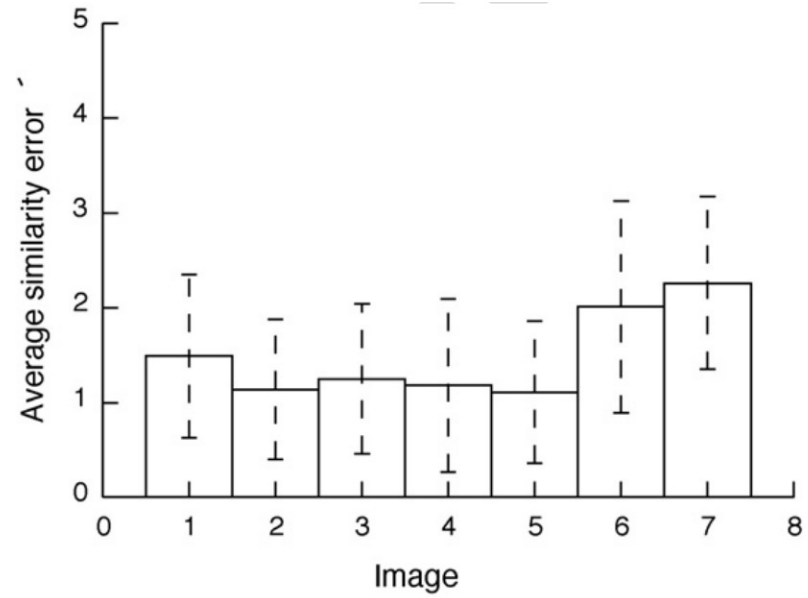
(b) standard deviation only



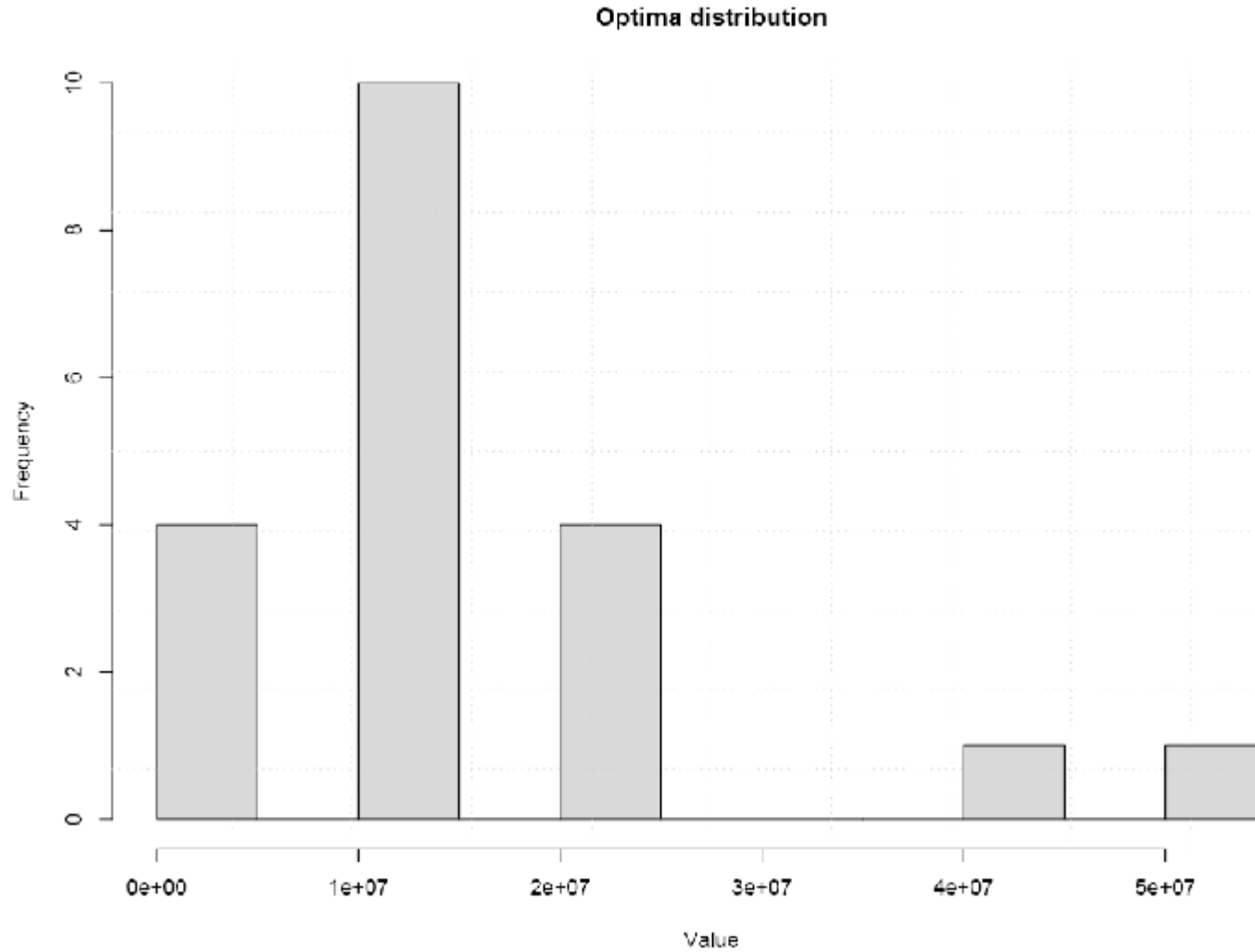
(a) horizontal displacement



(b) vertical displacement



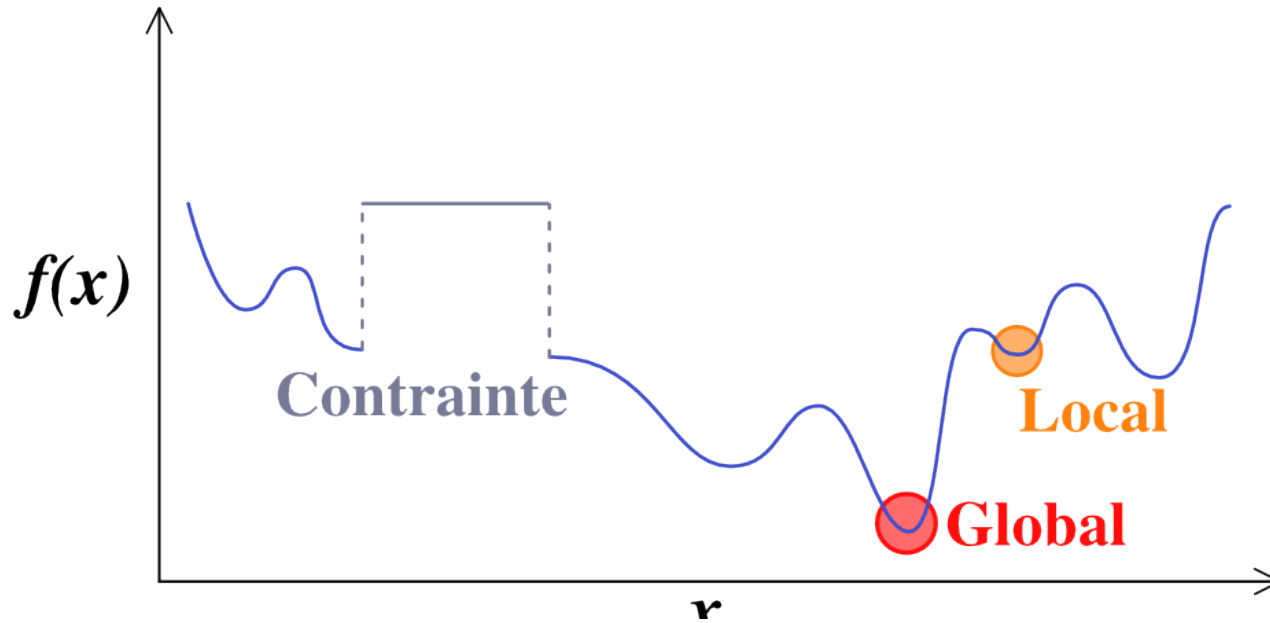
(b) CHEDA



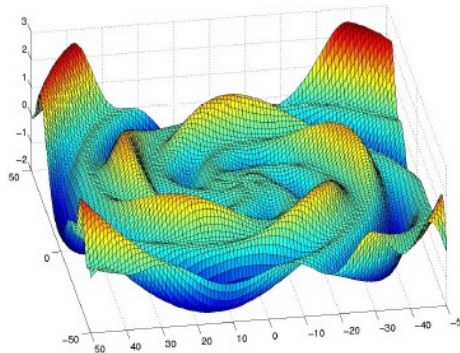


+ Z
+ T
+ déformations non-linéaires
etc.

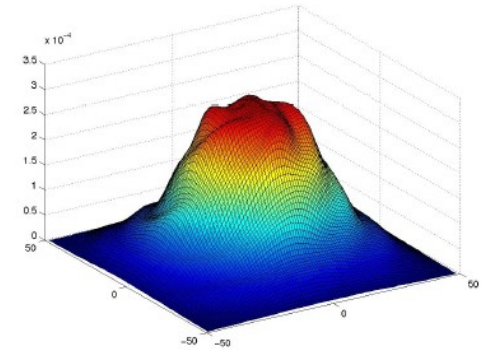




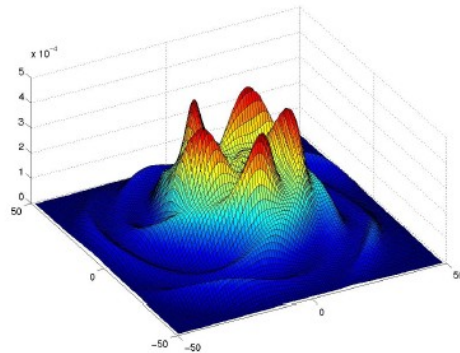
- ▶ Global
- ▶ Généraliste
- ▶ Stochastiques
- ▶ Facile à implémenter



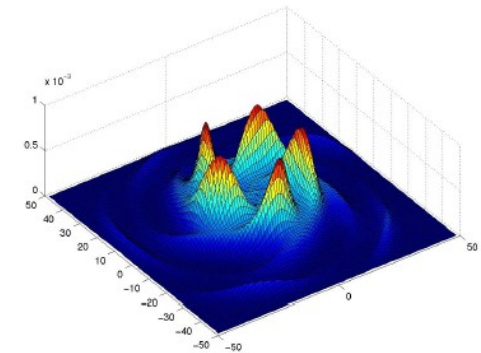
(a) Cost function



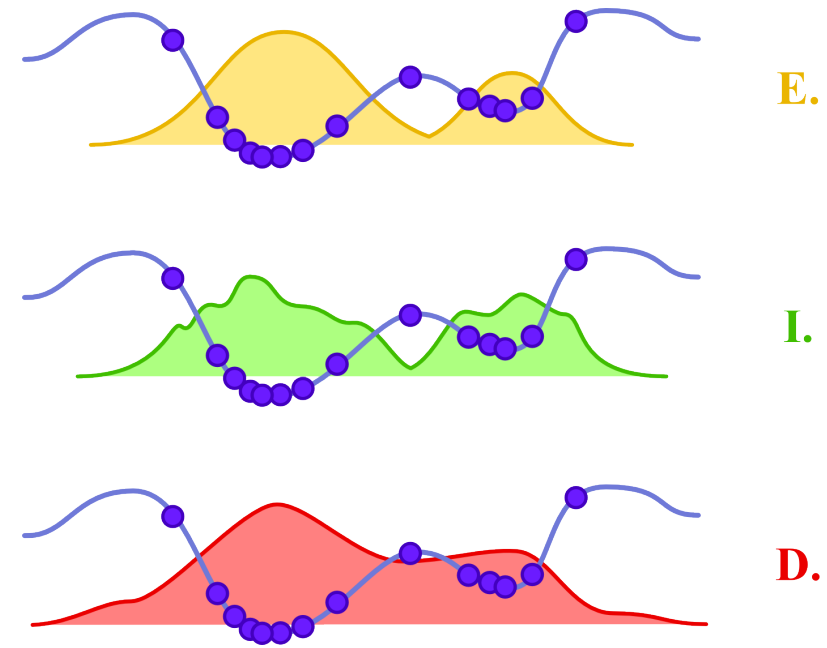
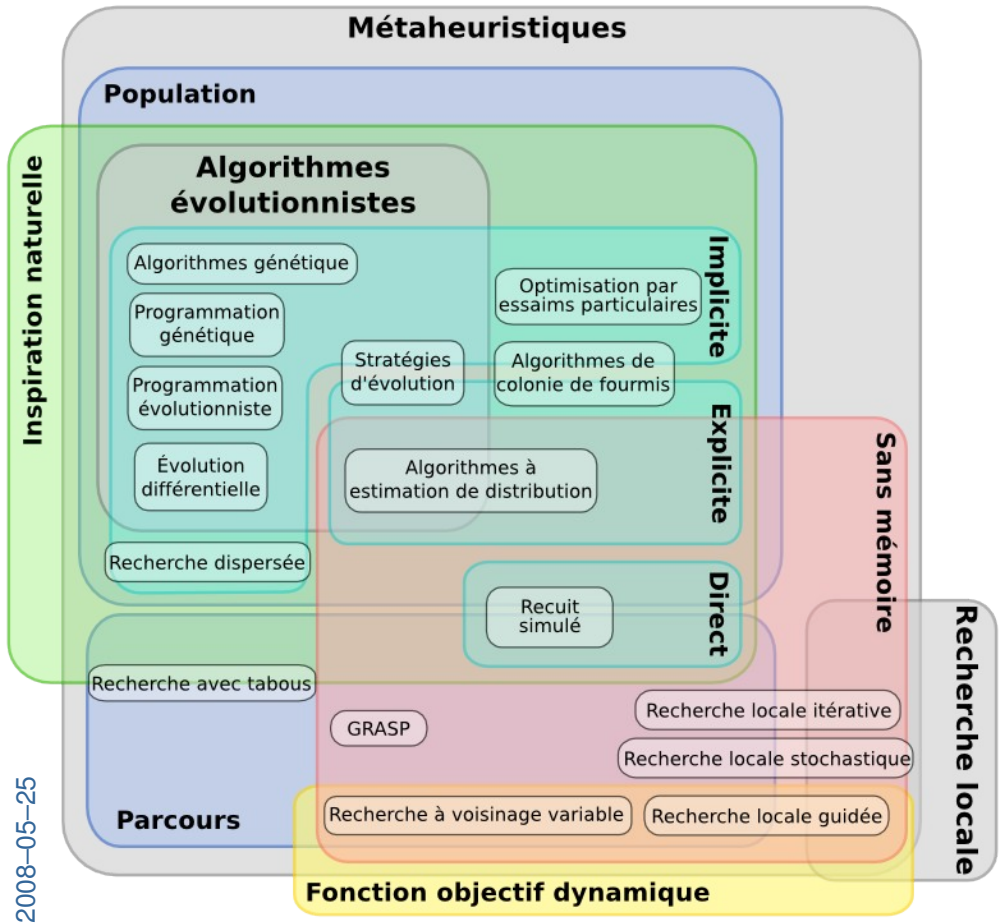
(b) $t=5$

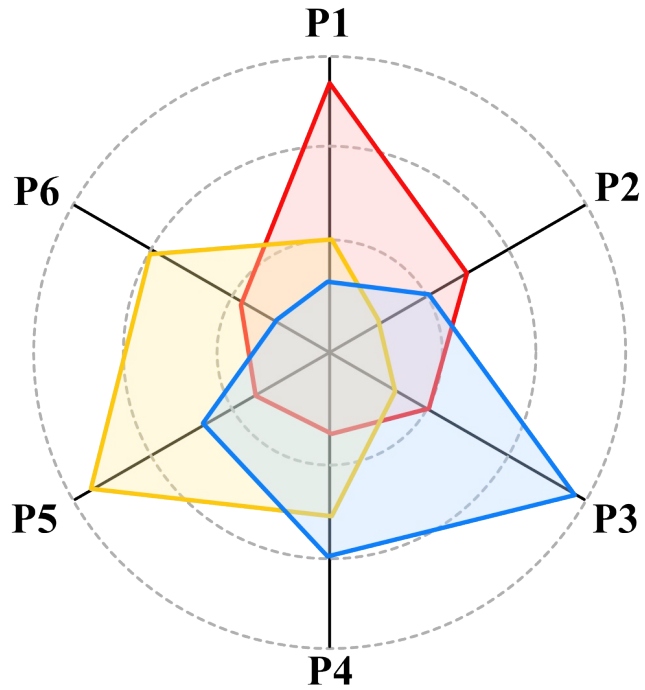


(c) $t=1$

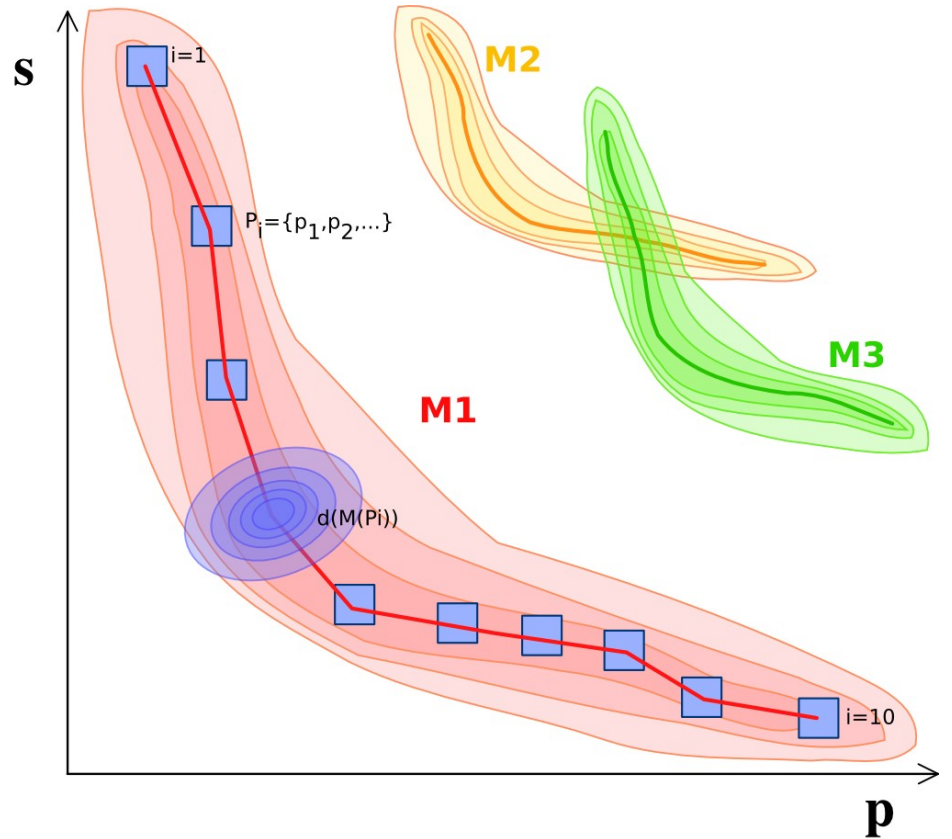


(d) $t=0.5$





- M1 
- M2 
- M3 





- (fr) Teghem & Pilot, **Optimisation approchée en recherche opérationnelle**, Hermes, 2002. ISBN 2746204622
- (fr) Dréo, Petrowski, Taillard & Siarry, **Métaheuristiques pour l'optimisation difficile**, Eyrolles, 2003, ISBN 2-212-11368-4.
- (en) Blum & Roli, **Metaheuristics in combinatorial optimization**: Overview and conceptual comparison, ACM Computing Surveys, volume 35, numéro 3, septembre 2003, pages 268-308. DOI:10.1145/937503.937505
- (en) Collet & Rennard, **Introduction to Stochastic Optimization Algorithms**, IDEA Group Inc, 2006, ISBN 1591409845.
- (en) Bartz-Beielstein, **Experimental Research in Evolutionary Computation**: The New Experimentalism, Springer, 2006, ISBN: 978-3-540-32026-5.



johann.dreo@thalesgroup.com

- ▶ Questions
- ▶ Stages
- ▶ Thèses
- ▶ ...