

Optimisation stochastique - 1

Le recuit simulé



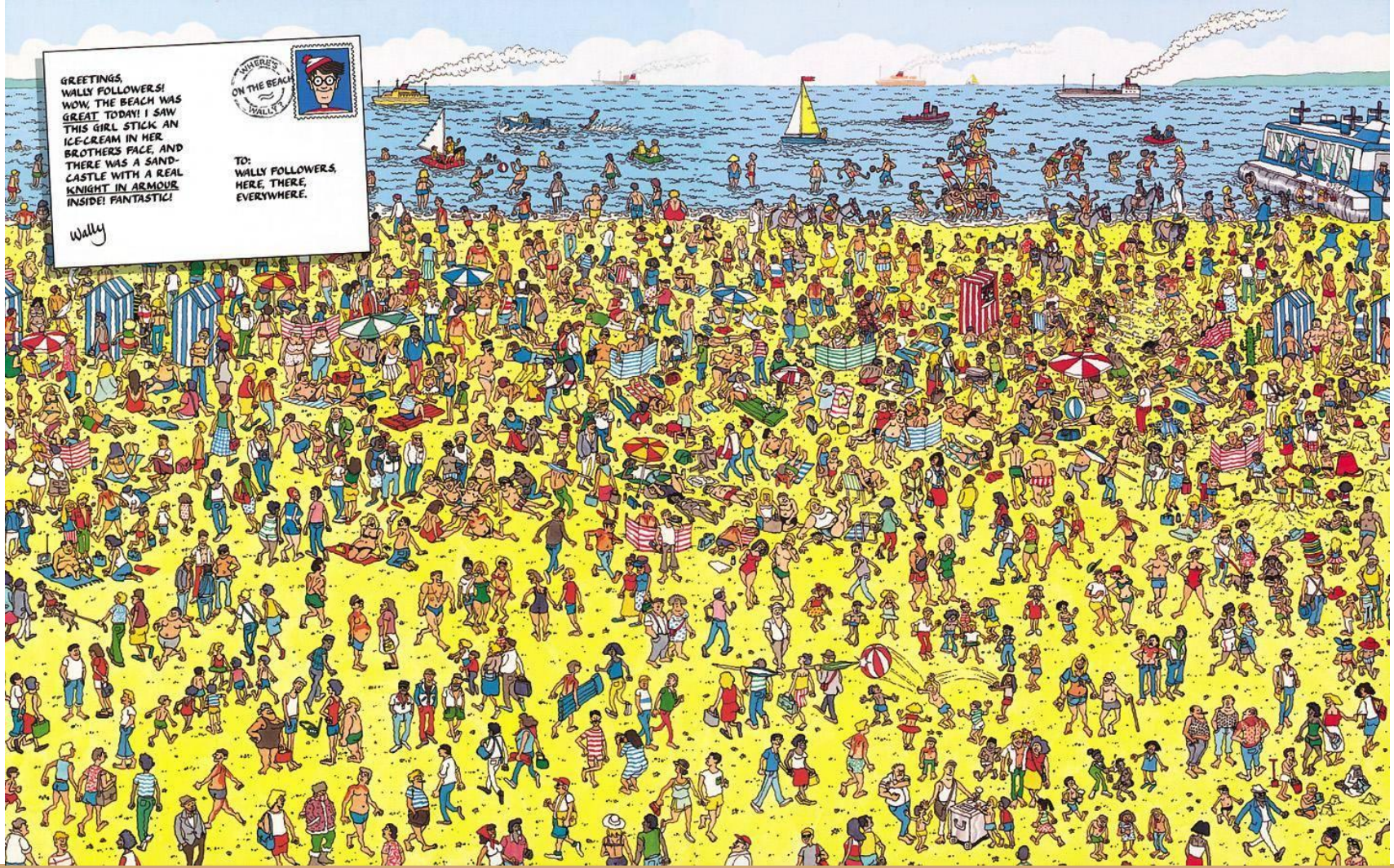
Optimisation stochastique

GREETINGS,
WALLY FOLLOWERS!
WOW, THE BEACH IS
GREAT TODAY! I SAW
THIS GIRL STICK AN
ICE-CREAM IN HER
BROTHERS FACE, AND
THERE WAS A SAND-
CASTLE WITH A REAL
KNIGHT IN ARMOUR
INSIDE! FANTASTIC!

Wally



TO:
WALLY FOLLOWERS,
HERE, THERE,
EVERYWHERE.

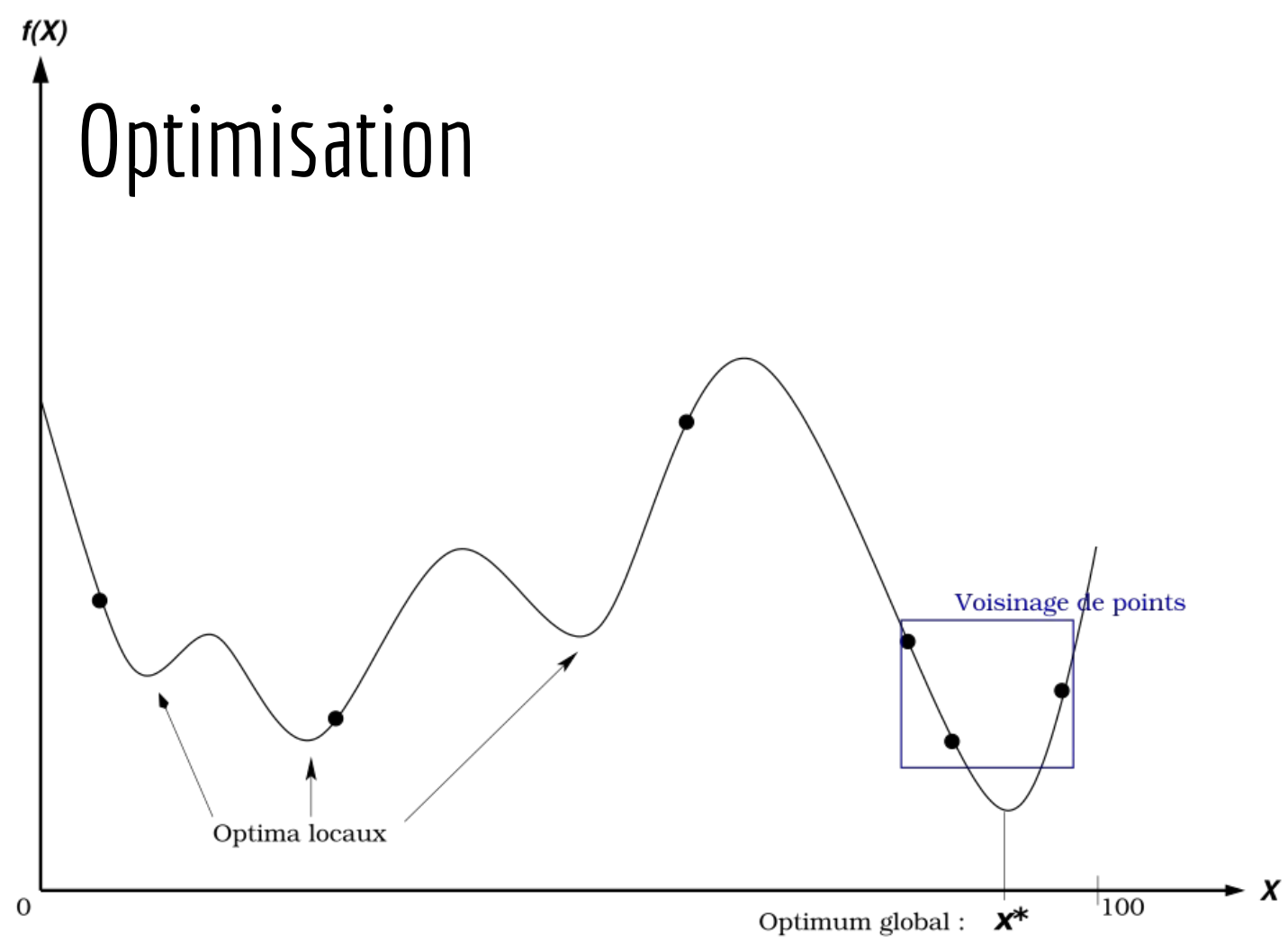


Optimisation

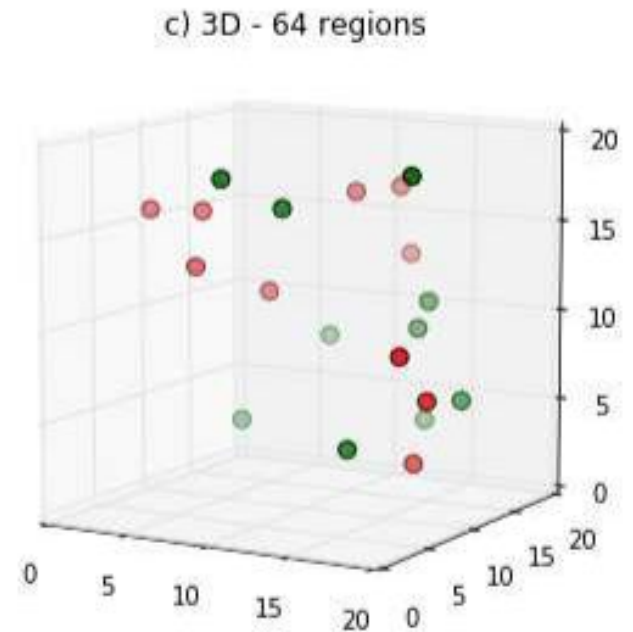
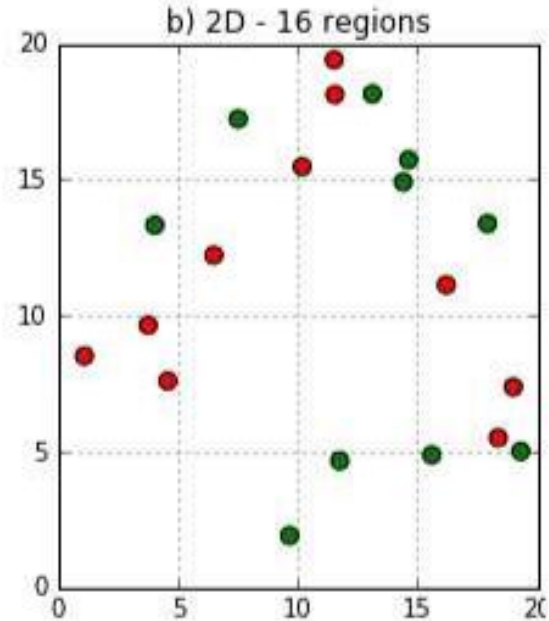
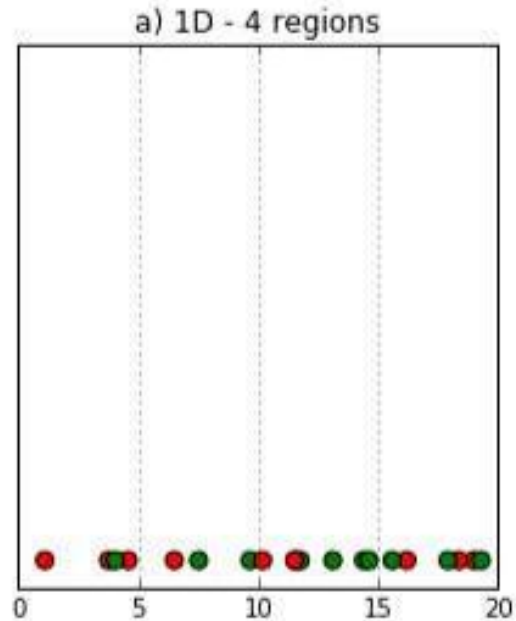


$f(x)$

Optimisation



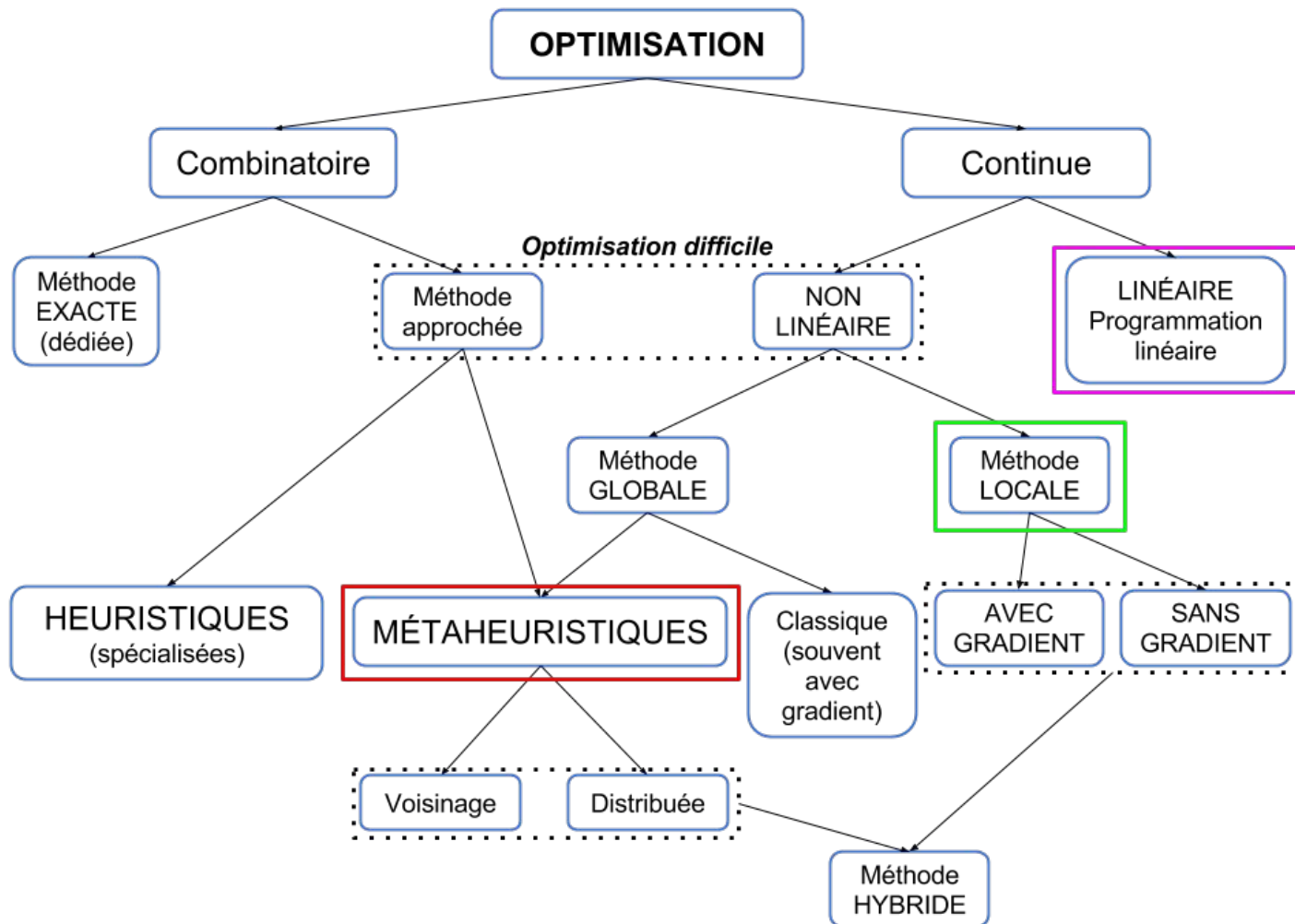
La malédiction de la dimension



<http://cleverowl.uk/2016/02/06/curse-of-dimensionality-explained/>

Cours science des données de Stéphane Mallat

L'apprentissage face à la malédiction de la grande dimension

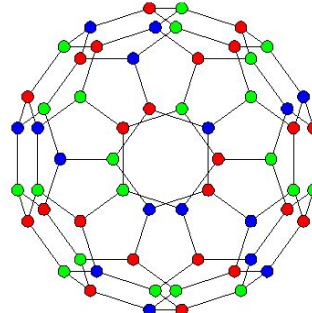


Typologie de problèmes

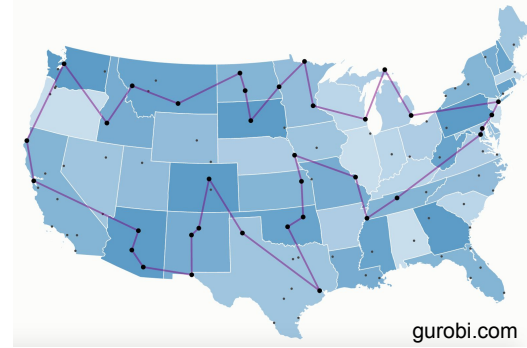
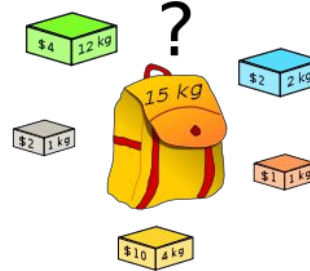
- Optimisation difficile ;
- Problème de minimisation / maximisation ;
- Problèmes mono-objectif, multi-objectifs, “many”-objectifs ;
- Problèmes à variables discrètes / continues / mixtes ;
- Problèmes sous contraintes ;
- Problèmes à grande dimension ;
- Problèmes dynamiques.

Typologie de problèmes à variables discrètes

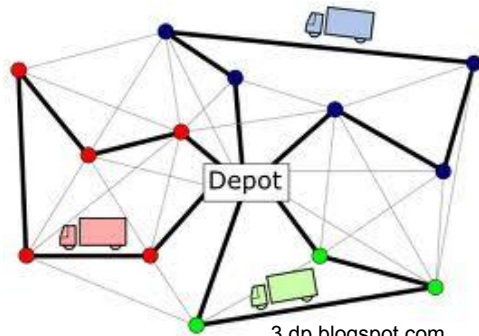
- Voyageur de commerce
- Tournée de véhicules
- Coloration de graphe
- Sac à dos
- Planification...



mathworks.com



gurobi.com

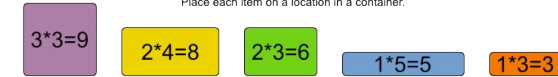


3.dp.blogspot.com

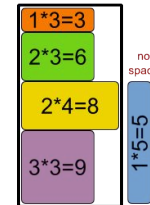
	lundi 25 sept. 2017	marti 26 sept. 2017	mercredi 27 sept. 2017	jeudi 28 sept. 2017	vendredi 29 sept. 2017
09:00 - 12:00	MOPT1 Travaux de signal E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Programmation dynamique E231.404	MOPT1 Optimisation différentielle E231.404	MOPT1 Exercice de complexité E231.404
12:00 - 13:00				MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404
13:00 - 14:00				MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404
14:00 - 15:00	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404
15:00 - 16:00					
16:00 - 17:00	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404	MOPT1 Exercice de complexité E231.404
17:00 - 18:00					
18:00 - 19:00					
19:00 - 20:00					
20:00 - 21:00					
21:00 - 22:00					
22:00 - 23:00					
23:00 - 00:00					

Bin packing

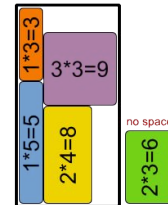
Place each item on a location in a container.



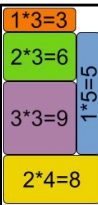
Largest size first



Largest side first

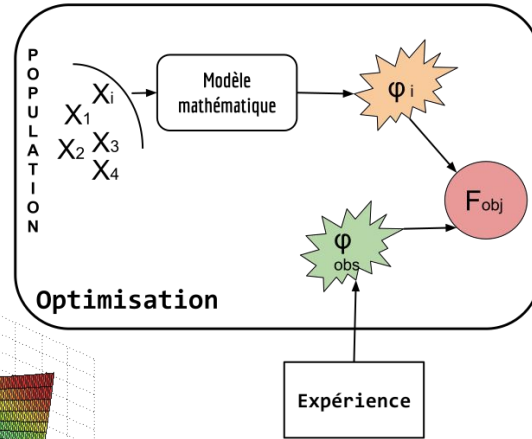
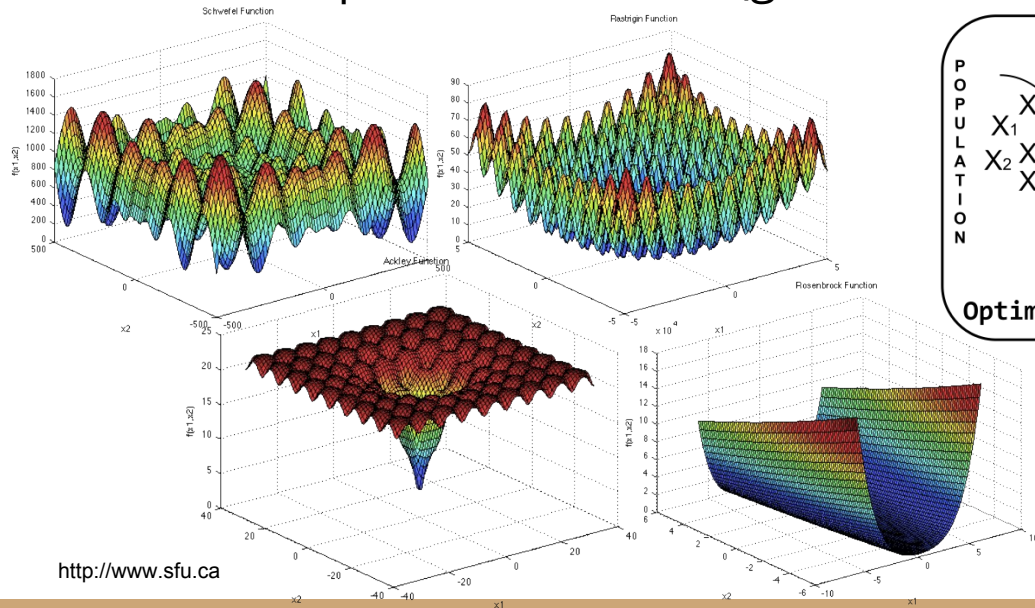


Drools Planner



Typologie de problèmes à variables continues

- Fonction analytique (benchmarks) ;
- Expression non analytique (code de calcul, problème inverse) ;
- Autres problèmes variés (géométrie, chimie, physique)...



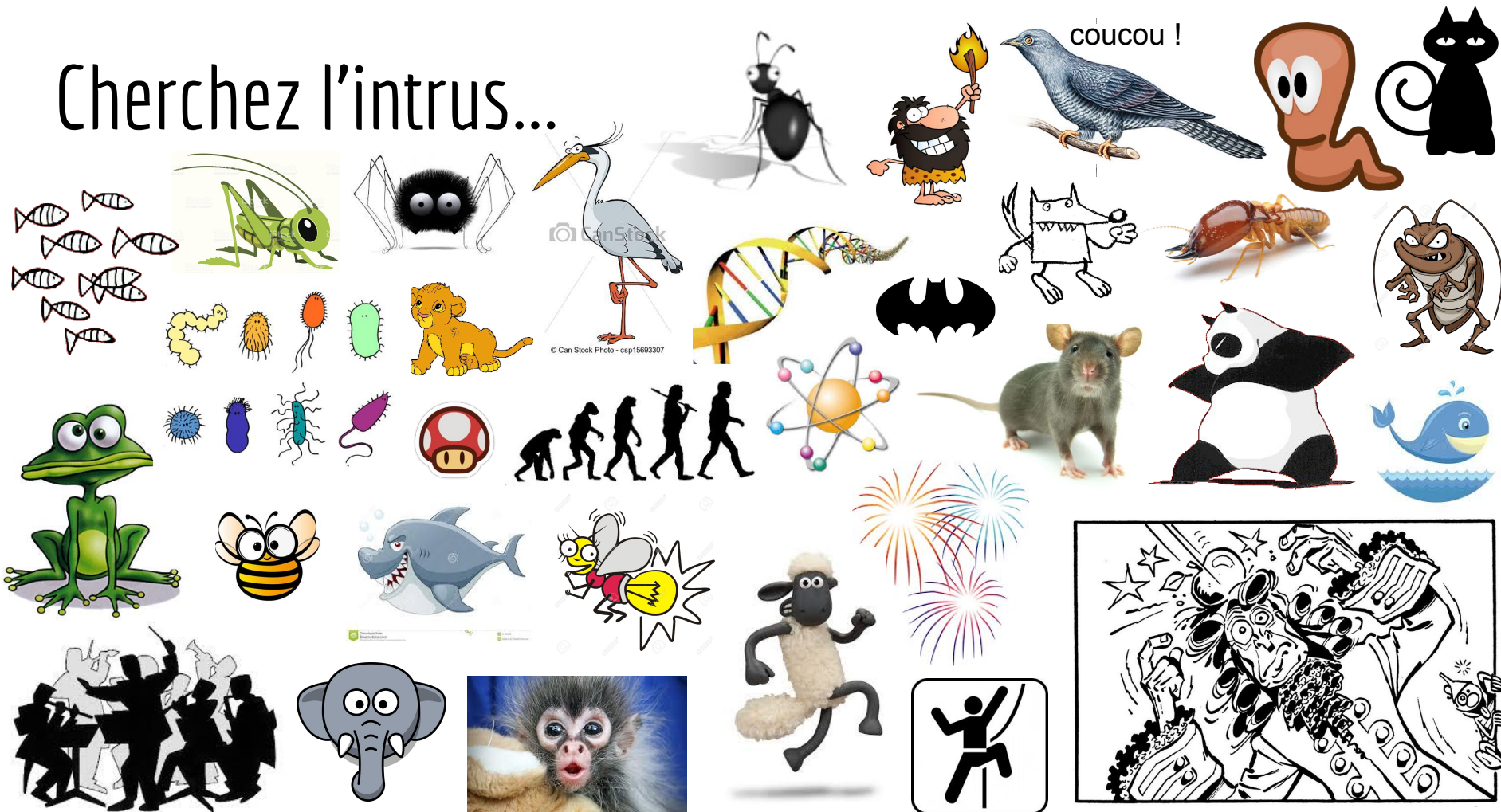
Métaheuristiques

Ce sont des méthodes :

- globales ;
- stochastiques ;
- génériques même si dépendantes du contexte continu/discret ;
- qui ne garantissent pas l'optimalité ;
- basées sur des analogies avec la biologie, la physique, l'éthologie...

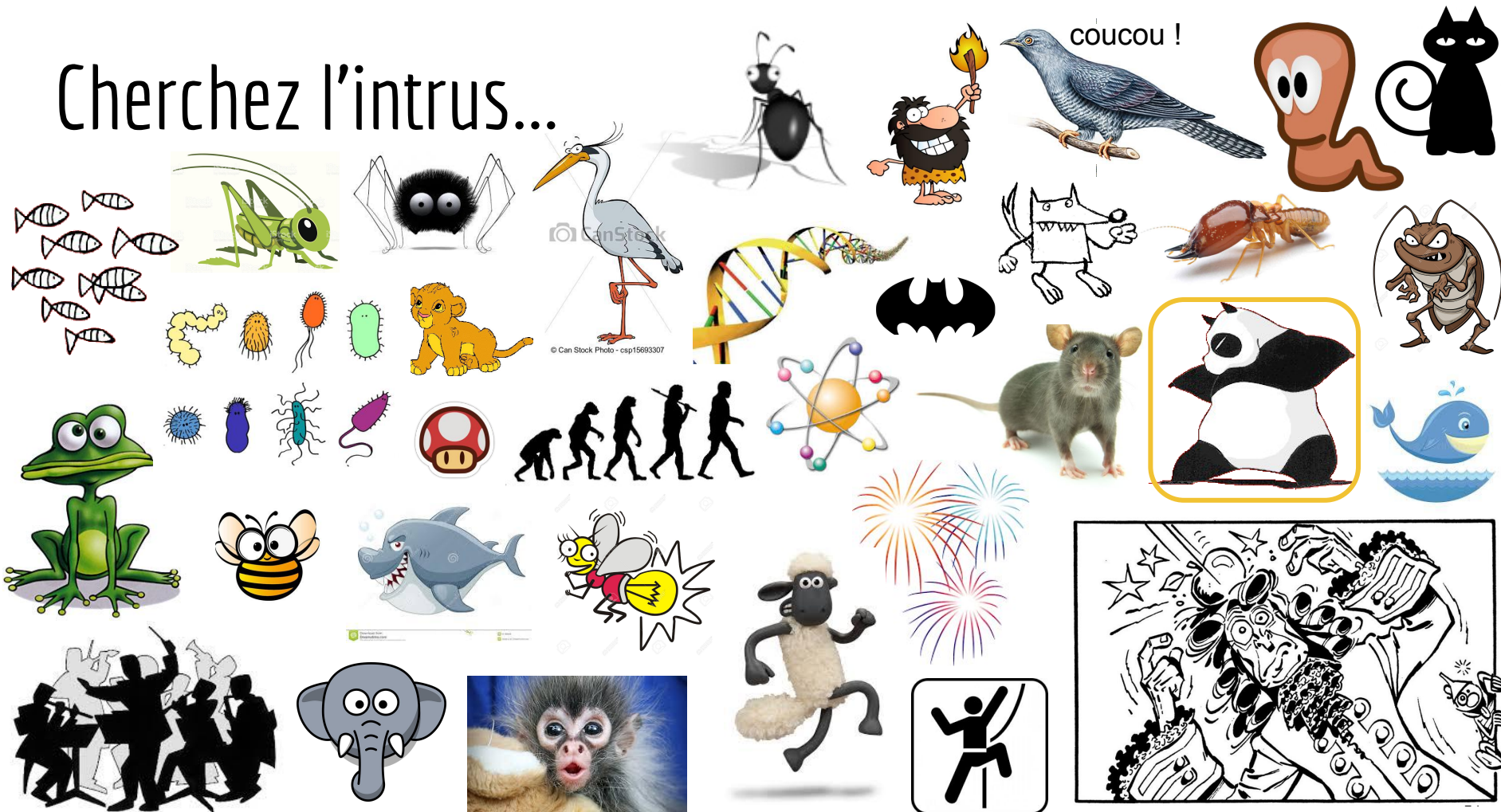
Cherchez l'intrus...

coucou !



Cherchez l'intrus...

coucou !



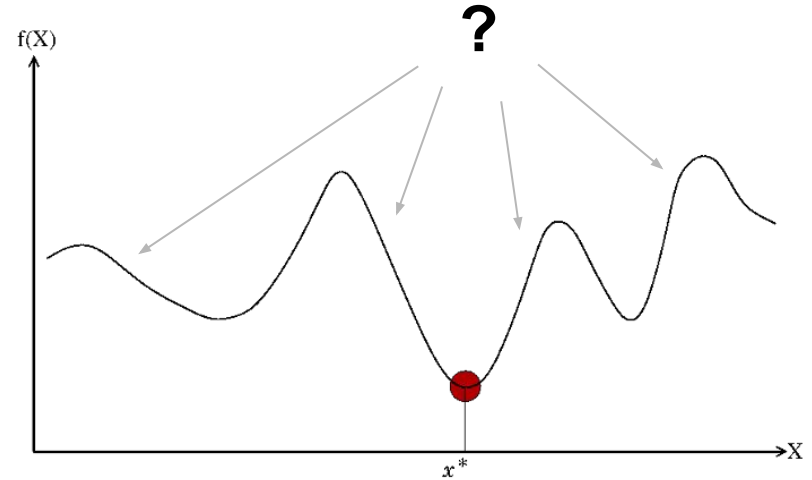
Éléments constitutifs et stratégies

Elles sont généralement composées de :

- Une ou plusieurs (population(s)) solutions initiales ;
- Une stratégie de génération de nouvelle solution basée sur l'aléatoire ;
- Un critère d'acceptation d'une nouvelle solution ;
- Des coefficients de contrôle de comportement de l'algorithme ;
- Le partage d'information, l'utilisation d'une mémoire de solutions explorées ;
- Un critère de convergence.

Initialisation

- Aléatoirement
- Manière intelligente
 - sampling
 - hypercubes latins
 - séquence de Halton
 - ...



Recherche

Exploration

VS

Exploitation



Voisinage

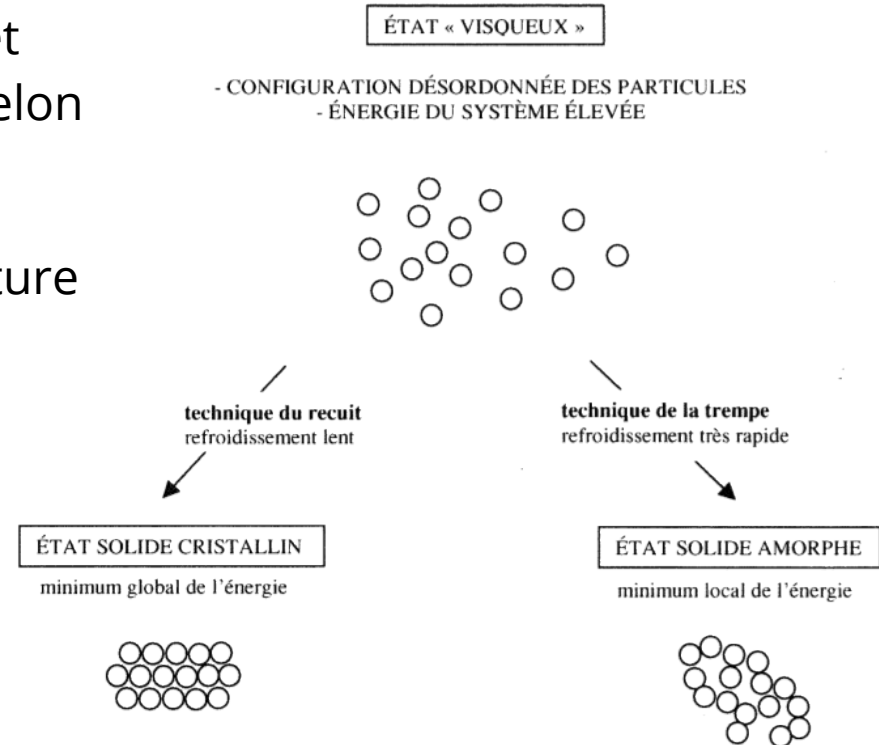
- Définition :
 - Cas des variables discrètes : décalage de composantes de la donnée,
 - Cas des variables continues : hyper(sphère | cube) centré autour de la donnée.
- Stratégies
 - Voisinage géographique,
 - Voisinage social,
 - Voisinage aléatoire.
- Méthodes du gradient, Newton-Raphson, dichotomie, polytope de Nelder-Mead...
- Hybridation avec des méthodes de recherche locale

Le recuit simulé

Pour l'histoire... un peu de physique !

Le recuit est une technique qui permet d'améliorer la qualité d'un matériau selon la méthode suivante :

- On le porte à très haute température pour le liquéfier
- On abaisse progressivement la température pour stabiliser la structure du matériau



Le recuit simulé (Simulated Annealing)

- Métaheuristique variante de l'algorithme de Metropolis-Hastings (voir exemple [ici](#)) ;
- Proposée en 83 par Kirkpatrick, Gelatt et Vecchi et en 85 par Cerny ;
- Première métaheuristique proposée ;
- Adaptée aux problèmes discrets (originellement au placement de composants électroniques sur un circuit imprimé).

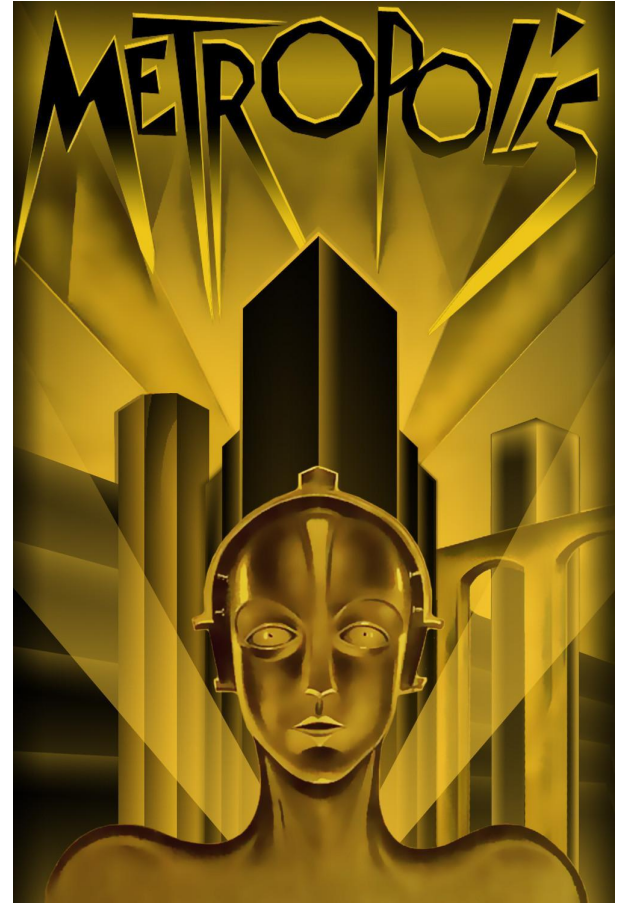
Le recuit simulé - l'analogie

- La fonction f à minimiser est l'énergie du système ;
- Une solution X représente un état du matériau ;
- L'équilibre thermodynamique est atteint lors d'un palier de température ;
- À température T , une perturbation de la solution courante est acceptée avec probabilité basée sur le critère de Metropolis.

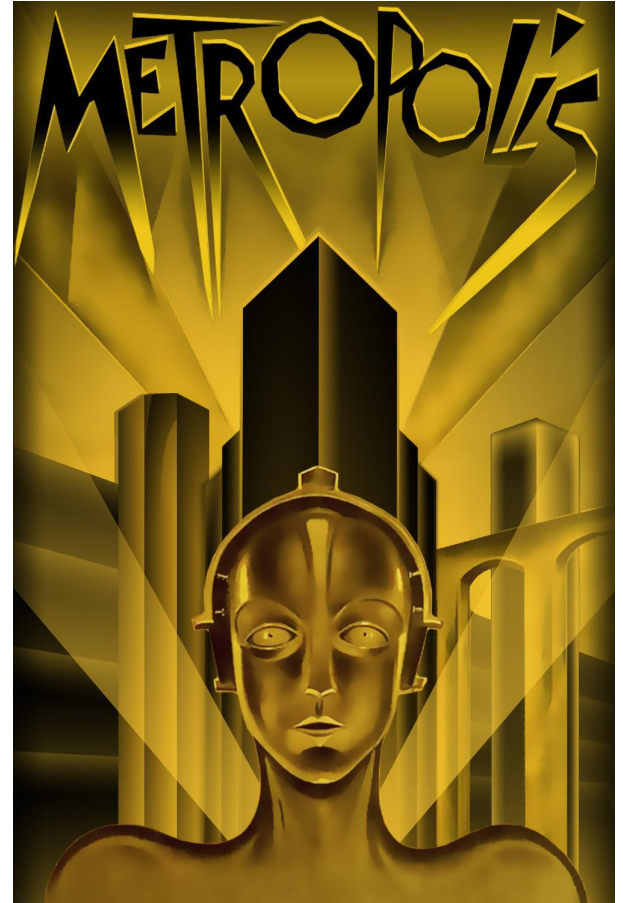
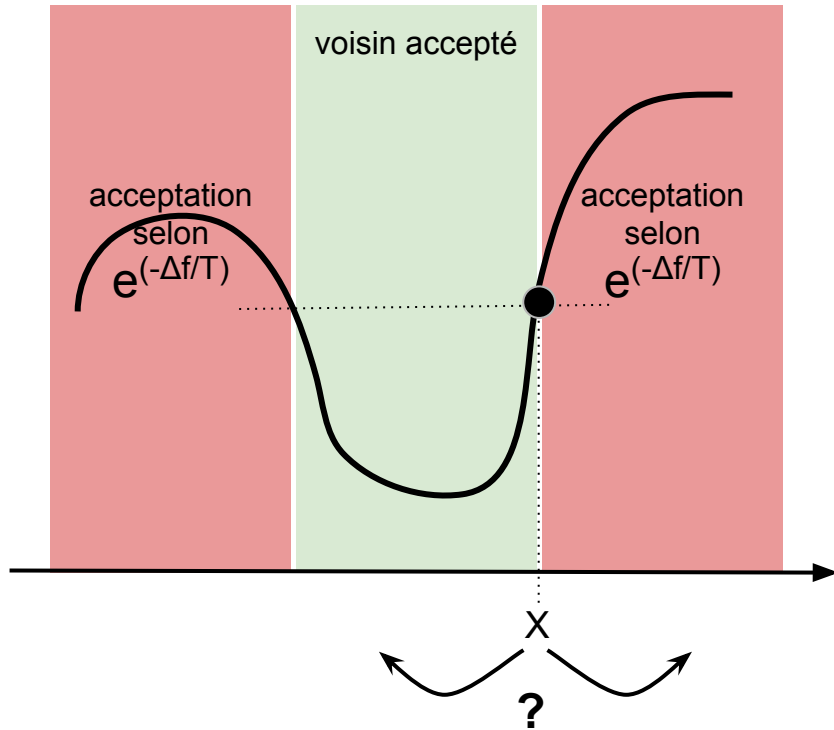
Critère de

```
critereMetropolis( $\Delta f, T$ ) :  
  Si  $\Delta f \leq 0$  alors  
    retourner VRAI  
  sinon  
    retourner  $\text{aléa}(0,1) < e^{(-\Delta f/T)}$ 
```

- $\Delta f \leq 0$, le voisin est accepté ;
- Une petite variation vers un voisin moins bon a plus de chance d'être acceptée qu'une importante ;
- Cette fonction est **stochastique**.



Critère de



Algorithme du recuit simulé

```
X, une solution, f(X) énergie du système, T Température initiale
fmin <- f(X)
Xmin <- X
Tant que T > Tmin ou non critèreConvergence()
  Tant que non équilibreThermodynamique() // palier de température
    Xvois <- perturbation(X)
    Δf = f(Xvois) - f(X)
    Si Δf < 0 ou acceptationCritèreMetropolis(Δf,T) alors
      Si Δf < 0 et f(Xvois) < fmin alors
        fmin <- f(Xvois)
        Xmin <- Xvois
      Fin si
      f(X) <- f(Xvois)
      X <- Xvois
    Fin si
  Fin tant que
  T <- refroidissement(T)
Fin tant que
```

Recuit simulé : Critères de convergence

Permet de terminer l'algorithme selon plusieurs conditions :

- La température atteint une valeur minimale ;
- Le nombre d'évaluations atteint une limite ;
- Il n'y a pas eu d'amélioration depuis un certain nombre d'itérations.

Recuit simulé : Équilibre thermodynamique

Permet de “fouiller” autour d’une bonne solution en cours :

- Le nombre d’évaluations atteint une limite ;
- Il n’y a pas eu d’amélioration depuis un certain nombre d’itérations.

Recuit simulé : Refroidissement

Permet de diminuer la probabilité d'acceptation d'une solution non améliorante selon le critère de Metropolis :

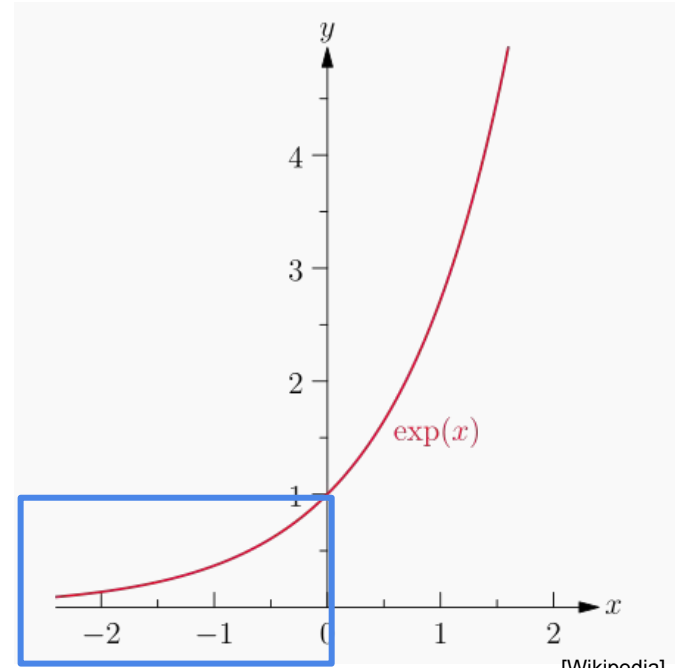
- Une forte décroissance empêche une exploration convenable, l'algorithme converge rapidement vers un minimum local ;
- Une faible décroissance permet une forte exploration lors des premières itérations de l'algorithme et rend peu probable la sélection d'une solution dégradante ;
- À $T=\infty$, toute solution est acceptée ; à $T=0$, aucune solution dégradante n'est sélectionnée.

Recuit simulé : Refroidissement

- À haute température, la valeur du critère de Métropolis est proche de 1, la plupart des solutions dégradantes sont acceptées :



- À faible température, la valeur du critère de Métropolis tend vers 0, la plupart des solutions dégradantes sont rejetées :



Espace de variation du critère de Métropolis

[Wikipedia]

Recuit simulé : Refroidissement

Différents schémas de réduction de la température :

- Géométrique (si l'on ne considère pas les paliers...) : $T_{k+1} = \alpha.T_k$, le plus couramment utilisé ;
- Logarithmique : $T_k = \mu / \log(1+k)$, où k : nb de paliers et μ une constante.
Très coûteux en temps de calcul, peu utilisée ;
- Exponentiel : $T_k = T_0 \cdot \exp(-k/\tau)$, où k : nb de paliers et τ une constante ;
- Ésotérique : on peut remonter la température selon un critère particulier.

Recuit simulé : Perturbation

- Génère une nouvelle solution au **voisinage** de la solution en cours ;
- Influence de la distance entre ces deux solutions (hypervolume du voisinage) ;
- Spécifique au problème à résoudre (discret/continu).

No free lunch theorem [Wolpert and Macready]

Des restaurants (méthodes de résolution) possèdent un menu proposant des plats (problèmes) à différents prix (performance de résolution). Chaque restaurant propose les mêmes plats mais à des prix différents.

Il n'existe pas de menu parfait pour un omnivore.

Un exemple d'application...





That's all folks !