Comment devenir riche rapidement?

Edward Laurence & Guillaume St-Onge

28 avril 2016

Département de physique, de génie physique, et d'optique
Université Laval, Québec, Canada

Il était une fois …

Il était une fois ...

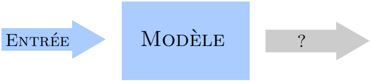
Modélisation

Différents stades de l’expansion d’un porte-feuille

Il était une fois ...

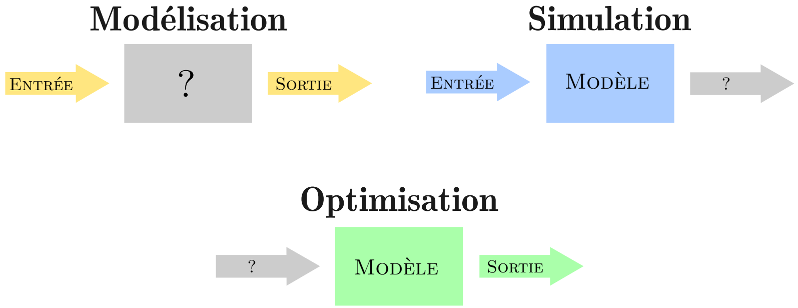
Modélisation

Simulation



Différents stades de l’expansion d’un porte-feuille

Il était une fois ...



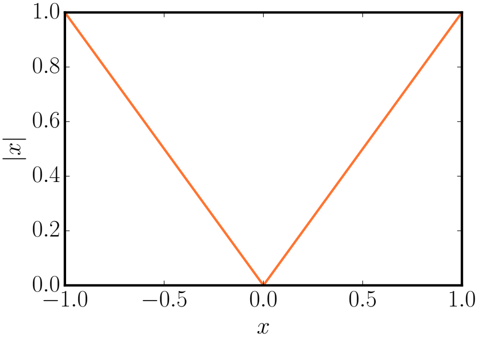
Fonction objective : Fonction de qualité d'une solution

Contrainte : Conditions à respecter

Fonction objective : Fonction de qualité d'une solution

Contrainte : Conditions à respecter

Exemple



Concepts

Concepts

Présentation de trois méthodes

- Algorithme tabou
- Algorithme des lucioles
- Algorithme évolutionniste

Plan de la présentation

Heuristique
Spécialisé à un problème et ne garantit pas la solution obtenue.

Métaheuristique
Algorithme général qu’on doit adapter au problème considéré.

Recherche tabou

Recherche Tabou

Type : Métaheuristique

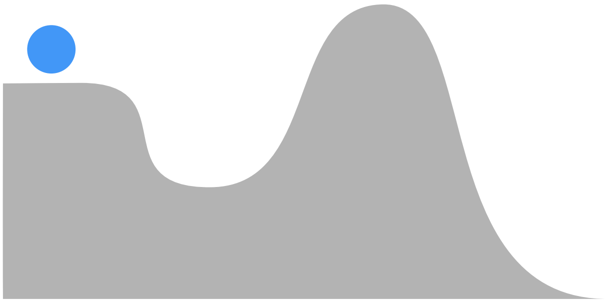
Stochastique : Non

Caractéristique : Recherche locale

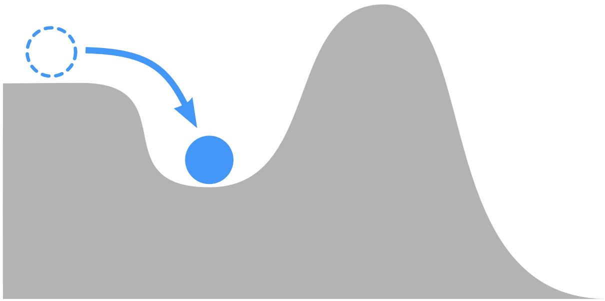
Principes

1. On recherche le mouvement qui minimise notre fonction.
2. On ne revient pas sur nos pas. (d'où *tabou*).
3. Mémoire limitée (*liste tabou*)

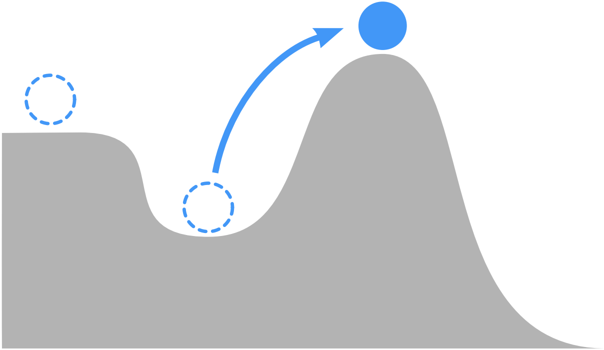
On veut aller au bas de la montagne.



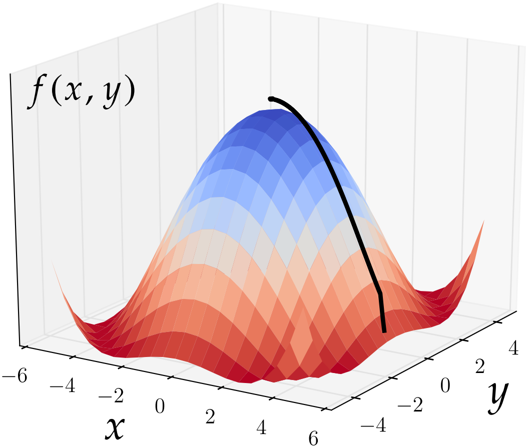
On veut aller au bas de la montagne.



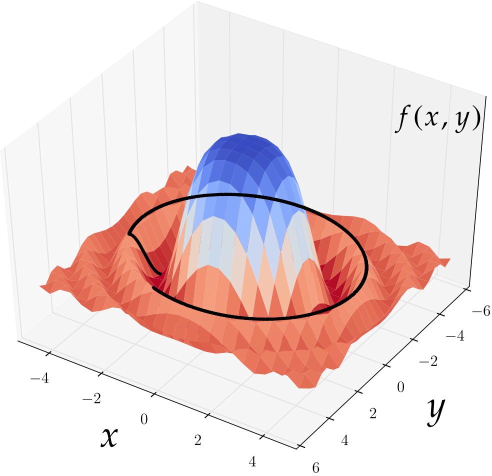
On veut aller au bas de la montagne.



Pour $f(x, y) = \sin(x^2 + y^2)$



$$\text{Pour } f(x,y) = \frac{\sin(x^2+y^2)}{x^2+y^2}$$



Algorithme des lucioles

Recherche par lucioles

Type : Métaheuristique

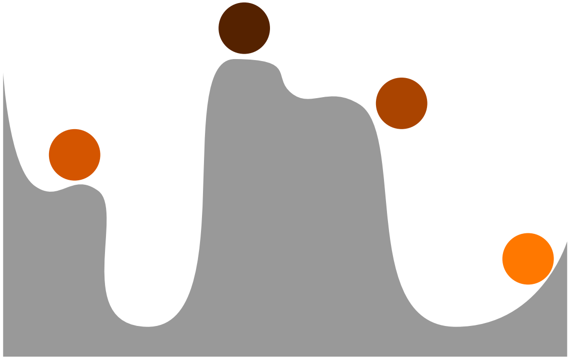
Stochastique : Oui

Caractéristique : Recherche globale

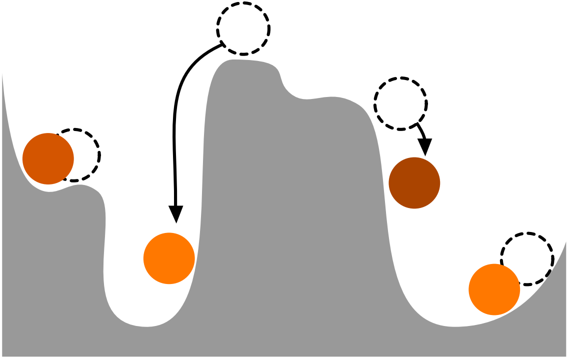
Principes

1. Chaque luciole a une luminosité I et une position.
2. Les lucioles sont attirées par les lucioles plus lumineuses.
3. L'attirance décroît lorsque la distance augmente.

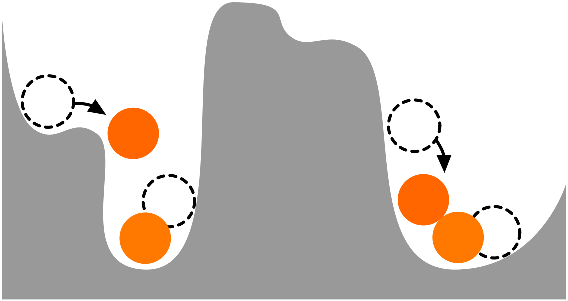
On veut aller au bas de la montagne.



On veut aller au bas de la montagne.



On veut aller au bas de la montagne.



N lucioles à des positions $\boldsymbol{x}_i$

On optimise la fonction $f(\boldsymbol{x})$

$I_i \propto f(\boldsymbol{x_i})$

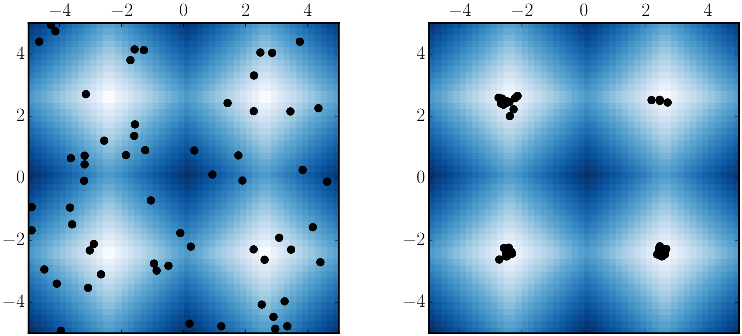
Si $I_j > I_i$

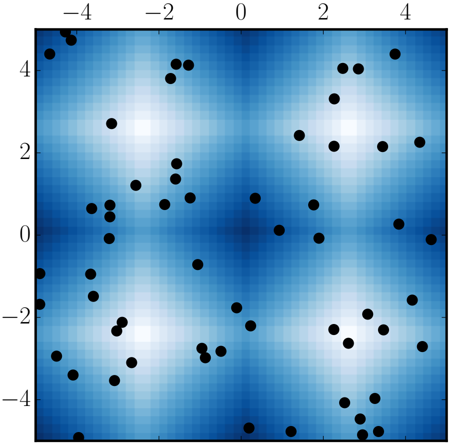
$$\boldsymbol{x}_i \rightarrow \boldsymbol{x}_i + \beta_0 \mathrm{e}^{-\gamma r_{ij}^2} (\boldsymbol{x}_j - \boldsymbol{x}_i) + \alpha \epsilon_i$$

$\beta_0 = 0$: Marche aléatoire

($\gamma = 0$: Optimisation par essaims particulaires)

Trouver un minimum en 2D




−4
−2
0
2
4
4
2
0
−2
−4

Algorithmes évolutionnistes (AE)

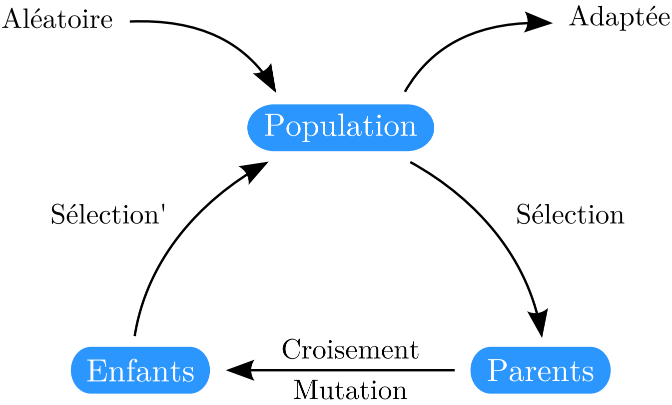
Type : Métaheuristique

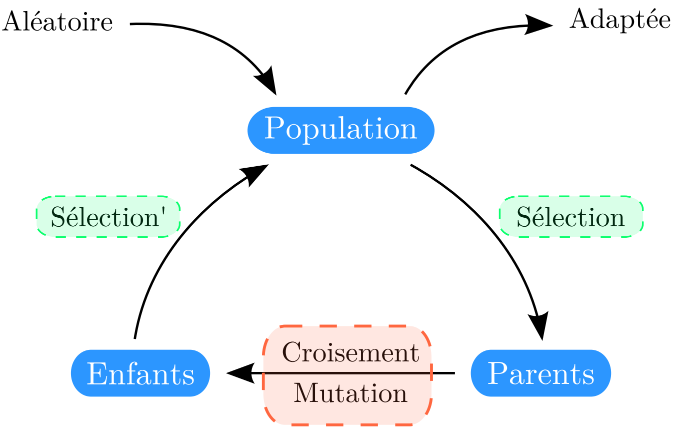
Stochastique : Oui

Caractéristique : Évolution d'une population de solutions

Principes

1. Chaque solution possède un niveau *d'adaptation*
2. Opérateurs de *variation* pour générer de nouvelles solutions
3. Opérateurs de *sélection* pour améliorer l'adaptation des solutions


Aléatoire
Adaptée
Population
Sélection'
Sélection
Croisement
Mutation
Enfants
Parents


Aléatoire
Adaptée
Population
Sélection'
Sélection
Enfants
Croisement
Mutation
Parents

Problème du sac à dos

Knapsack problem
Un revendeur de chocolat doit distribuer sa précieuse cargaison et récolter ses gains. Malheureusement, il n'a le temps de faire qu'une seule tournée avant que son fournisseur n'arrive et son sac à dos peut transporter au plus une masse M.
Quel est le sous-ensemble d'objets lui permettant de garder ses deux jambes ?

20$ - 5kg

5$ - 2kg

45$ - 12kg

Implémentation d’un algorithme génétique

- **Représentation du génome :** Chaîne de bits
- **Niveau d’adaptation :** Prix total des objets sélectionnés
- **Sélection des parents :** Tournoi
- **Croisement des parents :**

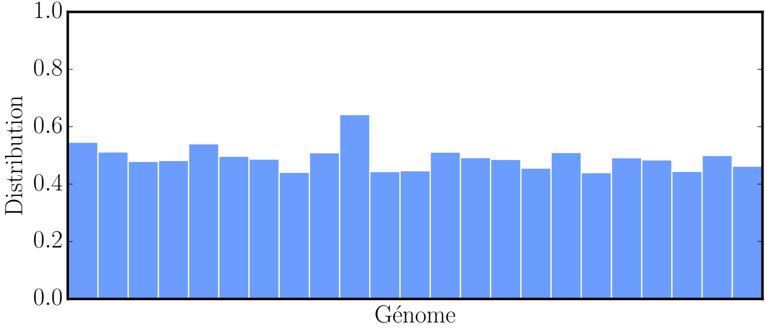
							→							
A	0	1	1	0	1	1		A'	0	1	0	1	0	1
B	1	1	0	1	0	1		B'	1	1	1	0	1	1

- **Mutation :**

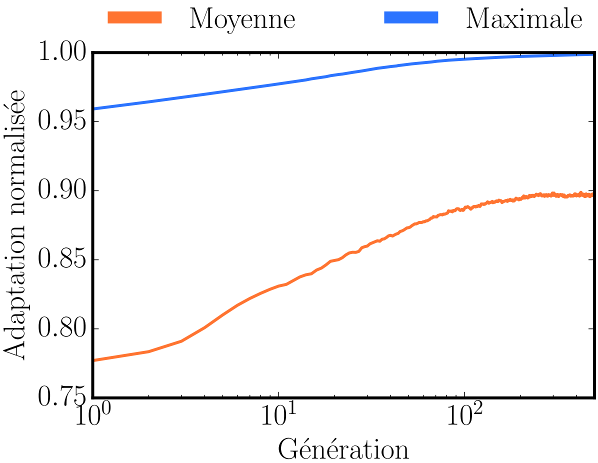
A	0	1	1	0	1	1	→	A'	0	1	0	0	1	1

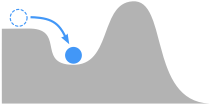
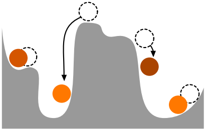
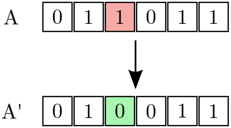
- **Élitisme :** Oui !

Problème du sac à dos - Distribution du génome



Niveau d'adaptation des populations



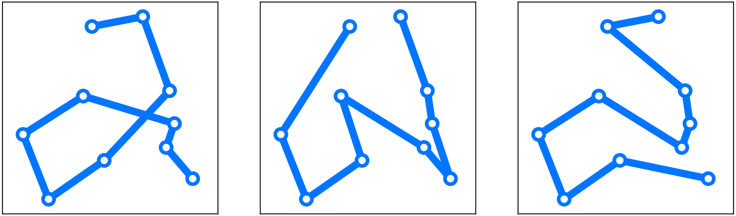
Tabou	**Lucioles**	**Génétique**
Local	Global	Global
Déterministe	Stochastique	Stochastique
-	β_0, γ, α	Modulaire
		

Problème du vendeur

Travelling salesman problem

Un vendeur veut visiter N habitations et marcher le moins possible.

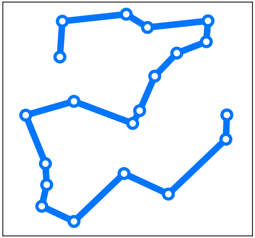
Dans quel ordre doit-il visiter les N maisons ?



Meilleurs parcours pour $N = 20$.

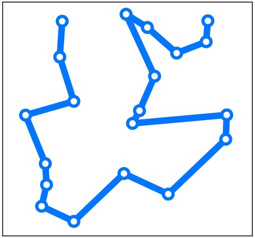
Tabou

35.766



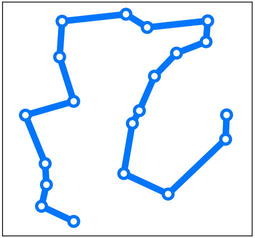
Lucioles

40.171

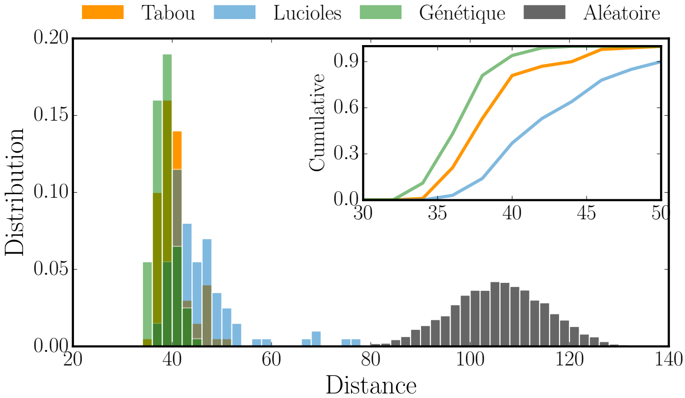


Génétique

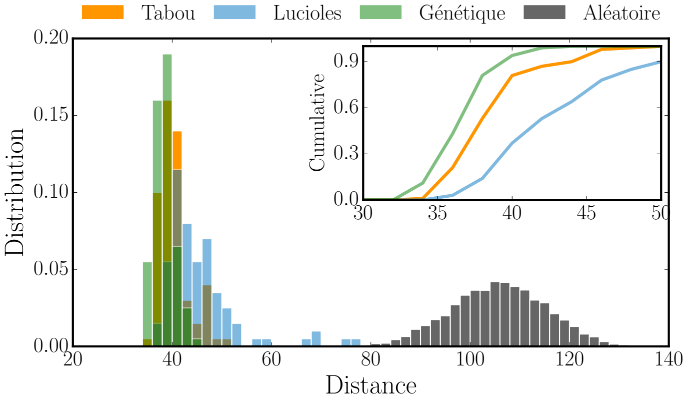
32.45



Distribution de la qualité des solutions

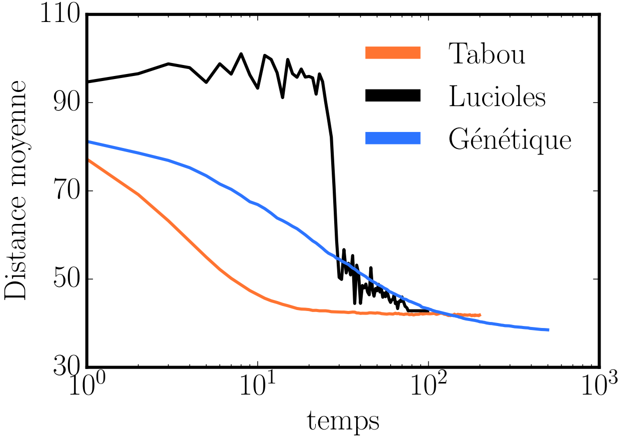


Distribution de la qualité des solutions



Probabilité d'avoir aléatoirement ces solutions : $\sim 10^{-13}$

Distance moyenne en fonction du temps algorithmique



Évaluation sommaire des méthodes

	Tabou	**Lucioles**	**Génétique**
Qualité	A	B-	A+
Vitesse de convergence	A+	B	A
Temps de calcul	A+	B	C
Implémentation	A+	B-	A-
Commentaires	Wow	Boff	Passable

- *Trois méthodes* : Tabou, Lucioles, Génétique.
- Chaque méthode a ses forces et faiblesses.
- Solution à des problèmes complexes.

- *Trois méthodes* : Tabou, Lucioles, Génétique.
- Chaque méthode a ses forces et faiblesses.
- Solution à des problèmes complexes.

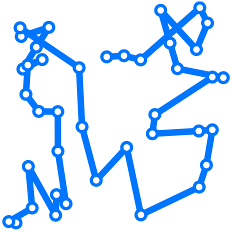
Et si les humains étaient encore meilleurs ?

Humain vs Machine

Pour $N = 30$, les trois meilleurs solutions humaines sont

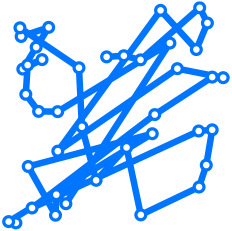
Edward Laurence

21.85



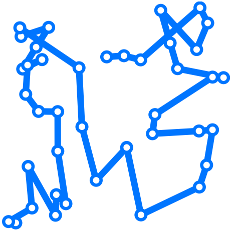
Jacques Rousseau

23.85

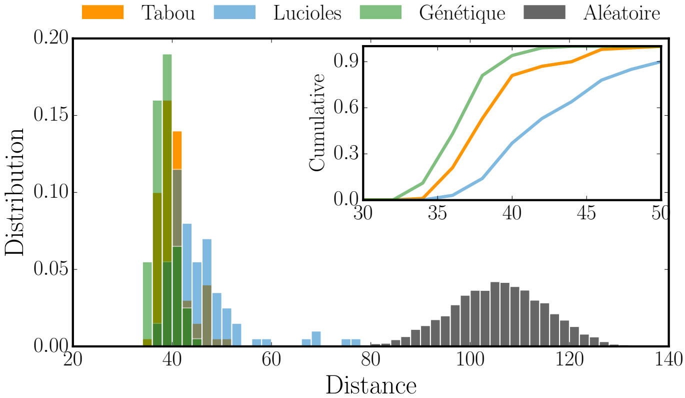


Yves

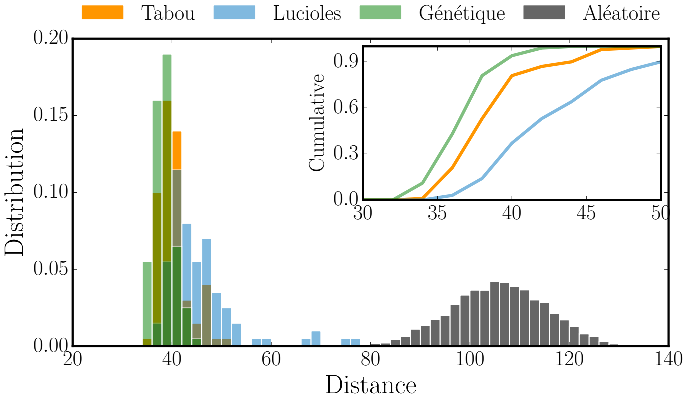
21.85



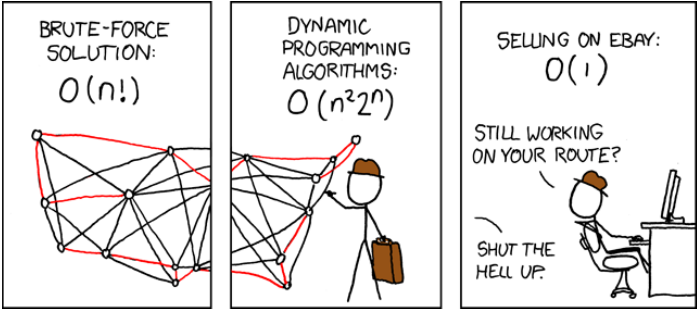
Distribution de la qualité des solutions



Distribution de la qualité des solutions



Probabilité d'avoir aléatoirement ces solutions : $\sim 10^{-13}$


BRUTE-FORCE SOLUTION:
O(n!)
DYNAMIC PROGRAMMING ALGORITHMS:
O(n²2ⁿ)
SELLING ON EBAY:
O(1)
STILL WORKING ON YOUR ROUTE?
SHUT THE HELL UP.