

TIPE : Recuit simulé et chaînes de Markov

Tess BRETON

Notations : On fixe $(\Omega, \mathcal{F}, \mathbb{P})$ un espace probabilisé, et E un ensemble fini d'états de cardinal N typiquement très grand. On choisit une énumération de E , ce qui permet de considérer des vecteurs et matrices indexés par des éléments de E .

$H : E \rightarrow \mathbb{R}$ est une fonction appelée "énergie", h son minimum sur E . On cherche à déterminer un point de $\mathcal{M} = \{x \in E : H(x) = h\}$.

On utilisera la notion de mesure de probabilité, une mesure sur E étant une forme linéaire μ sur $\mathcal{F}(E, \mathbb{C})$ qu'on peut identifier au vecteur ligne $(\mu_x)_{x \in E}$. On identifie alors les mesures de probabilité sur E aux vecteurs $(\mu_x)_{x \in E}$ à coordonnées positives qui somment à 1. On note \mathcal{P} l'ensemble de ces vecteurs.

On utilisera enfin la distance en variation totale entre deux mesures μ et ν :

$$\|\mu - \nu\| = \frac{1}{2} \sum_{y \in E} |\mu(y) - \nu(y)|$$

1 Le principe du recuit simulé

1.1 Idée générale

La méthode d'optimisation par recuit simulé s'inspire de la métallurgie et de la thermodynamique statistique : on simule un refroidissement progressif pour obtenir une configuration optimale minimisant l'énergie.

On explore le graphe de H en générant des successeurs aléatoires (selon un processus à déterminer). Si une transition fait diminuer l'énergie, on l'accepte toujours ; si elle fait augmenter l'énergie, on l'accepte avec une probabilité qui tend vers 0 quand la température diminue. On parvient ainsi, au moins au début, à échapper aux minima locaux.

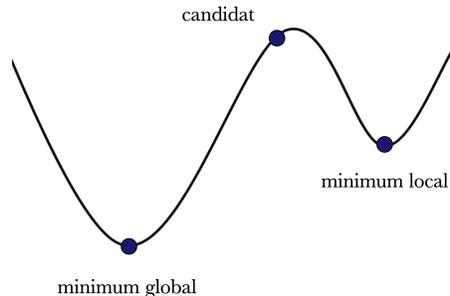


FIGURE 1 – Exemple de fonction présentant un minimum local non global

1.2 Mesures de Gibbs

L'analogie avec la thermodynamique fait naturellement appel aux mesures de probabilité qui gouvernent l'état d'un système à niveaux d'énergie discrets, appelées mesures de Gibbs.

Définition : La mesure de Gibbs associée à la fonction d'énergie H et à la température $T > 0$ est la mesure de probabilité μ_T donnée par :

$$\mu_T(x) = \frac{1}{Z_T} e^{-H(x)/T}, \quad Z_T = \sum_{x \in E} e^{-H(x)/T} \quad (1)$$

Z_T est une constante de normalisation (fonction de partition), généralement difficile à évaluer numériquement. Mais on a souvent plus facilement accès aux rapports $\frac{\mu_T(x)}{\mu_T(y)} = e^{(H(y)-H(x))/T}$, qui ne font intervenir que des différences d'énergie.

Lemme : Quand $T \rightarrow 0^+$, μ_T converge vers la mesure uniforme sur \mathcal{M} .

On va donc chercher à simuler les mesures de Gibbs μ_T sur E . On utilise pour cela l'algorithme de Metropolis et quelques notions sur les chaînes de Markov.

2 Généralités sur les chaînes de Markov

2.1 Premières définitions

Définition : Une chaîne de Markov est une suite $X = (X_n)_{n \geq 0}$ de variables aléatoires vérifiant la propriété de Markov : $\forall (x_0, \dots, x_{n+1}) \in E^{n+2}$,

$$\mathbb{P}(X_{n+1} = x_{n+1} | X_n = x_n, \dots, X_0 = x_0) = \mathbb{P}(X_{n+1} = x_{n+1} | X_n = x_n)^1$$

Par récurrence, si X est une chaîne de Markov, alors pour tout $(x_0, \dots, x_{n+1}) \in E^{n+2}$:

$$\mathbb{P}(X_{n+1} = x_{n+1}, \dots, X_0 = x_0) = \mathbb{P}(X_0 = x_0) \prod_{i=0}^n \mathbb{P}(X_{i+1} = x_{i+1} | X_i = x_i)$$

Définition : On dit que X est homogène si $\mathbb{P}(X_{n+1} = y, X_n = x)$ est indépendante de n pour tout $(x, y) \in E^2$. Dans ce cas, on note $P(x, y) = \mathbb{P}(X_{n+1} = y, X_n = x)$, appelée probabilité de transition de x à y . P ainsi définie est une matrice stochastique, appelée matrice de transition de X .

On a alors : $\mathbb{P}(X_{n+1} = x_{n+1}, \dots, X_0 = x_0) = \mathbb{P}(X_0 = x_0) \prod_{i=0}^n P(x_i, x_{i+1})$

1. sous réserve de bonne définition

La loi ν_n de X_n est donnée par $\nu_n = (\mathbb{P}(X_n = x))_{x \in E} = \nu_0 P^n$ (car $\nu_{n+1} = P\nu_n$).

Définition : Une loi π est invariante pour une chaîne de Markov de matrice P si $\pi = \pi P$.

Définition : On dit que X est réversible par rapport à la probabilité μ si :

$$\forall (x, y) \in E^2, \mu(x)P(x, y) = \mu(y)P(y, x)$$

Notons que si X est réversible par rapport à μ , alors μ est invariante pour X .

2.2 Condition de Doeblin

Définition : On dit que X vérifie la condition de Doeblin s'il existe $l \in \mathbb{N}^*$, $\alpha > 0$, et c une probabilité sur E tels que

$$\forall (x, y) \in E^2, P^l(x, y) \geq \alpha c(y) \tag{2}$$

Notons que s'il existe $k \in \mathbb{N}$ tel que $P^k > 0$, alors P vérifie la condition de Doeblin.

Théorème : Si X vérifie la condition de Doeblin, alors pour toute loi initiale ν_0 , la loi de X_n converge en variation totale vers la loi invariante de X .

Preuve : en annexe

3 L'algorithme de Metropolis

On utilise l'algorithme de Metropolis pour simuler une mesure de Gibbs μ_T sur E . La stratégie est claire : on va construire une chaîne de Markov X vérifiant la condition de Doeblin, et dont la² loi invariante est μ_T . On pourra alors déduire du théorème précédent que pour n assez grand, la loi de X_n est proche de μ_T .

Pour générer des successeurs potentiels, on utilise une matrice stochastique P appelée "matrice de sélection" : $P(x, y)$ est égal à la probabilité de choisir y en partant de x . P est une matrice stochastique que l'on supposera symétrique.

On pose alors :

$$\rho(x, y) = \min \left(1, \frac{\mu_T(y)}{\mu_T(x)} \right) = \begin{cases} e^{-(H(y)-H(x))/T} & \text{si } H(y) > H(x) \\ 1 & \text{sinon} \end{cases}$$

Finalement, l'algorithme de Metropolis se présente comme suit :

— on choisit un voisin y de x avec la probabilité $P(x, y)$

2. la condition de Doeblin implique l'unicité de la loi invariante

— on accepte la transition $x \rightarrow y$ avec la probabilité $\rho(x, y)$

Notons qu'on accepte toujours une transition vers un niveau d'énergie plus faible.

3.1 Formulation mathématique

Soient $Y = (Y_{n,x})_{(n,x) \in \mathbb{N}^* \times E}$ et $U = (U_n)_{n \geq 1}$ deux suites indépendantes de variables aléatoires indépendantes telles que $\mathbb{P}(Y_{n,x} = y) = P(x, y)$ et $U_n \sim \mathcal{U}([0, 1])$.

Soit X_0 une variable aléatoire à valeurs dans E , indépendante de Y et U . On définit par récurrence la suite $X = (X_n)_{n \geq 0}$ en posant :

$$X_{n+1} = Y_{n+1, X_n} \cdot \mathbb{1}_{\{U_{n+1} \leq \rho(X_n, Y_{n+1, X_n})\}} + X_n \cdot \mathbb{1}_{\{U_{n+1} > \rho(X_n, Y_{n+1, X_n})\}}$$

X ainsi définie est une chaîne de Markov de matrice de transition Q donnée par :

$$Q(x, y) = \begin{cases} P(x, y)\rho(x, y) & \text{si } x \neq y \\ 1 - \sum_{z \neq x} Q(x, z) & \text{sinon} \end{cases}$$

Elle est réversible par rapport à la probabilité μ_T , donc μ_T est invariante pour X .

Théorème : Si P vérifie la condition de Doeblin, alors Q aussi. En particulier, X converge en variation totale vers μ_T , son unique loi invariante.

Preuve : en annexe

3.2 Conclusion sur le recuit simulé

Avec l'algorithme de Metropolis, on a donc réussi à approcher les mesures de Gibbs sur E . Le recuit simulé repose sur le même principe, à ceci près qu'on fait décroître la température au cours du temps pour converger vers un point de \mathcal{M} . La chaîne de Markov construite par le même procédé n'est plus homogène, ce qui complique considérablement les preuves de convergence. On en trouvera une dans [3] pour un schéma de température en $c/\ln(n)$.

4 Application au problème du voyageur de commerce

4.1 Présentation du problème

On cherche à minimiser la distance parcourue par un voyageur devant traverser n villes, en passant exactement une fois par chaque ville. Une fois une énumération des villes fixée, on peut identifier un chemin à une permutation de $\{1, \dots, n\}$. L'espace d'états E est donc le groupe symétrique \mathcal{S}_n de cardinal $n!$. L'énergie H est la somme des distances entre deux villes successivement parcourues par le voyageur.

On considère ici $n = 60$ villes générées aléatoirement :

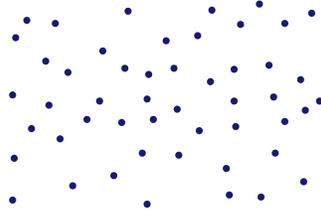


FIGURE 2 – 60 points représentant les villes du TSP considéré

4.2 Choix des paramètres

On choisit un schéma de température en $T_0/\ln(n)$, et la matrice de sélection est définie par le processus suivant : on choisit deux indices i, j aléatoires et on reverse la portion de chemin correspondante :

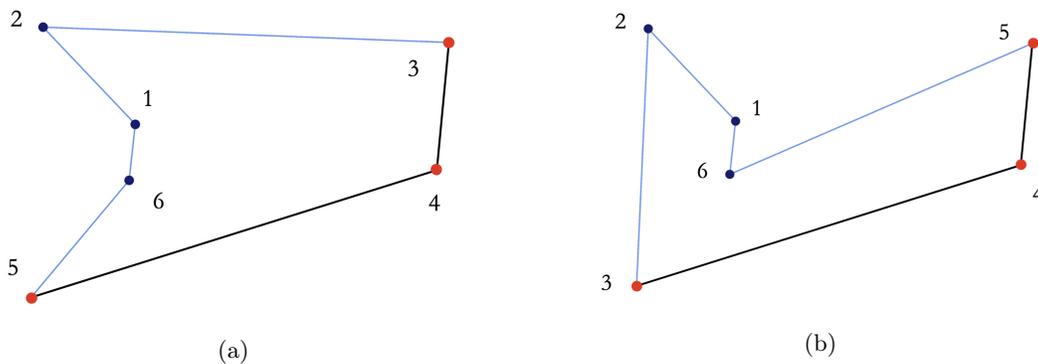


FIGURE 3 – Exemple de renversement entre les indices 3 et 5

Enfin, pour initialiser l'algorithme, on applique un algorithme glouton : on choisit un noeud aléatoire et on construit un chemin en choisissant toujours le plus proche voisin disponible.

4.3 Résultats

4.3.1 Premiers résultats

En appliquant l'algorithme implémenté en python, on remarque que la solution en sortie dépend beaucoup de la solution initiale générée par l'algorithme glouton.

Dans les Figures 4 et 5 : (a) représente la solution initiale générée par l'algorithme glouton, (b) la solution finale renvoyée par l'algorithme de recuit simulé.

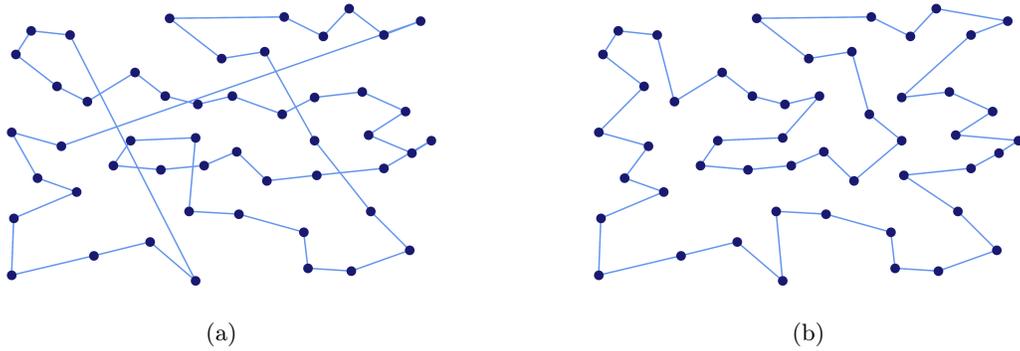


FIGURE 4 – Premier essai ; $H(a)=1724$, $H(b)=1446$

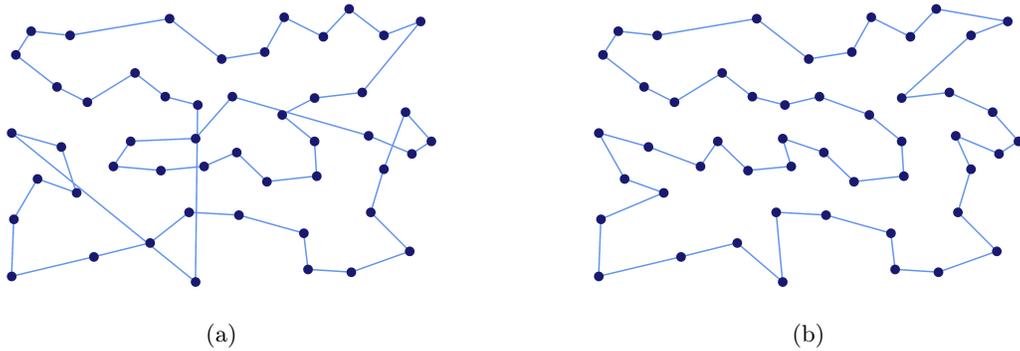


FIGURE 5 – Deuxième essai ; $H(a)=1599$, $H(b)=1356$

4.3.2 "Batch annealing"

Pour remédier au biais de la solution initiale, on peut utiliser la méthode de "batch annealing", qui consiste à appliquer plusieurs fois l'algorithme du recuit. À la fin de la première itération, au lieu de relancer l'algorithme, on garde la sortie en mémoire et on la compare à une nouvelle solution gloutonne ; la meilleure des deux sert alors de solution initiale et on relance l'algorithme. On itère ce procédé, et on obtient de meilleurs résultats. La meilleure solution trouvée pour le problème considéré est présentée en Figure 6.

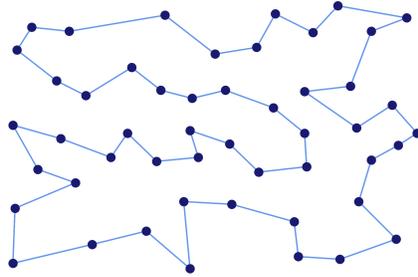


FIGURE 6 – Meilleure solution trouvée : $H=1348$

4.4 Commentaires

Le choix de la matrice de sélection utilisée a été suggéré par [3] pour son efficacité pratique. Mais il convient de souligner l'importance d'un bon choix, car deux voisins doivent rester relativement "proches" : une matrice de sélection correspondant à la loi uniforme donne ainsi de très mauvais résultats, puisqu'une transition détruit dans ce cas toute l'information connue au voisinage d'un état satisfaisant.

On notera enfin que la performance de l'algorithme dépend fortement de la régularité du graphe de H , malgré d'éventuels résultats théoriques de convergence.

Références

- [1] Cihan CEYHAN. *Simulated Annealing algorithm to solve Travelling Salesman Problem in Python*. URL : <https://github.com/chncyhn/simulated-annealing-tsp> (visité le 05/05/2021).
- [2] Djalil CHAFAÏ et Florent MALRIEU. *Recueil de modèles aléatoires*. hal-01897577v4. 2021.
- [3] Jean-François DELMAS et Benjamin JOURDAIN. *Modèles aléatoires : Applications aux sciences de l'ingénieur et du vivant*. 2006.

5 Annexes

5.1 Démonstration : convergence sous condition de Doeblin

Premier cas : $l = 1$:

Si μ et ν sont deux mesures de probabilité sur E , puisque $\sum_{x \in E} \mu(x) = \sum_{x \in E} \nu(x)$ et $\sum_{y \in E} c(y) = 1 = \sum_{y \in E} P(x, y) = 1$, d'après la condition de Doeblin avec $l = 1$:

$$\begin{aligned}
 \|\mu P - \nu P\| &= \frac{1}{2} \sum_{y \in E} |(\mu P)(y) - (\nu P)(y)| \\
 &= \frac{1}{2} \sum_{y \in E} \left| \sum_{x \in E} (\mu(x)P(x, y) - \nu(x)P(x, y)) \right| \\
 &= \frac{1}{2} \sum_{y \in E} \left| \sum_{x \in E} (\mu(x) - \nu(x))(P(x, y) - \alpha c(y)) \right| \\
 &\leq \frac{1}{2} \sum_{y \in E} \sum_{x \in E} |\mu(x) - \nu(x)| |P(x, y) - \alpha c(y)| \\
 &= \frac{1}{2} \sum_{x \in E} |\mu(x) - \nu(x)| \sum_{y \in E} (P(x, y) - \alpha c(y)) \\
 &= (1 - \alpha) \|\mu - \nu\|
 \end{aligned}$$

Ainsi, l'application $\mu \mapsto \mu P$ est une application de \mathcal{P} dans \mathcal{P} , contractante de rapport $1 - \alpha < 1$ pour $\|\cdot\|$, donc (complétude) admet un **unique** point fixe π . Par récurrence : $\pi = \pi P = \pi P^n$ pour tout $n \in \mathbb{N}$. Puis :

$$\|\nu_0 P^n - \pi\| = \|\nu_0 P^n - \pi P^n\| \leq (1 - \alpha)^n \|\nu_0 - \pi\| \leq (1 - \alpha)^n$$

Donc puisque $\alpha > 0$, la loi de X_n converge en variation totale vers π quand n tend vers $+\infty$.

Second cas : $l > 1$:

On se ramène au premier cas en remarquant que $(X_{kl})_{k \geq 0}$ est une chaîne de Markov de matrice de transition P^l : pour toute loi ν_0 de X_0 , on a donc $\|\nu_0 P^{kl} - \pi\| \xrightarrow{n \rightarrow +\infty} 0$, où π est l'unique probabilité vérifiant $\pi = \pi P^l$.

Soit $n \in \mathbb{N}$. Notons $n = kl + r$, où $0 \leq r < l$. Toujours d'après le cas précédent :

$$\|\nu_0 P^n - \pi\| = \|\nu_0 P^{kl+r} - \pi P^{kl}\| \leq \|\nu_0 P^r - \pi\| (1 - \alpha)^k \leq (1 - \alpha)^k \xrightarrow{n \rightarrow +\infty} 0$$

Donc la loi de X_n converge en variation totale vers π quand n tend vers $+\infty$.

Or $\pi = \pi P^l$, donc $\pi P = \pi P^l P = \pi P^{l+1} = (\pi P) P^l$, donc πP est une loi invariante pour P^l .

Par unicité, $\pi = \pi P$: π est une loi invariante pour P .

Enfin, si μ est invariante pour P , alors elle l'est aussi pour P^l , donc $\mu = \pi$.
 π est donc l'unique loi invariante pour P .

5.2 Démonstration : convergence dans l'algorithme de Metropolis

On pose $a = \min \left\{ \frac{\mu_T(y)}{\mu_T(x)} : (x, y) \in E^2 \right\} \in]0, 1]$.

Si $x \neq y$: $Q(x, y) = P(x, y)\rho(x, y) = P(x, y) \min \left(1, \frac{\mu_T(y)}{\mu_T(x)} \right) \geq P(x, y) \min(1, a) = aP(x, y)$

D'autre part : $Q(x, y) \geq P(x, y) \times 1 = P(x, y)$

Donc $Q(x, x) = 1 - \sum_{z \neq x} Q(x, z) \geq 1 - \sum_{z \neq x} P(x, z) = P(x, x) \geq aP(x, x)$

Ainsi, pour tout $(x, y) \in E^2$, $Q(x, y) \geq aP(x, y)$ et :

$$Q^2(x, y) = \sum_{z \in E} Q(x, z)Q(z, y) \geq a^2 \sum_{z \in E} P(x, z)P(z, y) = a^2 P(x, y)$$

Puis par récurrence : pour tout $n \geq 1$, $Q^n(x, y) \geq a^n P(x, y)$.

Donc si $P^l(x, y) \geq \alpha c(y)$, alors $Q^l(x, y) \geq a^l \alpha c(y)$. Donc Q vérifie aussi la condition de Doeblin. D'après le théorème précédent, X converge en variation totale vers sa loi invariante, qui est μ .