

Rapport Tetravex

Ségolène Denjoy
Etienne Sharpin
Laura Lacambra
Promo 2022 - SCIA



6 février 2022

Table des matières

1	Introduction	2
2	Algorithme de recuit	2
2.1	Données d'entrée	2
2.2	Algorithme	2
2.3	Modélisation probabiliste	3
2.3.1	Fonction de température	4
2.3.2	Visualiser	4
3	Résultats obtenus	6
3.1	Mesures de performance	6
3.2	Résultats	6
4	Conclusion	6
5	Annexe	8

1 Introduction

On cherche à résoudre un problème combinatoire, à l'ordinateur. La méthode de résolution est un algorithme de recuit de type Metropolis-Hastings. On joue au jeu Tetravex, illustré ci-dessous.



FIGURE 1 – Jeu du Tetravex

Ce sont les chaînes de Markov et leur propriétés qui ont permis de construire l'algorithme du recuit simulé qui s'appuie sur l'algorithme de Metropolis-Hastings.

Les chaînes de Markov ont la propriété suivante, la distribution conditionnelle de probabilité des états futurs, étant donné les états passés et l'état présent, ne dépend en fait que de l'état présent et non pas des états passés.

2 Algorithme de recuit

2.1 Données d'entrée

En entrée, on dispose d'un fichier texte qui donne les tuiles d'un tetravex ; ce peut-être une configuration de 2x2 tuiles à 6x6 tuiles.

2.2 Algorithme

Dans la méthode du recuit simulé on alterne entre des cycles de refroidissement lent et de réchauffage (recuit), qui ont pour effet de minimiser l'énergie du matériau. Cette méthode est transposée en optimisation pour trouver les extrema d'une fonction. Le recuit simulé peut trouver un état correspondant à la meilleure solution, si on le laisse chercher indéfiniment. Dans notre cas, on cherche à minimiser le nombre de tuiles mal placées, qui représente l'énergie de notre système.

Partant d'un état donné du système, en le modifiant, on obtient un autre état.

Si le nombre de tuiles mal placées diminue, alors on a fait baisser l'énergie du système, et on tend à chercher l'optimum dans le voisinage de l'état de départ.

Sinon, on accepte un « mauvais » état, qui permet alors d’explorer une plus grande partie de l’espace des états, et tend à éviter de s’enfermer trop vite dans la recherche d’un optimum local.

L’état initial :

L’état initial s_0 est pris au hasard dans l’espace des états possibles.

A cet état correspond une énergie initiale E_0 .

Une température élevée T_0 est également choisie de façon arbitraire.

À chaque itération une modification élémentaire de l’état est effectuée.

Si la variation d’énergie est négative, l’énergie diminue, et la modification de l’état est appliquée.

Sinon, la modification de l’état est acceptée avec une probabilité $\exp(-\Delta E/T)$.

On fait baisser la température de façon continue $T_{i+1} = \lambda T_i$, avec λ proche de 1, mais inférieur à 1.

Condition d’arrêt : quand toutes les tuiles sont bien placées. C’est à dire, une énergie de 0.

A la fin, on return l’état optimal dans le fichier txt.

Notre recherche de minima doit aller vite (peu d’itérations) sans s’arrêter dans un minimum local.

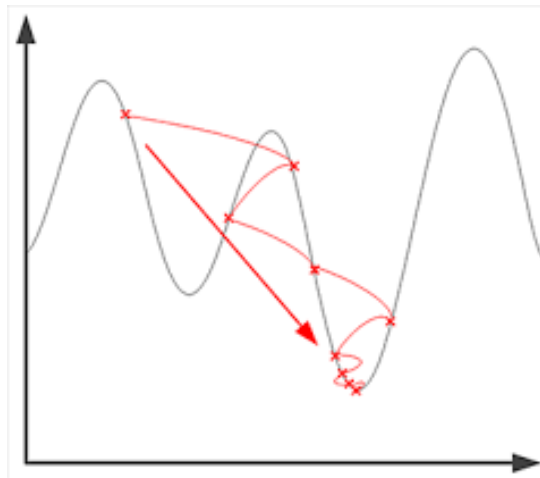


FIGURE 2 – Recuit simulé, Source : ici

2.3 Modélisation probabiliste

Pour optimiser notre algorithme, l’élément le plus impactant à faire varier est la fonction de température. La notre évolue selon une suite géométrique.

2.3.1 Fonction de température

Soient T_0 la température initiale, T_x la température fixée sous certaines conditions, $\{m, n, i\} \in \mathbb{N}^3$, avec i le nombre d'itérations

$$T_n = \begin{cases} \lambda T_0, & \text{if } n = 0 \\ \lambda T_x, & \text{if } E_{n-1} = E_{n-2} = \dots = E_{m+1} = E_m \text{ and } m - n - 1 \geq i \\ \lambda T_{n-1}, & \text{sinon} \end{cases}$$

Nous pouvons faire varier :

- Le lambda λ
- La température initiale T_0
- Le nombre d'itérations i au bout duquel on remonte/reset la température
- La température remontée T_x

Nous avons aussi une dernière option à tester :

$$T_n = \begin{cases} \lambda T_0, & \text{if } n = 0 \\ \lambda(T_{n-1} + T_x), & \text{if } E_{n-1} = \dots = E_m \text{ and } m - n - 1 \geq i \\ \lambda T_{n-1}, & \text{sinon} \end{cases}$$

Dans ce cas là, nous devrions sortir plus lentement d'un minimum local selon le T_x choisi.

2.3.2 Visualiser

Si l'on trace la courbe de température en fonction du nombre d'itérations, nous devrions avoir en théorie une courbe qui ressemblerait à cela :

Le Recuit Simulé.

- Inspiré du procédé de recuit des métaux :
 - Pour arriver à un état stable (optimal) après fusion, un métal est refroidit lentement.
 - Au cours du refroidissement, il est recuit par intermittence.
 - Ceci lui permet d'arriver à un état d'énergie minimal stable, optimal.

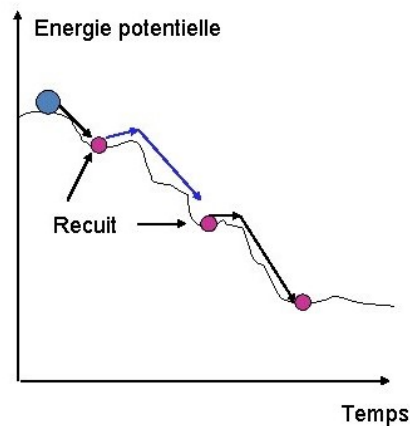


FIGURE 3 – Source : ici

Nous obtenons ces graphes avec nos variables fixées :

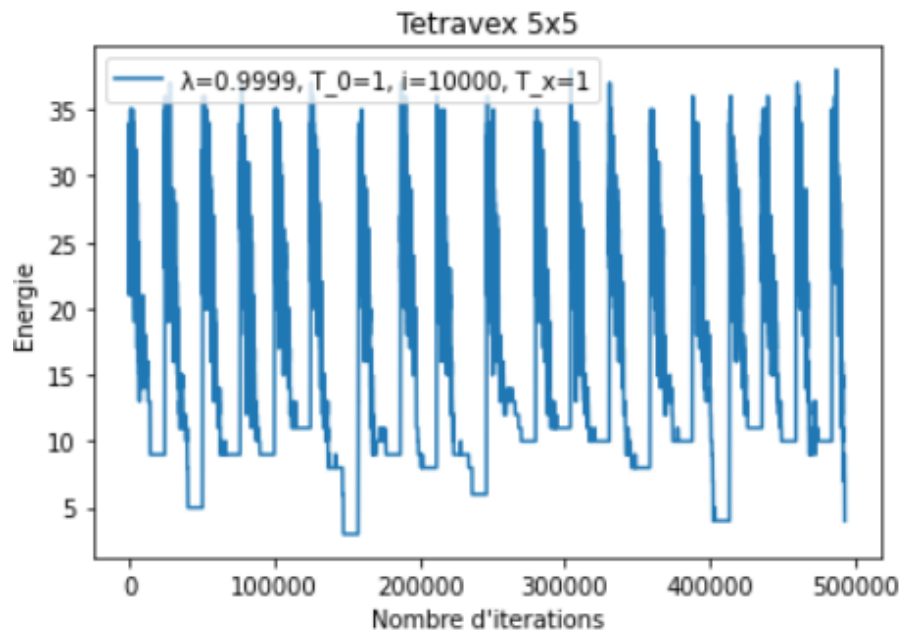


FIGURE 4 – Graphe d'énergie tetravex 5x5

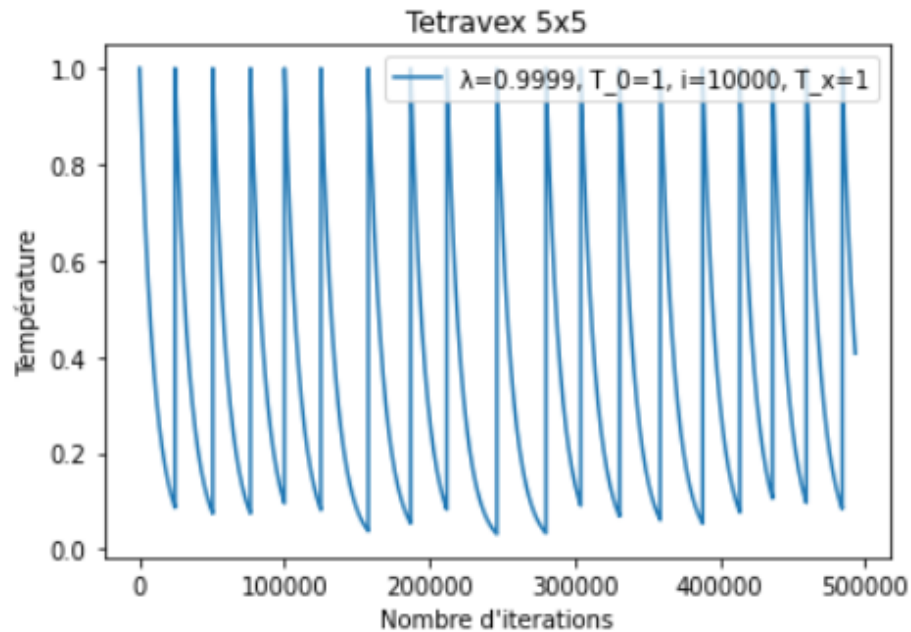


FIGURE 5 – Graphe de Température tetravex 5x5

3 Résultats obtenus

3.1 Mesures de performance

Nos mesures sont principalement basées sur :

- le temps d'exécution : Nous testons notre code grâce à la fonction shell **time**
- le nombre d'iterations total

3.2 Résultats

Testé sur des Tetravex 5x5 sans tuile fixée.

Paramètres					Résultats obtenus			
Test	λ	T_0	i	T_x	Temps d'exécution	MTE	Nbr d'iter total	Graphes
Test 1	0.9999	1	10000	1	[0,698s ; 1min]	15.8s	1.4e6	Figure 7
Test 2	0.9999	500	10000	500	[2,365s ; 3min]		1.4e6	Figure 9
Test 3	0.9999	1	1000	1	[3,6s ; 31s]	15.6s	1.2e6	Figure 11
Test 4	0.9999	1	100000	1	[24,544s ; 1m0,089]		2e6	Figure 13
Test 5	0.995	1	10000	1	[1,682s ; 41,974s]	11s	0.8e6	Figure 15

MTE : Moyenne temps d'exécution

Test 2 : En augmentant le T_0 , l'algorithme est beaucoup plus soumis à l'aléatoire. Les tuiles vont être bien plus mélangées. Le temps d'exécution oscille donc entre 2s et 3m2,906s selon nos quelques tests.

Test 3 : On observe moins de palliers sur le graphe d'énergie (vers 10 d'énergie) 11.

Test 4 : Ce test confirme notre intuition : en augmentant T_0 nous voyons bien plus de palliers d'énergie. La fonction de température n'est donc pas optimale.

Nous pourrions faire une descente de température plus logarithmique. Aussi, nous nous demandons si les optimisations que l'on peut apporter ne peuvent pas varier en fonction de la taille du tetravex.

4 Conclusion

La puissance du recuit simulé réside dans le fait de pouvoir trouver systématiquement un état correspondant à la meilleure solution, si on le laisse chercher indéfiniment.

La température joue un rôle important. À haute température, le système est libre de se déplacer dans l'espace des états en choisissant des états ne minimisant pas forcément l'énergie du système. À basse température, les modifications baissant l'énergie du système sont choisies, mais d'autres peuvent être acceptées, empêchant ainsi l'algorithme de tomber dans un minimum local.

Les principaux inconvénients du recuit simulé résident dans le choix des nombreux paramètres, tels que la température initiale, la loi de décroissance de la température, les critères d'arrêt ou la longueur des paliers de température. Ces paramètres sont souvent choisis de manière empirique.

Au delà de l'algorithme, nous aurions pu optimiser le code C pour être encore plus rapide comme par exemple en unrollant les boucles à la main, minimisant les appels de fonctions ou en utilisant plus le préprocesseur. En effet, si l'on compile avec le flag `Ofast`, on a un temps d'exécution divisé par 3. Cependant, nous avons délibérément fait le choix de garder un code lisible pour l'humain.

“Programs are meant to be read by humans and only incidentally for computers to execute.”

— Donald Knuth

5 Annexe

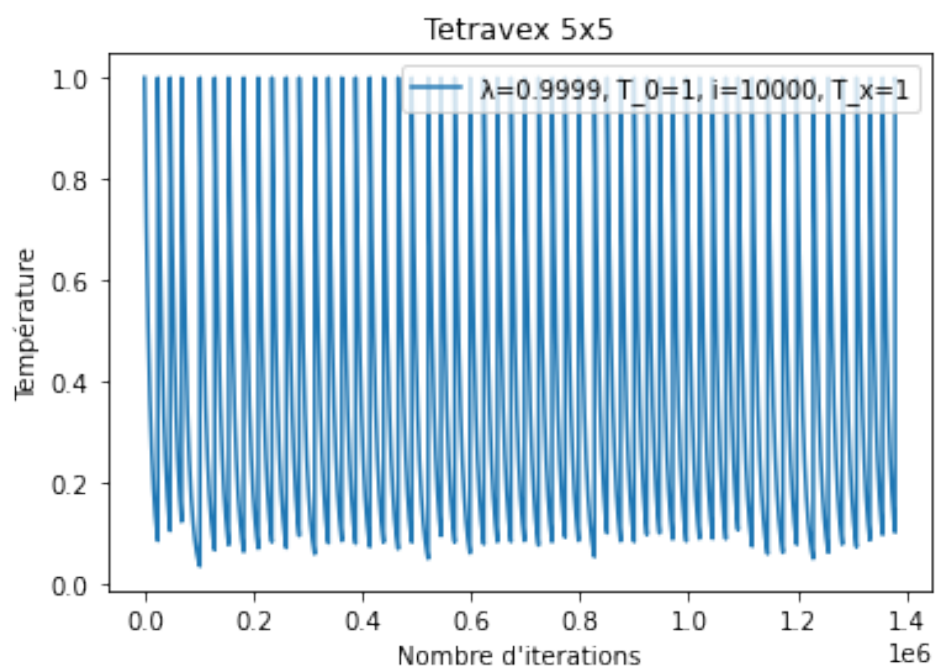


FIGURE 6 – Graphe d'énergie tetravex 5x5

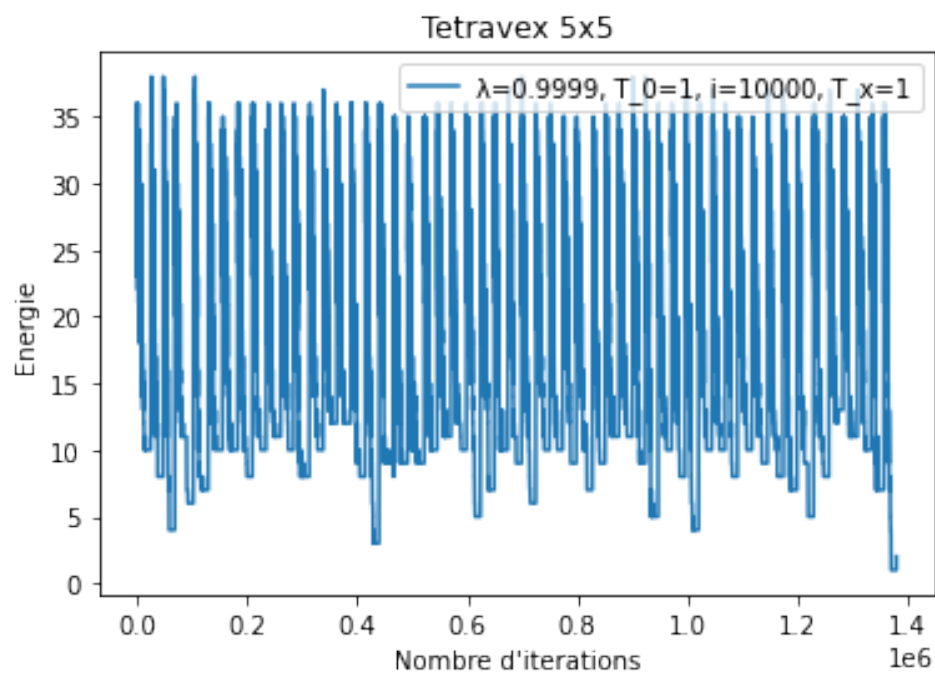


FIGURE 7 – Graphe de Température tetravex 5x5

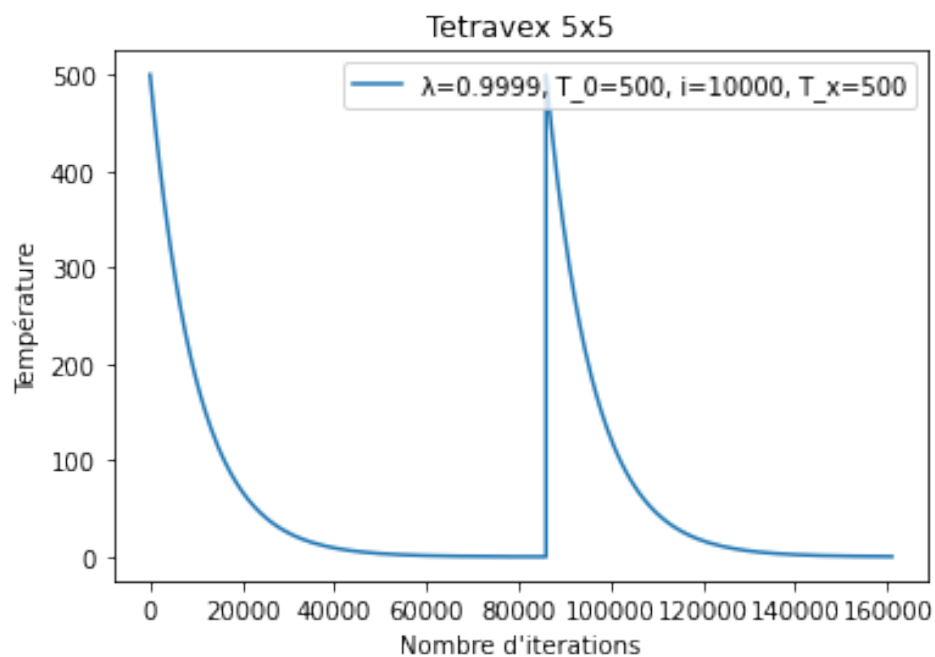


FIGURE 8 – Graphe d'énergie tetravex 5x5, version 2s

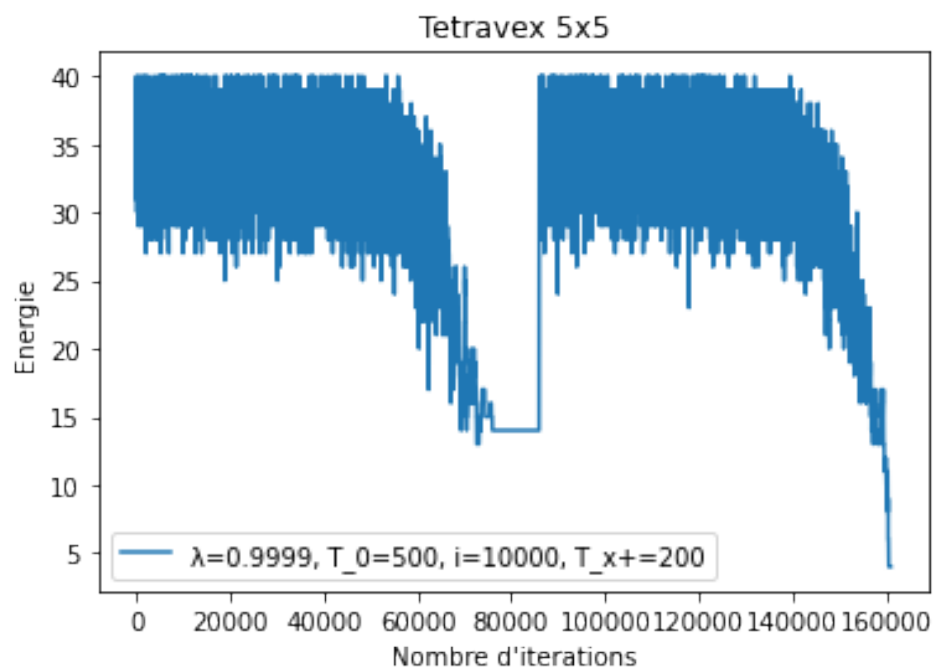


FIGURE 9 – Graphe de Température tetravex 5x5, version 2s

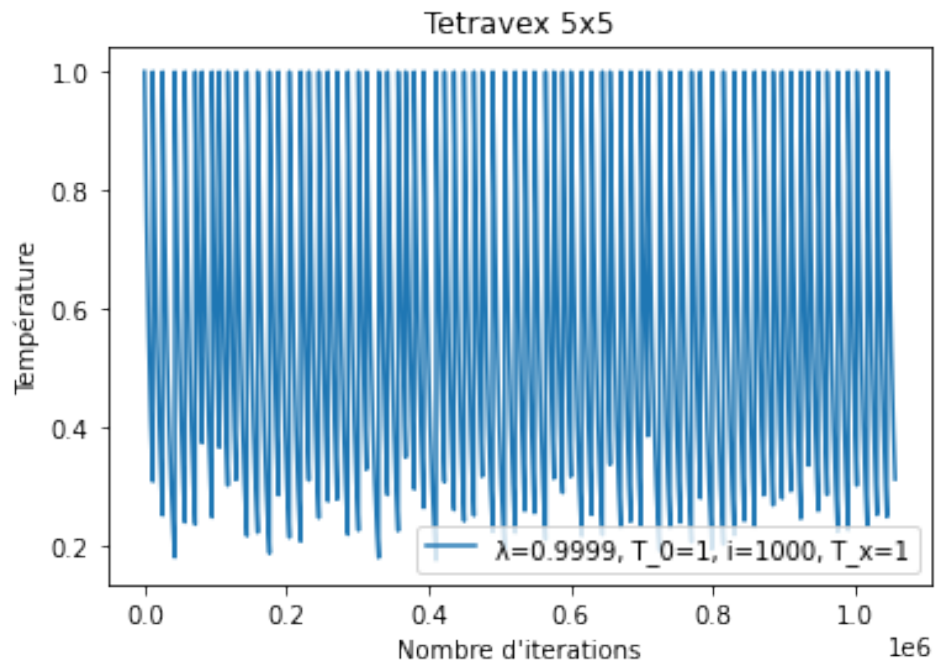


FIGURE 10 – Graphe d'énergie tetravex 5x5

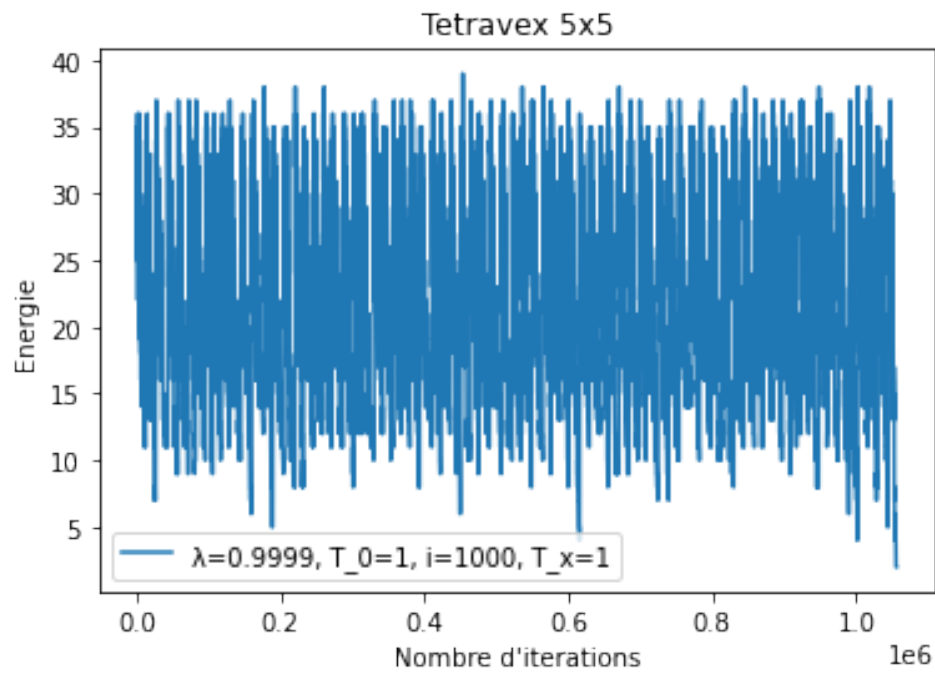


FIGURE 11 – Graphe de Température tetravex 5x5

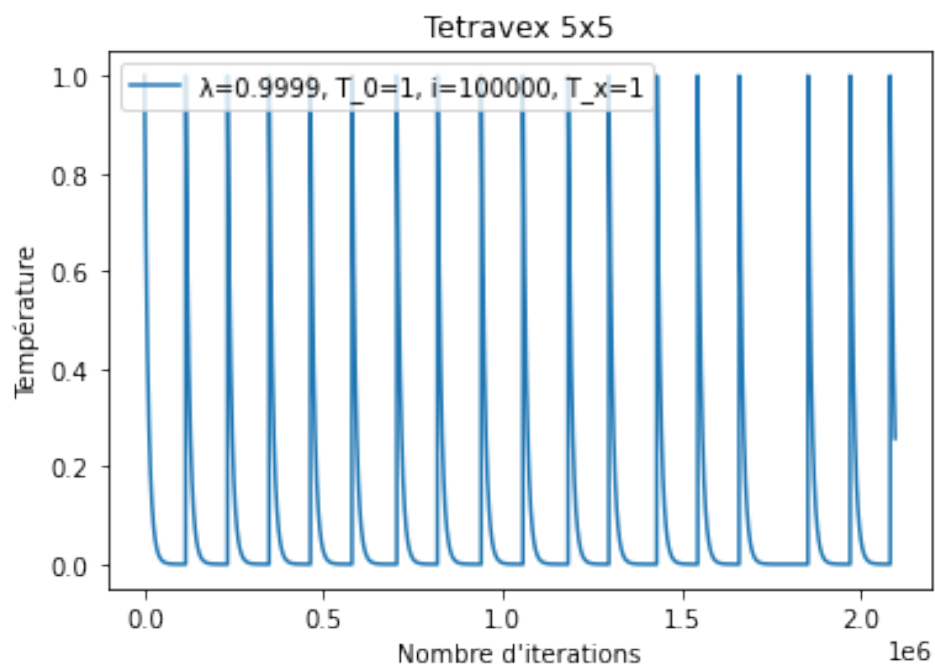


FIGURE 12 – Graphe d'énergie tetravex 5x5, version 24s

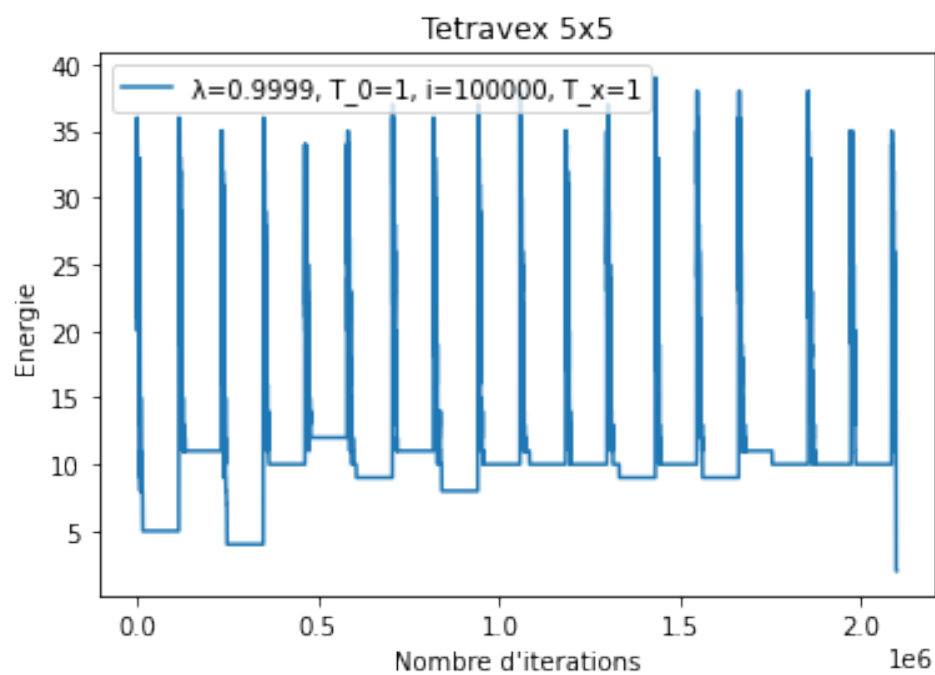


FIGURE 13 – Graphe de Température tetravex 5x5, version 24s

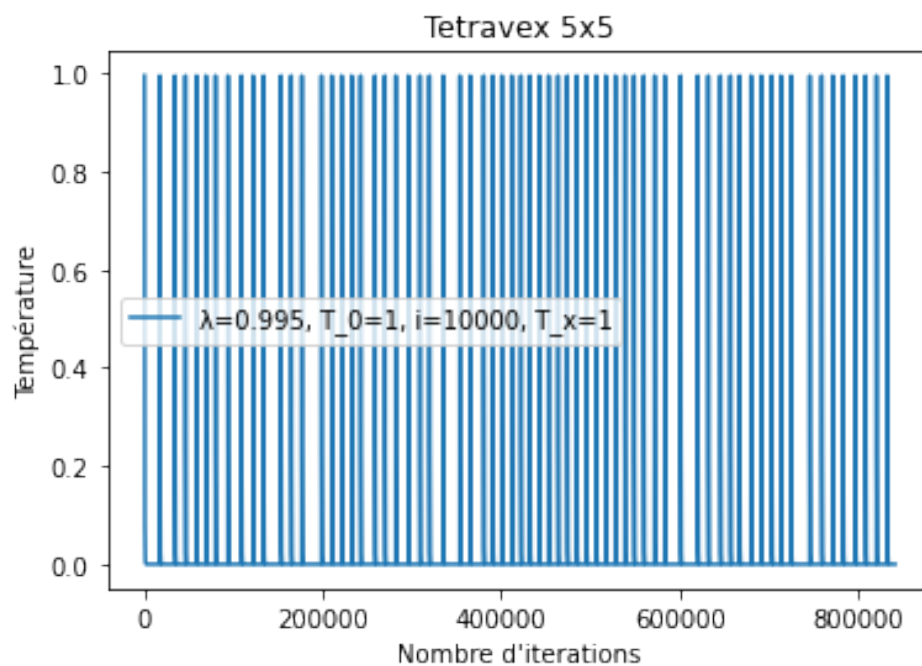


FIGURE 14 – Graphe d'énergie tetravex 5x5, version 7s

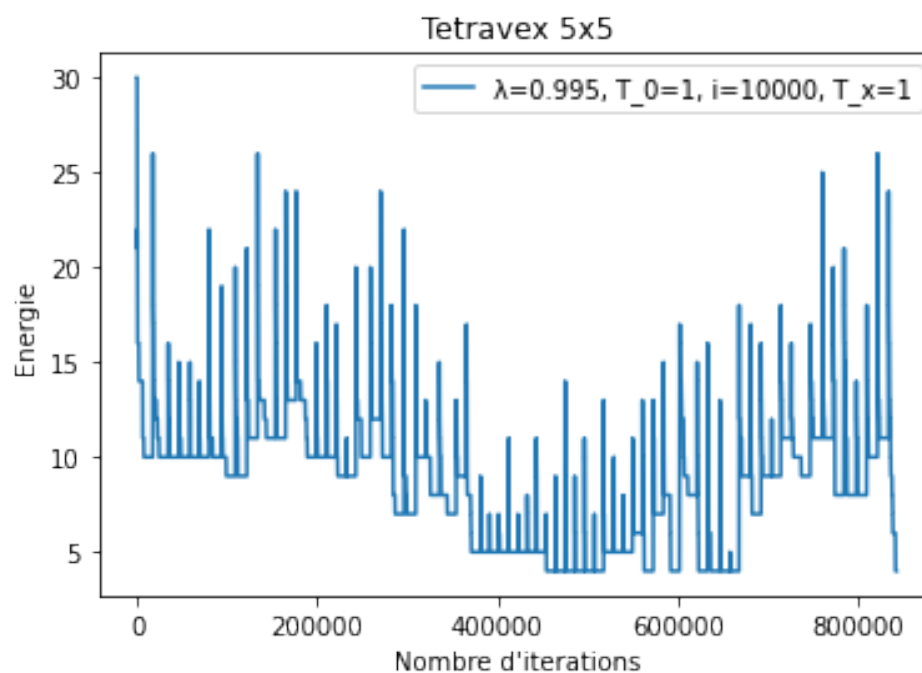


FIGURE 15 – Graphe de Température tetravex 5x5, version 7s