TP : Automates cellulaires – le jeu de la vie [1]

Objectifs du TP

- Reprendre en main python et en particulier les fonctions, les listes et/ou les tableaux, la lecture dans les fichiers.
- Faire de petits calculs de complexité.
- Découvrir les automates cellulaires.

Préliminaires

Jeu de la vie. Ce problème est consacré à l'étude de l'évolution d'une population de *cellules*, dont la naissance et la mort se décide au moyen de règles algorithmiques inventées par John Conway en 1 970.

L'univers considéré est un quadrillage, où chaque case peut être dans deux états possibles : soit vide, soit occupée par une cellule. La configuration de l'univers à un instant t est ainsi l'état d'un automate.

Chaque cas a 8 voisines, 4 en diagonale et 4 orthogonales. Pour passer de l'instant t à l'instant $t+1$, les règles ci-dessous définissent la nouvelle configuration.

- **Règle de mort.** Si, à l'instant t, une case contient une cellule vivante qui a exactement 2 ou 3 voisines vivantes, elle contient une cellule vivante à l'instant $t+1$. Sinon, la cellule meurt et la case devient vide.
- **Règle de naissance.** Si, à l'instant t, une case est vide et a exactement 3 voisines vivantes, elle contient une cellule vivante à l'instant $t+1$. Sinon, cette case reste vide.

La suite d'entiers positifs (u_n) ci-dessous servira à définir la configuration initiale de l'univers.

- $u_0 = 12345$
- $u_n = 16383 \times u_{n-1}\ [59047]$ pour $n > 0$.

1. Que valent u_{996} et u_{9996} ?

2. On définit $v_i = 1$ si $u_i \equiv 0\ [3]$ et $v_i = 0$ sinon. Combien y a-t-il d'indices i tels que $v_i = 0$ pour $0 \leqslant i < 10000$?

1. Librement inspiré d'un TP du concours informatique des ENS

Le jeu torique

Dans cette section, l'univers est un tore de dimensions $k \times k$. Cela signifie que l'univers est un carré de dimensions $k \times k$ où les cases du bord supérieur sont voisines de celles du bord inférieur et où les cases du bord droit sont voisines de celles du bord gauche.

On représentera, au choix, l'univers comme une liste (python) de k listes de k nombres parmi 0 et 1, ou comme un tableau (numpy) de format (k, k) de type `bool`.

3. Écrire une fonction `grille_vide(k)` renvoyant une grille (c'est-à-dire un univers) vide (c'est-à-dire sans cellule vivante) de taille $k \times k$.

Lecture d'univers dans des fichiers

On vous fournit des univers de départ dans les fichiers `grilleX.txt` qui sont des fichiers textes contenant des caractères 0 ou 1 et donc chaque ligne représente une ligne de l'univers.

4. Écrire une fonction `file2grid(nomFichier)` qui prend en entrée une chaîne de caractère correspondant à un nom de tel fichier et qui renvoie la grille correspondante.

On donne dans le fichier `affichage.py` deux fonctions permettant un affichage graphique des univers. Ouvrir le fichier et comprendre ces fonctions. L'enregistrer dans le répertoire de travail.

On utilisera `from affichage import *` pour importer ces fonctions dans le fichier de travail.

5. Tester en affichant une grille contenue dans un des fichiers.

Évolution de l'univers

6. Écrire une fonction `nb_vois_vivantes(grille, i, j)` renvoyant le nombre de cellules voisines de `(i, j)` vivantes dans `grille`.

7. Écrire une fonction `evolue(grille)` qui, à partir d'une grille, renvoie une nouvelle grille correspondant à l'évolution de la grille selon les règles du jeu de la vie. Quelle est sa complexité en fonction de k ?

8. Écrire une fonction `visualiser(grille, fin)` qui prend en entrée une grille et qui la fait évoluer en affichant à chaque étape la nouvelle grille, de $t = 0$ jusqu'à $t =$`fin`.

Tester avec les grilles des fichiers. Que remarque-t-on ?

Univers basé sur (u_n)

Pour les questions suivantes, on prendra successivement $k = 9,\ 20,\ 40$. La configuration de départ, correspondant à l'instant $t = 0$, est définie comme suit : la case de coordonnées (i, j) pour $0 \leqslant i, j < k$ contient une cellule vivant si et seulement si $v_{i+j\times k} = 1$.

9. Écrire une fonction `init_grille(k)` renvoyant la grille d'une telle configuration de départ, ainsi que le nombre de cellules vivantes qu'elle contient.

10. Pour la configuration de départ, $(t = 0)$, la case de coordonnées $(1, 0)$ contient-elle une cellule vivante ?

11. Combien y a-t-il de cellules vivantes aux instants $t = 0$, $t = 1$, $t = 10$, $t = 100$, $t = 1000$ et $t = 10000$? On pourra modifier la fonction `evolue` pour qu'elle renvoie le nombre de cellules vivantes après évolution, éviter d'afficher les grilles ce qui prend du temps, et tirer profit de la remarque de la question **8** en arrêtant les calculs prématurément. Évaluer la complexité de la fonction en fonction de k et de t.

Détermination de la période

L'univers, soumis aux règles du jeu de la vie, est un automate déterministe. Il est possible qu'il existe deux instante $t_0 < t_1$ où l'univers est dans la même configuration.

12. Justifier que dans le cas de l'univers torique, deux tels instants existent toujours.

On notera t_0 et t_1 les plus petites valeurs telles que l'univers est dans la même configuration. On appelle alors **période de l'attracteur** la valeur $t_1 - t_0$ et **temps d'attraction** la valeur t_0.

13. Calculer la période de l'attracteur et le temps d'attraction. Évaluer la complexité de la fonction en fonction de k et de t_1.

Des îlots

On appelle **chemin** dans un quadrillage une suite de cases telle que deux cases consécutives soient voisines (on rappelle que chaque case a 8 voisines). L'ensemble des cellules vivantes de l'univers peut être découpé en **îlots** qui sont les composantes connexes définies par la relation de voisinage : entre deux cellules vivantes d'un même îlot, il y a toujours un chemin ne passant que par des cellules vivantes.

14. Pour la configuration de départ $(t = 0)$, calculer le nombre de cellules de l'îlot contenant la cellule de coordonnées $(0, 3)$.

On pourra écrire une fonction récursive ou utiliser une liste pour empiler les cellules dont on doit inspecter les voisines.

15. Calculer le nombre d'îlots aux temps $t = 0$, $t = 1$, $t = 10$ et $t = t_0$.

Des îles

De la même façon que le voisinage d'une case contient 9 cases (la case et ses 8 voisines), on peut définir le **X-voisinage** qui contient 25 cases : les voisines de ses voisines. On appelle **X-chemin** dans un quadrillage une suite de cases telle que deux cases consécutives soient X-voisines. L'ensemble des cellules vivantes de l'univers peut aussi être découpé en **îles**, qui sont les composantes connexes définies par la relation de X-voisinage : entre deux cellules vivantes d'une même île, il y a toujours un X-chemin ne passant que par des cellules vivantes.

16. Pour la configuration de départ $(t = 0)$, calculer le nombre de cellules de l'île contenant la cellule de coordonnées $(0, 3)$.

17. Calculer le nombre d'îles aux temps $t = 0$, $t = 1$, $t = 10$ et $t = t_0$.

Le jeu infini

Vous avez vraiment déjà fini ? ? ?

Dans cette section, l'univers est infini. Le paramètre k définit la taille de la zone initiale.

On conseille pour traiter efficacement cette partie d'utiliser la structure d'ensemble `set` de python. La méthode `add` permet d'ajouter un élément à un ensemble.

18. Combien y a -t-il de cellules vivantes pour $k = 20$ aux instants $t = 0$, $t = 1$, $t = 10$, $t = 100$, $t = 1000$, $t = 10000$? Même question pour $k = 40$ aux instants $t = 0$, $t = 1$, $t = 10$, $t = 100$.

*Comme précédemment, il est possible qu'il existe deux instants $t_0 < t_1$ où l'univers est dans la même configuration. On dit alors que la configuration initiale était dans un **bassin d'attraction**. Malheureusement, on peut prouver qu'il n'est pas possible de concevoir un algorithme qui permette de décider, pour toute configuration, si celle-ci est dans un bassin d'attraction.*

*Un exemple de configuration particulière est le **glisseur**. C'est une configuration qui n'est pas périodique, mais qui après quelques itérations réapparaît à l'identique, décalée de quelques cases.*

En pratique pour le jeu de la vie infini, à partir d'une configuration de départ aléatoire, seuls deux cas sont très probables :

- *Après un nombre suffisant d'itérations, l'univers est périodique (la configuration initiale est dans un bassin d'attraction).*

- *Après un nombre suffisant d'itérations, on peut décomposer l'univers en deux parties : un cœur périodique, et quelques glisseurs qui s'éloignent.*

Prolongements

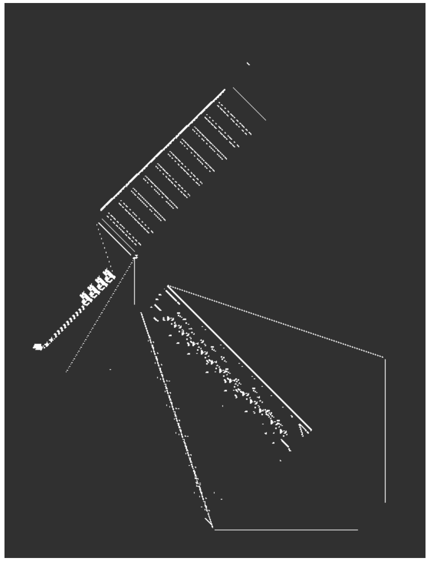
Il a été démontré que tout algorithme pouvait être calculé par le jeu de la vie à partir d'univers suffisamment grand.

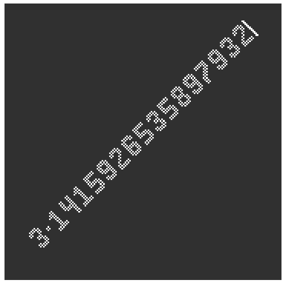
En février 2010, Adam Goucher propose une grille initiale permettant d'écrire les chiffres en bases 10 du nombre d'or et de π ! Pour calculer le nombre π, la configuration de départ comporte 1 389 325 cellules vivantes, et il ne faut pas être trop pressé pour voir apparaître les décimales (complexité en $O(n^6)$ pour avoir n décimales).

Voir

http://pentadecathlon.com/lifeNews/2011/01/phi_and_pi_calculators.html

Voici la configuration de départ et les décimales qui apparaissent en haut à droite.





On connaît aussi des configurations permettant d'obtenir les nombres premiers. Voir l'article de Delahaye

http://www.lifl.fr/~jdelahay/dnalor/Jeudelavie.pdf

Réponses à trouver

```
# 1. u(996) = 1 703 et u(9 996) = 16 562

# 2. 6 595

# 10.  Non

# 11.

#    Pour k = 9,
#    A t = 0, il y a 22 cellule(s) vivante(s)
#    A t = 1, il y a 24 cellule(s) vivante(s)
#    A t = 10, il y a 15 cellule(s) vivante(s)
#    A t = 15, il y a 4 cellule(s) vivante(s)
#
#    Pour k = 20,
#    A t = 0, il y a 127 cellule(s) vivante(s)
#    A t = 1, il y a 153 cellule(s) vivante(s)
#    A t = 10, il y a 71 cellule(s) vivante(s)
#    A t = 100, il y a 41 cellule(s) vivante(s)
#    A t = 130, il y a 10 cellule(s) vivante(s)
#
#    Pour k = 40,
#    A t = 0, il y a 549 cellule(s) vivante(s)
#    A t = 1, il y a 591 cellule(s) vivante(s)
#    A t = 10, il y a 362 cellule(s) vivante(s)
#    A t = 100, il y a 106 cellule(s) vivante(s)
#    A t = 280, il y a 22 cellule(s) vivante(s)

# 13.

# Pour k = 9, t0 = 14 et t1 = 15.
# La période est donc 1 et le temps d'attraction est 14.

# Pour k = 20, t0 = 129 et t1 = 130.
# La période est donc 1 et le temps d'attraction est 129.

# Pour k = 40, t0 = 278 et t1 = 280.
# La période est donc 2 et le temps d'attraction est 278.

# 14.

# Pour k = 9, le nombre de cellules de l'ilôt contenant la cellule
# de coordonnées (0, 3) est 7.

# Pour k = 20, le nombre de cellules de l'ilôt contenant la cellule
# de coordonnées (0, 3) est 59.
```

```
# Pour k = 40, le nombre de cellules de l'ilôt contenant la cellule
# de coordonnées (0, 3) est 19.

# 15.

# Pour k = 9,
# A t =    0, il y a   6 ilôts.
# A t =    1, il y a   1 ilôts.
# A t =   10, il y a   4 ilôts.
# A t =   14, il y a   1 ilôts.

# Pour k = 20,
# A t =    0, il y a  13 ilôts.
# A t =    1, il y a   6 ilôts.
# A t =   10, il y a   6 ilôts.
# A t =  129, il y a   2 ilôts.

# Pour k = 40,
# A t =    0, il y a  50 ilôts.
# A t =    1, il y a  24 ilôts.
# A t =   10, il y a  61 ilôts.
# A t =  278, il y a   6 ilôts.

# 16.

# Pour k = 9, le nombre de cellules de l'île contenant la cellule
# de coordonnées (0, 3) est 22.

# Pour k = 20, le nombre de cellules de l'île contenant la cellule
# de coordonnées (0, 3) est 127.

# Pour k = 40, le nombre de cellules de l'île contenant la cellule
# de coordonnées (0, 3) est 549.

# 17.

# Pour k = 9,
# A t =    0, il y a   1 îles.
# A t =    1, il y a   1 îles.
# A t =   10, il y a   1 îles.
# A t =   14, il y a   1 îles.

# Pour k = 20,
# A t =    0, il y a   1 îles.
# A t =    1, il y a   1 îles.
# A t =   10, il y a   1 îles.
# A t =  129, il y a   2 îles.

# Pour k = 40,
# A t =    0, il y a   1 îles.
```



```
# A t =    1, il y a   2 îles.
# A t =   10, il y a   2 îles.
# A t =  278, il y a   5 îles.

# 18.

# Pour k = 20,
# A t = 0, il y a 127 cellule(s) vivante(s).
# A t = 1, il y a 143 cellule(s) vivante(s).
# A t = 10, il y a 78 cellule(s) vivante(s).
# A t = 100, il y a 85 cellule(s) vivante(s).
# A t = 1000 et t=10000, il y a 87 cellule(s) vivante(s).

# Pour k = 40,
# A t = 0, il y a 549 cellule(s) vivante(s).
# A t = 1, il y a 564 cellule(s) vivante(s).
# A t = 10, il y a 368 cellule(s) vivante(s).
# A t = 100, il y a 236 cellule(s) vivante(s).
# A t = 1000, il y a 158 cellule(s) vivante(s).
```