

Initiation aux automates cellulaires.

Présentation : (screen support page 1)

Les automates cellulaires sont de petits programmes, petits robots qui avec des règles, des algos simples

a/ permettent d'observer l'émergence de structures ou de processus depuis un comportement chaotique. (c'est le cas de la fourmi de Langton que nous allons voir en détail).

b/ créer rapidement toute sorte d'objets et de comportements interagissant entre eux parfois,, (Le jeu de la vie de Conway). Avec a/ et b/ on voit un lien avec la physique, la biologie et la philosophie.

c/ permettent de simuler l'auto-reproduction ou la propagation d'information, (Travaux de Von Neuman ou La boucle de Langton dont j'espère faire la démo en JS). Lien très fort avec la biochimie.

d/ d'autre part il y a dans certains cas, la création de motifs esthétiques, de mosaïques de fractales.

Lien avec l'art donc.

Règles de la Fourmi originale : (screen support page 2)

On a un tableau (1 matrice 2D) avec au croisement des lignes et colonnes des cases blanches ou noires. En théorie le tableau est infini, en pratique dépend de la taille de l'écran et grandeur des cases.

On dessine et pose une fourmi sur une des cases.

L'algorithme est simplissime :

Si la fourmi est sur une case

blanche => elle tourne de 90° dans le sens des aiguilles d'une montre.

noire => elle tourne de 90° dans le sens inverse des aiguilles).

Puis dans les 2 cas :

Elle avance d'une case dans la direction où elle est.

Et elle inverse la couleur de la case. (case blanche devient noire, case noire devient blanche).

On va démarrer avec la fourmi +ou- au milieu de l'écran et le tableau avec que des cases blanches.

Pratique :

Aller sur [lien github abrégé](#) + download zip

Nous allons faire tourner la version basique mais aussi explorer plusieurs variantes.

Il y a un "moteur" qui gère le graphisme, le visuel, bref une programmation assez laborieuse mais pas intéressante pour ce qui nous concerne.

J'ai donc décidé de vous faire piloter le cœur de l'application et ses variantes avec des paramètres qui se trouvent dans un fichier java-script : index.js

Ouvrir ./js/index.js

expliquer les paramètres :

```
Driver={}
```

```
Driver.period=1;
```

```
Driver.maxStep=4 ;
```

```
Driver.colorChain
```

```
    avec initColorChain=initColorChain1;
```

```
    Détailler initColorChain ()
```

Chrome à 100%

--- Tester --- en double cliquant sur [index.html](#)

```
Driver.maxStep=8 ;
```

```
Driver.period=4;
```

--- Tester ---

// 4 et 8 => ça semble symétrique. C'est normal car la règle est symétrique !

```
Driver.maxStep=96 ;
```

```
Driver.period=12;
```

--- Tester ---

// Encore symétrique... de traviole !

```
Driver.maxStep=472 ;
```

```
Driver.period=24;
```

--- Tester ---

// La fourmi a dessiné une jolie fleur ! On ne pouvait déjà pas le prévoir en ne regardant que la règle.

```
Driver.maxStep=9999;
```

```
Driver.period=100;
```

--- Tester ---

// La fourmi s'énerve... elle détruit sa jolie fleur ! Ses mouvements semblent complètement chaotique !

// C'est le grand porteNinWak !

```
Driver.maxStep=-1;  
Driver.skip=true;  
Driver.skipStart=9000;  
Chrome à 100%
```

--- Tester ---

// Attention !! A partir de +ou- 10100 il va se passer un truc ! :

// Elle fait un motif régulier et continue...

// Question :

peut-elle retourner vers le chaos ?

// Réponse :

Elle continuera à l'infini car malgré une impression chaotique, les règles sont déterministes et facilement prévisible quand il n'y a plus que des cases blanches.

On appelle ça l'autoroute !

Récapitulatif :

1 à 472 : Symétries.

473 à 10100 : Chaos

10100 à infini : Autoroute

On a là une véritable émergence d'une structure avec des règles simples.

Et l'émergence d'une régularité après un chaos !

Moi ça m'interpelle et j'espère vous avoir "accrochés".

Nous allons maintenant nous amuser à changer légèrement les règles et voir d'autres motifs et apparitions.

// Voir le code de initColorChain2()

```
initColorChain=initColorChain2;
```

```
Driver.maxStep=-1;
```

```
Driver.skip=false;
```

```
Driver.skipStart=0;
```

--- Tester rapide ---

Q: Que remarquez vous ? ...

R: symétrie réelle

Pour mieux voir :

```
Driver.skip=true;
```

```
Driver.skipMethod="pauseOnZero";
```

--- Tester ---

Les symétries sont évidentes et toutes différentes

Essayons une autre symétrie moins expansive:

```
initColorChain=initColorChain3;
```

--- Tester ---

// symétrie moins expansive:avec une grande fréquence de retour à l'origine.

Essayons une règle NON symétrique :

```
initColorChain=initColorChain6;
```

```
Driver.skipStart=10000;
```

```
Driver.skip=true;
```

```
Driver.skipValue=1000;
```

```
Driver.skipMethod="";
```

--- Tester ---

// Il semble qu'on a trouvé une règle ne générant PAS d'autoroute NI symétrie et par contre chaotique.

NOTONS qu'on n'a jamais trouvé de cas cyclique ET non expansive. (fourmi qui tourne en rond!).

Je l'ai testée jusqu'à plus de 2 millions de cycles sans voir d'autoroute !! à voir donc, car ...*

Essayons la règle 15 elle aussi NON symétrique :

```
initColorChain=initColorChain15;
```

--- Tester ---

// Elle semble comme la 6 Mais vous pourrez le tester une autoroute apparaît au cycle 258.350 !!!

La 7 est sympa, elle est peu expansive comme la 3 mais crée une enveloppe qui s'étend lentement.

Voyons une dans la catégorie esthétique la 8.

```
initColorChain=initColorChain8;
```

```
Driver.skipStart=0 ;
```

```
Driver.skip=true;
```

```
Driver.skipValue=2;
```

```
Driver.skipMethod="pauseOnZero";
```

--- Tester ---

// Elle est symétrique, expansive et fait régulièrement des motifs flash de couleur sur une base noir,blanc,gris,jaune.

On supprime le saut pour analyser :

```
Driver.skip=false;
```

--- Tester ---

// On voit qu'elle construit un motif coloré puis elle le déconstruit en revenant au centre ... Reste un temps plus ou moins long très proche du centre. Ce qui donne cette impression de flash, renforcée quand on fait l'arrêt sur image au centre.

On arrête mais il est intéressant artistiquement de la laisser grossir avec chrome à 33% !

NB : J'appelle "autoroute" toute structure répétitive même si elle a une forme carrée, en triangle, etc.

La 9 est expansive, fait une enveloppe autour qui semble presque faire une autoroute mais reste chaotique.

La 10 patatoïde sans autoroute (?)

Vous les testerez...

Essayons la 11

```
initColorChain=initColorChain11;
```

--- Tester ---

// Elle fait une autoroute immédiatement.

La 12 crée des bourgeons. La 13 crée une autoroute à 1000 ;

Voyons la 14

```
initColorChain=initColorChain14;
```

```
Driver.skip=true;
```

```
Driver.skipMethod="pauseOnZero";
```

--- Tester ---

// elle est symétrique et expansive rapidement et assez jolie. Voir la photo à 821.878 dans /joint

passer 16,17, 18 car il faut attendre beaucoup d'itérations.

Les super variantes :

```
initColorChain=initColorChain19;
```

```
Driver.initRotation=toRight*2 ;
```

```
Driver.skip=true;
```

```
Driver.skipMethod="";
```

```
Driver.skipStart=30000;
```

--- Tester ---

// Après full chaos, après 30.000, elle fait une autoroute bien plus complexe, qui comporte plus de 2000 itérations entre chaque forme identique. On dirait une sorte de vilebrequin !

Tester et/ou montrer photos des résultats.

// Test du début de la 20

```
initColorChain=initColorChain20;
```

```
Driver.initRotation=up;
```

```
Driver.skipStart=9000;
```

--- Tester ---

// Jolie cravate ! Regardons les images 20 et 21 dans joint.

// Test du début de la 22

```
initColorChain=initColorChain22;
```

```
Driver.skipStart=10000;
```

--- Tester ---

// En attendant regardons l'image dans joint.

```
// Test de la 23
initColorChain=initColorChain23;
Driver.skip=true;
Driver.skipValue=1000;
Driver.skipStart=3000;
--- Tester ---
// Démarre immédiatement une "autoroute" carrée !
```

[// Voir la 24 dans joint](#)

// Autoroute 2D un carré avec une spirale à l'intérieur ! (à partir de 15.000). Elle agrandit le périmètre de son carré, tout en construisant 1 "cocon" sur un des bords, puis un moment fait une oblique pour rejoindre le côté adjacent.

Si on veut la paramétrer :

```
initColorChain=initColorChain24;
Driver.skip=true;
Driver.skipValue=1;
Driver.initRotation=up;
Driver.skipStart=2000;
```

[// Voir les artistiques de 27 à 31 :](#)

La 27 a déjà de belle couleurs

de 28 à 29, il peut être intéressant de faire des recherches de couleurs (photoshop)

La 31 est à la fois artistique et a une structure d' "autoroute" 2D.

Si on la pousse jusqu'à plusieurs millions le chaos du milieu devient de plus en plus petit, relativement à l'autoroute !

FIN (ou voir Langton's loop)