
Chap IX. Automates cellulaires

I. Introduction et exemple du jeu de la vie

II. Définitions

III. Quelques exemples simples

IV. Automates LGA (Lattice Gas Automata)

V. Autres exemples

I. Introduction et exemple du jeu de la vie

Idée initiale (Neumann, 1940) :

Trouver un système/algorithme se reproduisant, sans perte de complexité, à l'aide de règles simples

Concrétisation par les automates cellulaires :

- Univers constitué de cellules discrètes représentant **l'état** « **microscopique** » du système
 - Lois d'évolution des cellules (*lois simples et communes à toutes les cellules*) définies **localement** à partir de l'état des cellules voisines
- ➡ Comportement global « **macroscopique** » complexe !

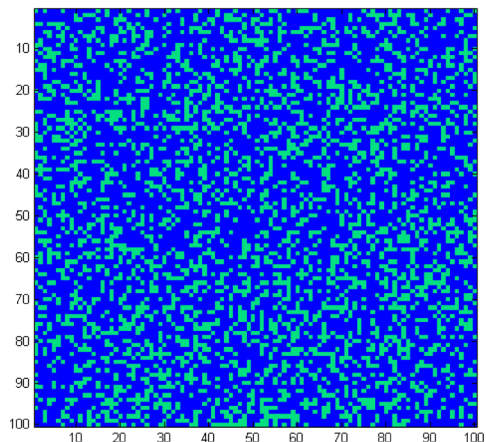
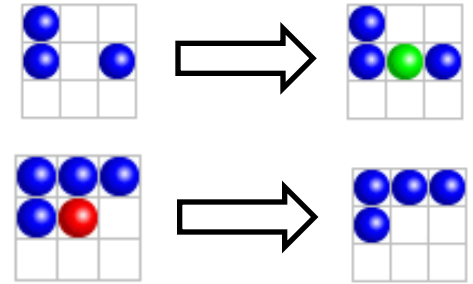
Un automate cellulaire simple (*lois microscopiques*) peut faire émerger des comportements (*lois macroscopiques*) d'une complexité infinie...

Exemples :

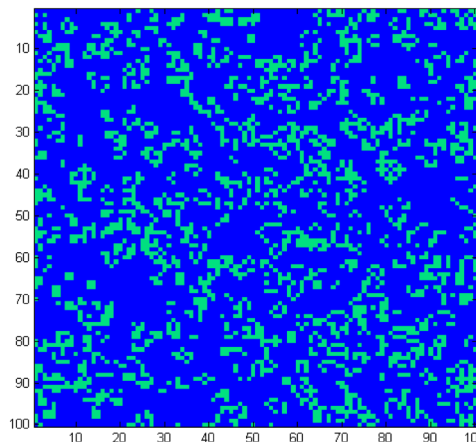
- Simulation du comportement des éléments d'un ordinateur (*transistors et câbles*)
- Simulation des équations de Navier Stokes (*Lattice Gaz Boltzmann*)
- ...

La « qualité » de la simulation (*échelle macro*) dépendra de la « qualité » des règles d'évolution choisies (*échelle micro*) !

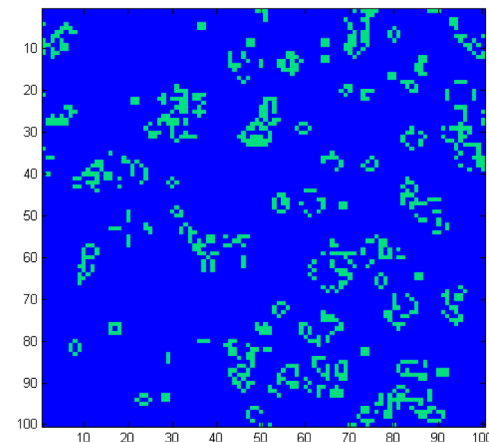
- **Univers** = Maillage 2D carré
- **États** des cellules : 1 = cellule vivante
0 = cellule morte
- **Règles d'évolution d'une génération/d'un instant à l'autre** :
 - 1 cellule morte entourée de 3 vivantes devient vivante
 - 1 cellule vivante entourée par 2 ou 3 voisines vivantes le reste, sinon elle meurt



génération 1



10



100

II. Définitions

Un **automate cellulaire** est constitué :

- d'un **réseau régulier de cellules**. Une cellule (*un site*) sera noté \vec{r}
- d'un ensemble de **variables attachées à chacune des cellules** et donnant l'état de celles-ci à un temps discret $t = 0, 1, 2, 3...$

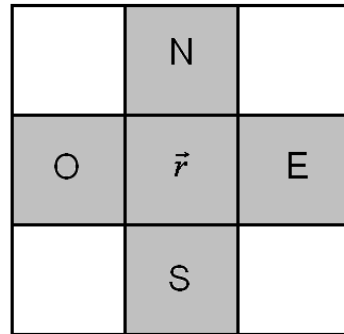
$$\Phi(\vec{r}, t) = \{ \Phi_1(\vec{r}, t), \Phi_2(\vec{r}, t), \Phi_3(\vec{r}, t), \dots, \Phi_m(\vec{r}, t) \}$$

- de **règles R** qui spécifient l'évolution temporelle des états de chaque cellule :

$$\Phi(\vec{r}, t + \tau) = R \left(\Phi(\vec{r}, t), \Phi(\vec{r} + \vec{\delta}_1, t), \Phi(\vec{r} + \vec{\delta}_2, t), \dots, \Phi(\vec{r} + \vec{\delta}_q, t) \right)$$

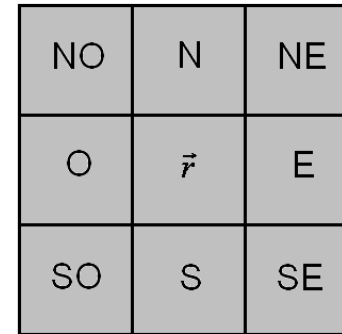
$\vec{r} + \vec{\delta}_k$ désigne les k cellules appartenant au **voisinage** de \vec{r}

2 voisinages très courants :



Voisinage de Neumann

$k = 4$ voisins



Voisinage de Moore

$k = 8$ voisins

Plus rarement :

- le voisinage peut atteindre d'autres cellules environnantes plus éloignées
- le réseau peut être constitué de cellules hexagonales

Les règles d'évolution et la définition du voisinage étant fixes, l'automate cellulaire ainsi créé est déterministe :

son évolution à partir d'un état initial est **unique** !

On peut modérer les règles de transition en leur donnant une composante aléatoire, l'automate devenant alors probabiliste (ou stochastique)

Avantages des automates cellulaires :

Modéliser un système complexe par ses lois microscopiques de base permet de ne pas passer par les EDP classiques...

La structure même du problème est adaptée au traitement numérique par parallélisation massive

Inconvénients des automates cellulaires :

Nécessité de traiter des très grosses populations de cellules pour avoir une représentativité statistique du système

Tous les problèmes, et de loin, ne se traitent pas par cette approche

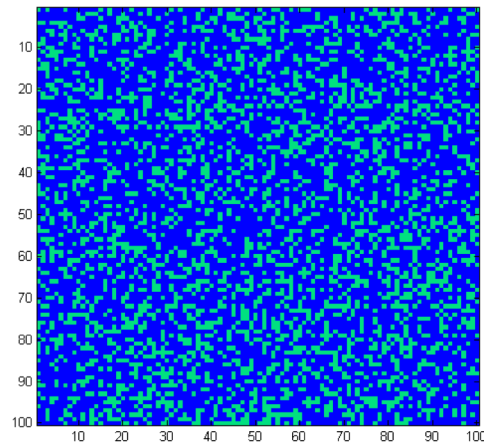
Automate : maillage 2D ; chaque cellule a la valeur $\Phi = 0$ (*mort*) ou 1 (*vivant*)

Modèle de croissance :

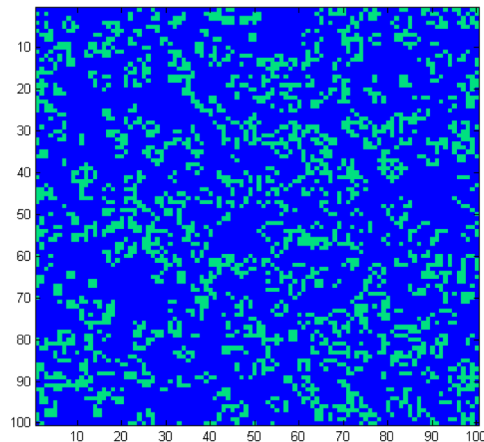
$$\text{si } \sum_k \Phi(\vec{r} + \vec{\delta}_k, t) = 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8$$

$$\text{et si } \Phi(\vec{r}, t) = 1 \quad \text{alors} \quad \Phi(\vec{r}, t+1) = 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

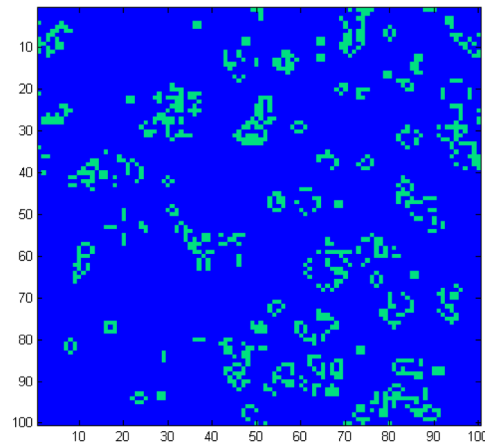
$$\text{et si } \Phi(\vec{r}, t) = 0 \quad \text{alors} \quad \Phi(\vec{r}, t+1) = 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$



génération 1



10



100

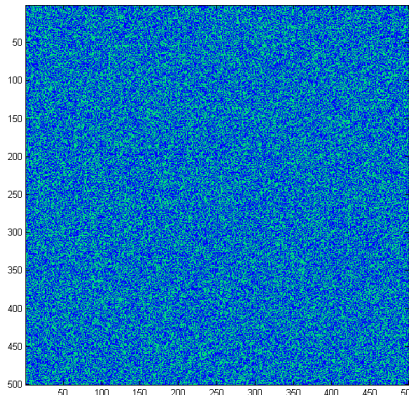
Peu d'intérêt autre qu'esthétique ...

Automate : maillage 2D ; chaque cellule prend la valeur $\Phi = 0$ ou 1

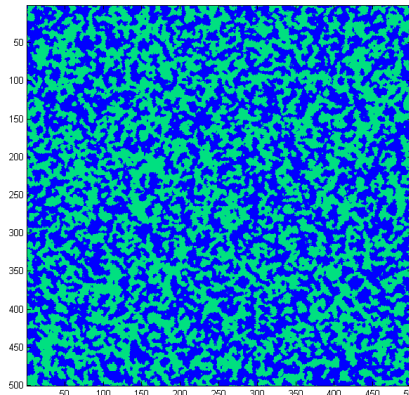
Modèle de croissance = règle de la majorité sur un voisinage de Moore

$$\text{si } \sum_k \Phi(\vec{r} + \vec{\delta}_k, t) + \Phi(\vec{r}, t) = \begin{matrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{matrix}$$

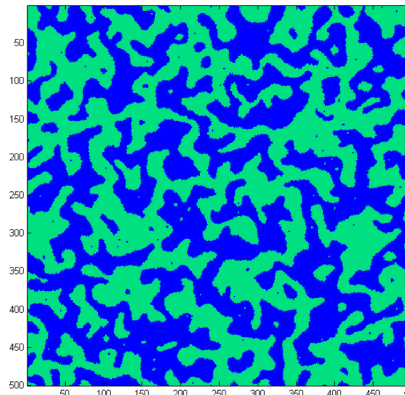
$$\text{alors } \Phi(\vec{r}, t+1) = \begin{matrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \end{matrix}$$



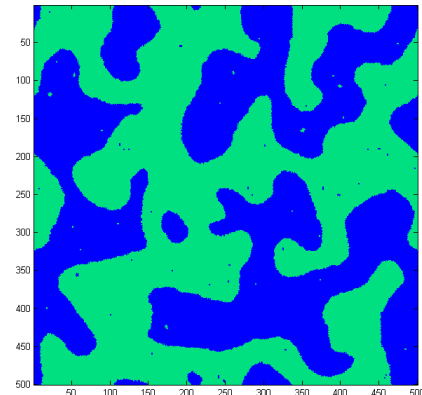
génération 1



10



100



1000

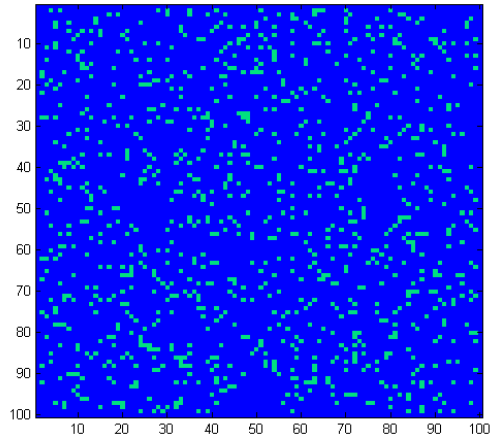
La vitesse de croissance normale de l'interface obtenue est proportionnelle à son rayon de courbure ...

Simulation réaliste d'un recuit, pourtant aucune donnée physique utilisée...

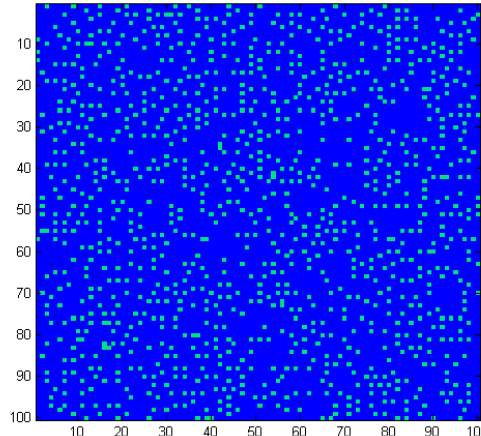
Automate : maillage 2D ; chaque cellule prend la valeur $\Phi = 0$ ou 1

Modèle de croissance = règle de la compétition sur un voisinage de Moore

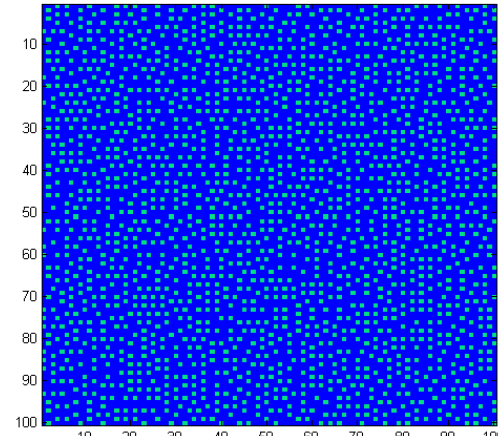
$$\begin{array}{l} \text{si } \sum_k \Phi(\vec{r} + \vec{\delta}_k, t) = 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \\ \text{alors } \Phi(\vec{r}, t+1) = 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \end{array}$$



génération 1



5



100

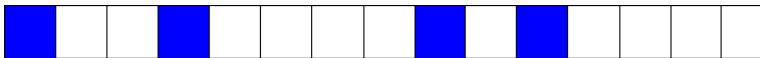
Représentation acceptable de la différenciation des cellules dans les embryons

Automate : maillage 1D ; chaque cellule prend la valeur $\Phi = 0$ ou 1

Règle de l'automates : $\{ \Phi(\vec{r} - 1, t) \Phi(\vec{r}, t) \Phi(\vec{r} + 1, t) \} \rightarrow \Phi(\vec{r}, t + 1)$

$(111) \rightarrow 1$; $(110) \rightarrow 0$; $(101) \rightarrow 1$; $(100) \rightarrow 1$

$(011) \rightarrow 1$; $(010) \rightarrow 0$; $(001) \rightarrow 0$; $(000) \rightarrow 0$

Cas d'une circulation fluide : 

Cas d'une circulation dense : 

Améliorations possibles :

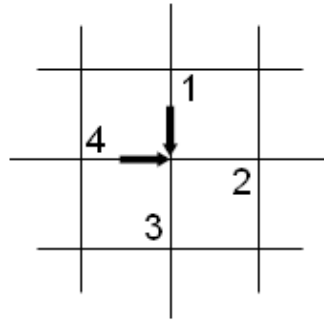
- 2D avec gestion des carrefours et priorités
- Accélération et freinages aléatoires ...

Applications : trafic urbain, évacuation d'un immeuble ...

IV. Automates LGA (Lattice gas automata)

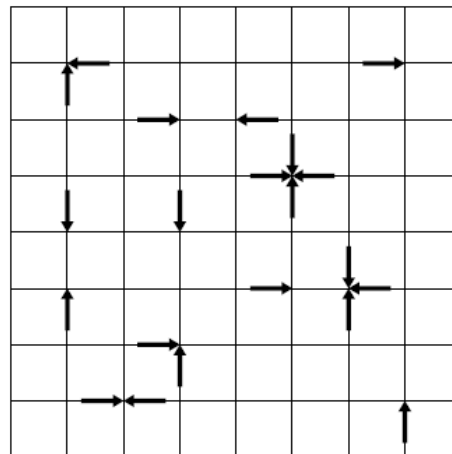
Les automates de type « lattice gas automata » forment une classe particulière : ils représentent des particules se mouvant sur une matrice carrée, de manière à imiter la dynamique moléculaire d'un gaz

$$\Phi(\vec{r}, t) = \{1, 0, 0, 1\}$$



Voisinage type Neumann

$\Phi(\vec{r}, t) = \{1, 0, 0, 1\}$ indique la présence de deux particules se déplaçant respectivement selon les directions 1 et 4



Exemple de configuration typique durant une simulation de type LGA

Règles d'évolution :

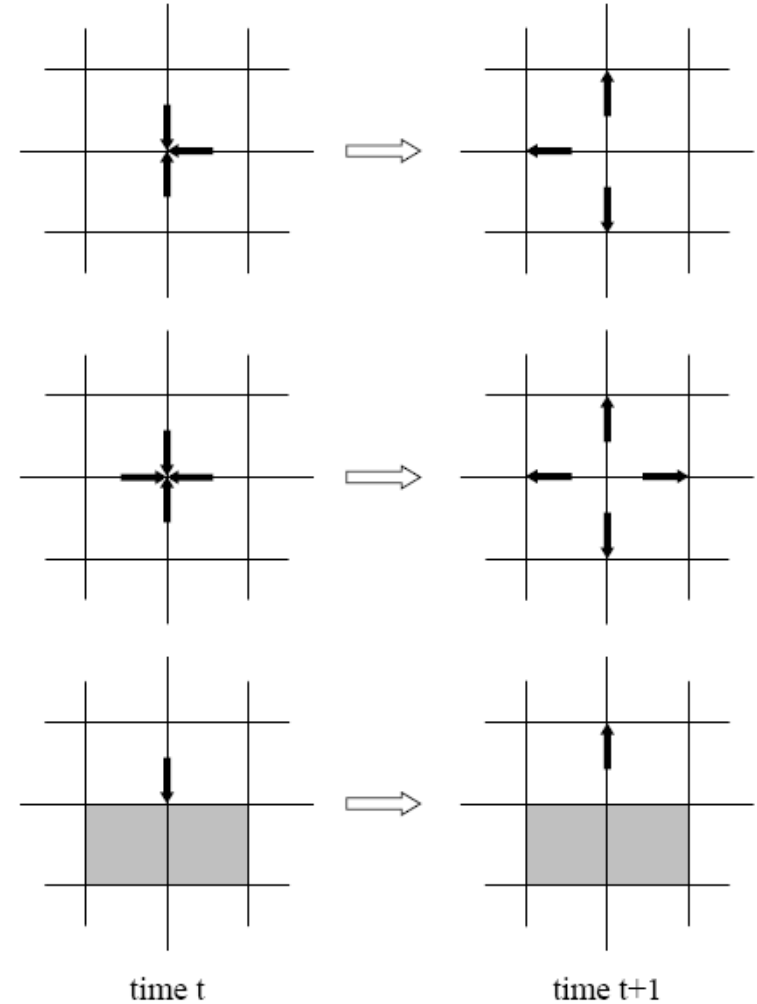
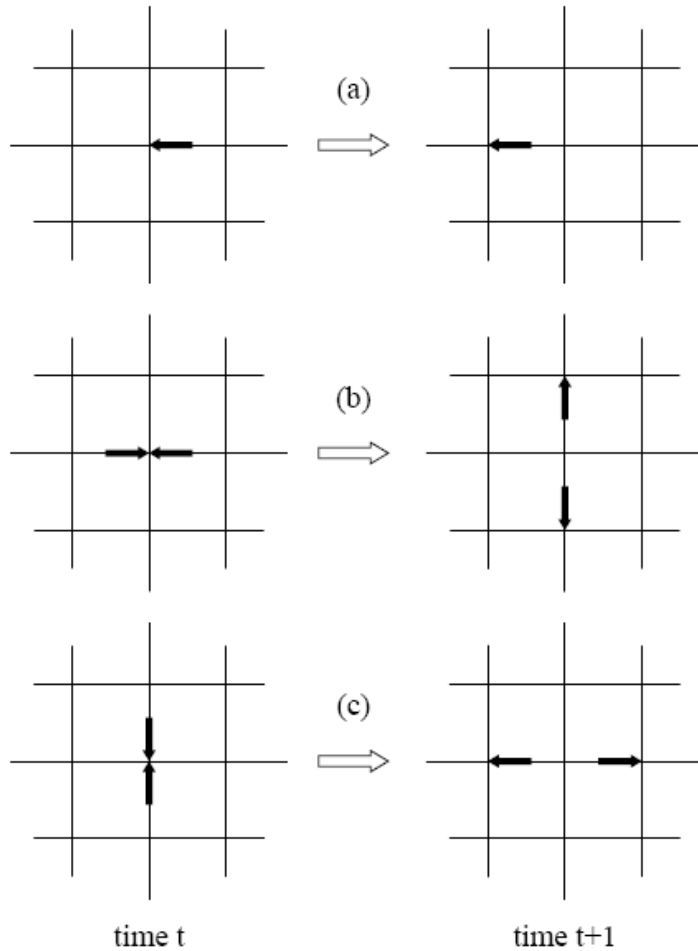
à chaque étape - gérer les collisions site par site

PUIS

- faire se déplacer les particules d'un site à l'autre
en n'oubliant pas de gérer les rebonds / collisions
sur les obstacles

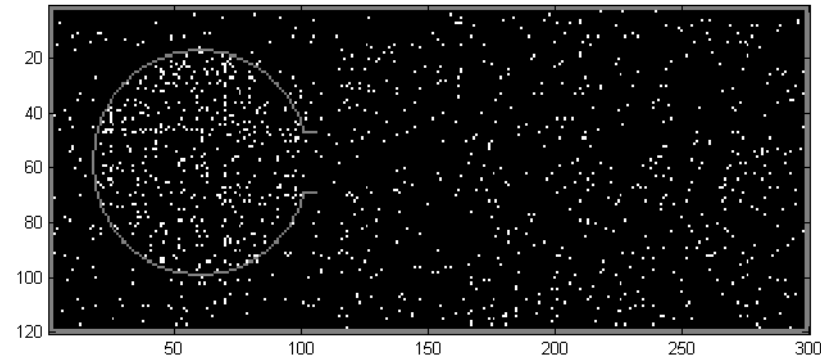
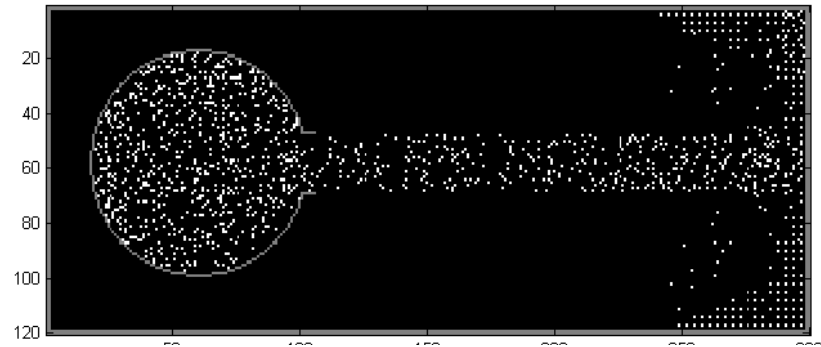
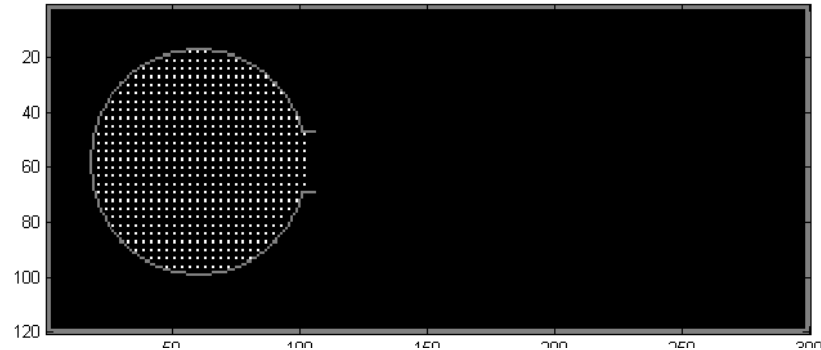
Simulation d'un gaz dont le libre parcours moyen est grand (*vide poussé*)

Règles d'évolution :

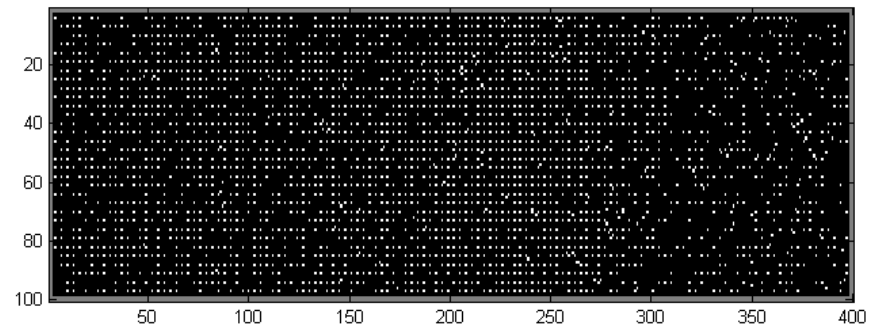
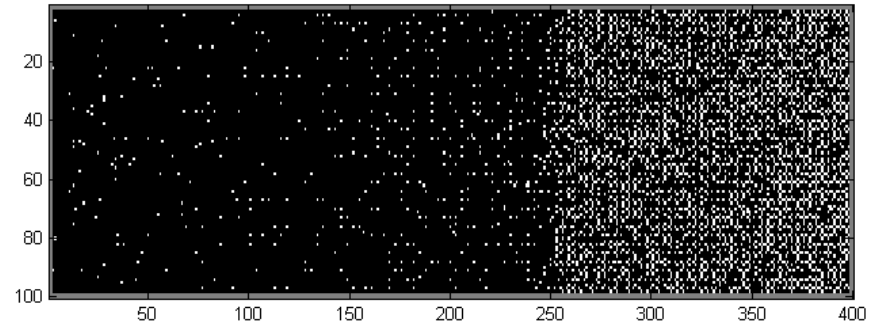
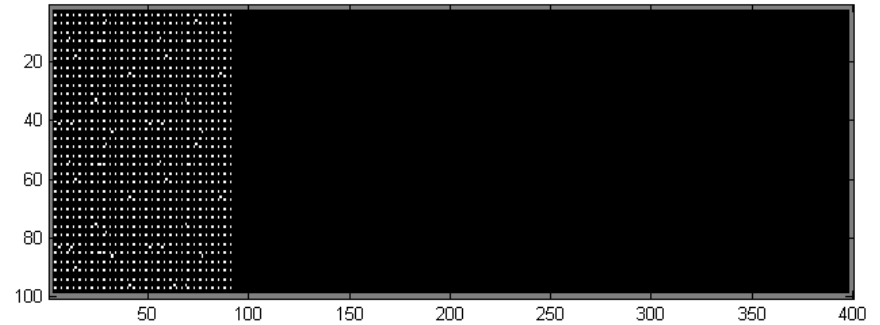


Simulation d'un gaz dont le libre parcours moyen est grand (*vide poussé*)

vidange d'une sphère



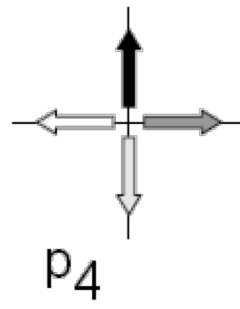
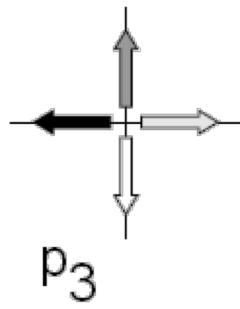
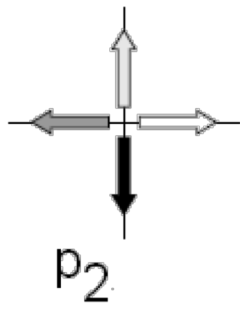
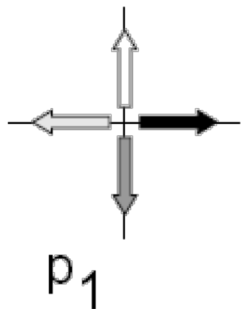
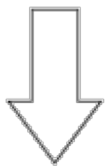
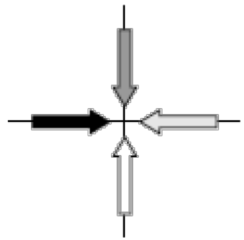
tube de choc



Simulation d'un gaz dont le libre parcours moyen est court (pression atmosphérique)

Cette fois le nombre de chocs inter-particules est grand

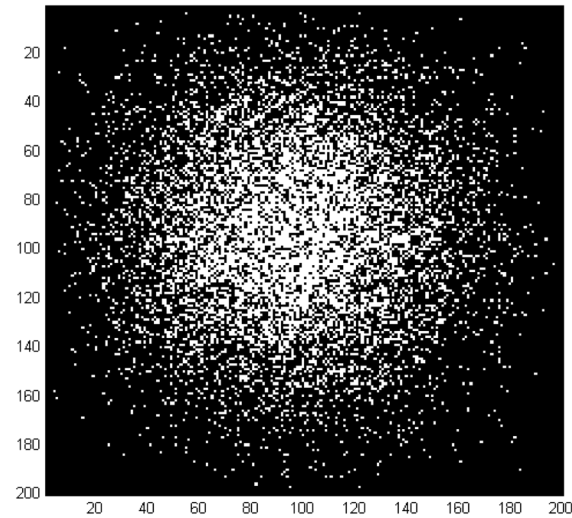
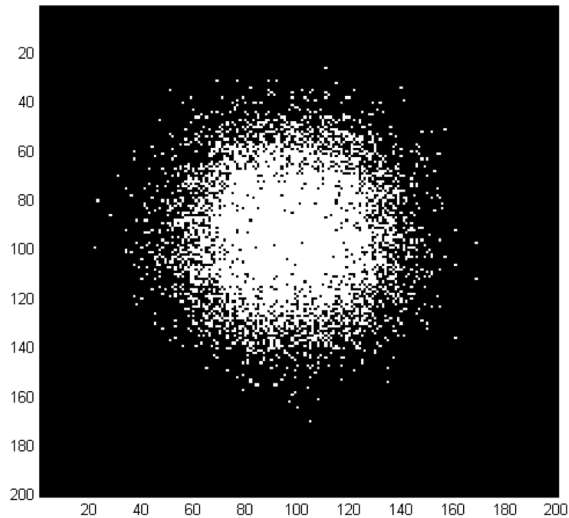
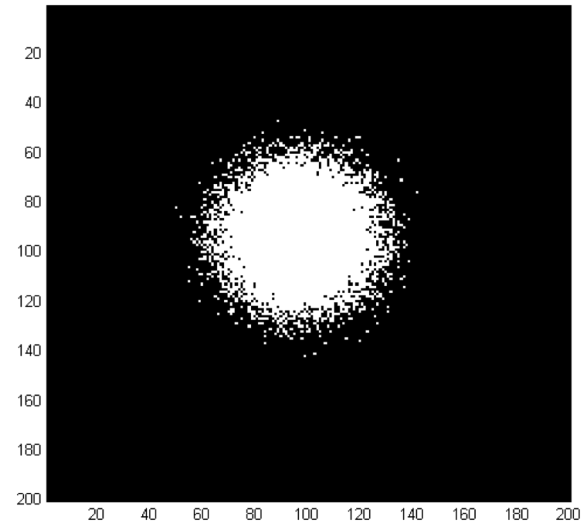
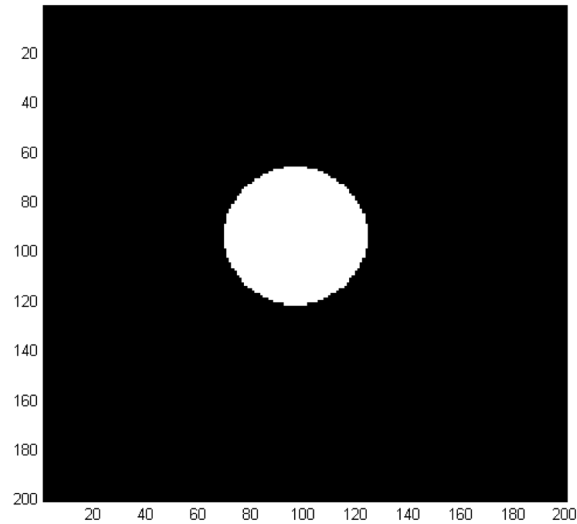
➡ on considère que chaque particule prend une direction aléatoire à chaque pas de temps ET que le principe d'exclusion empêche deux particules de partir par la même direction

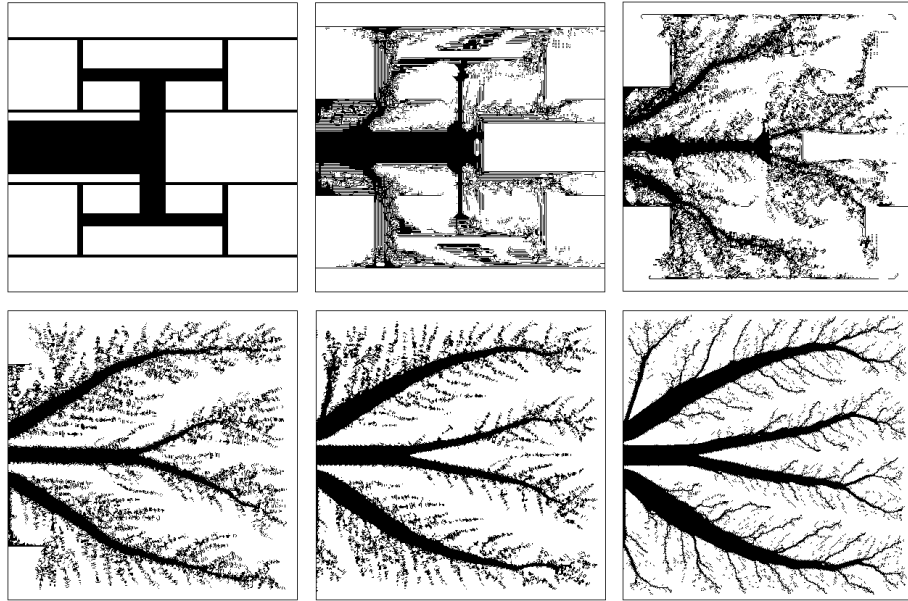


Méthode de la marche au hasard ou « random walk » :

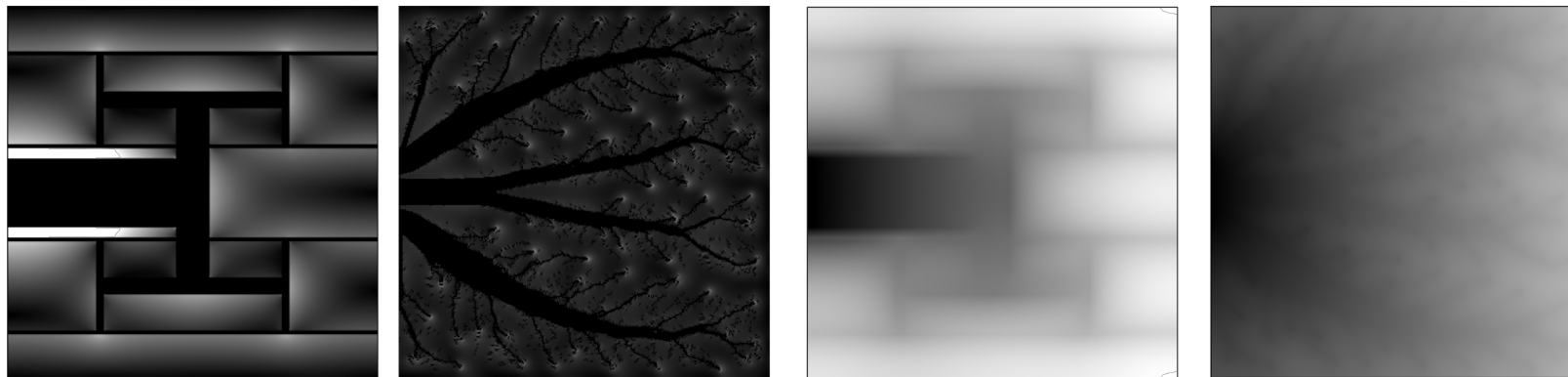
Les collisions sont simulées en prenant une des quatre configurations suivantes avec une probabilité de 25% pour chaque site (les flèches peuvent représenter des directions vides)

Diffusion d'un spot de matière





Optimisation assistée par automate cellulaire



Initial thermal gradients

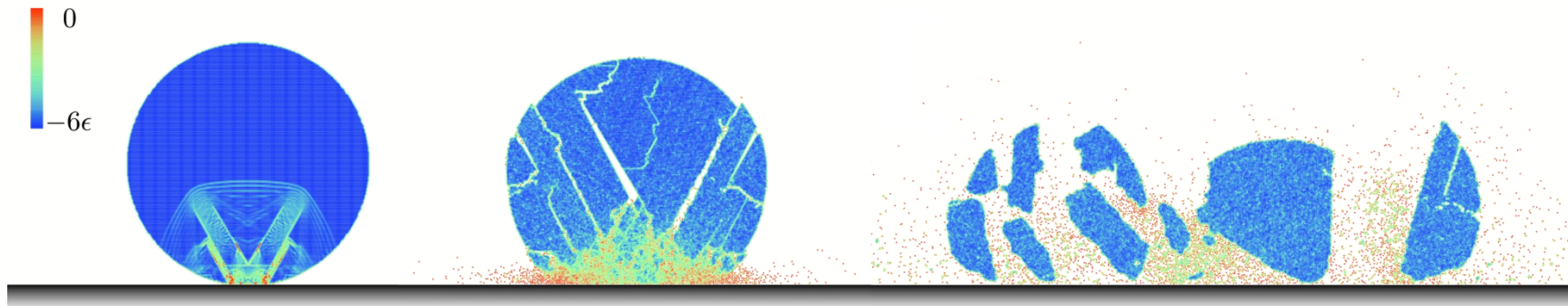
Thermal gradients at convergence

Initial temperature

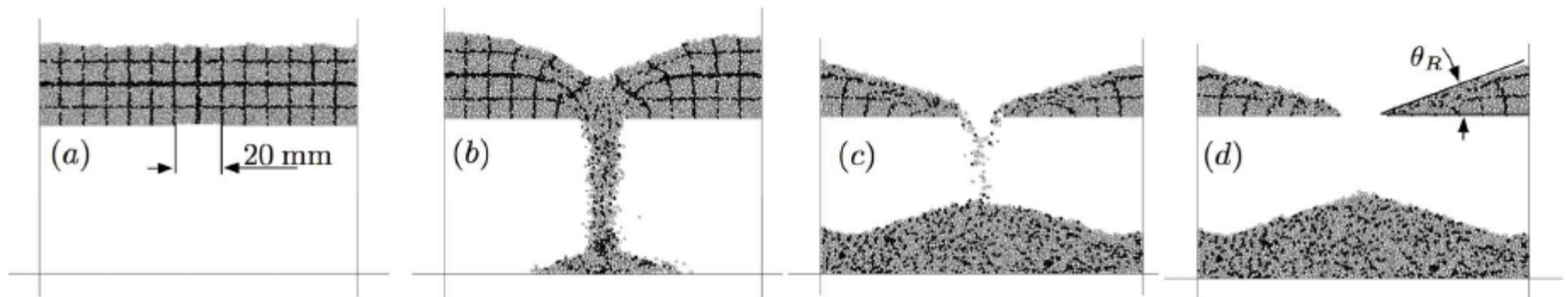
Temperature at convergence

□ Maximum ■ Minimum

□ Maximum ■ Minimum



Écoulement d'une poudre



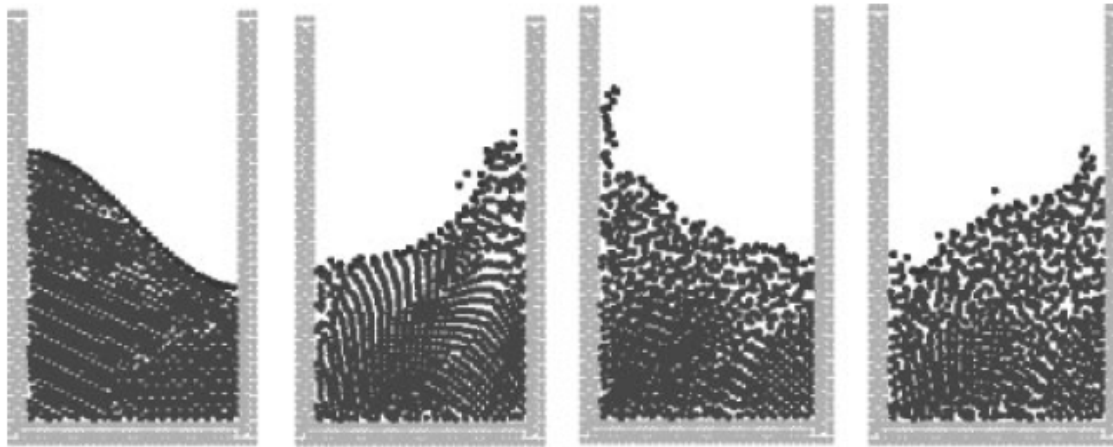


Figure 5. Sloshing: Different time step for large amplitudes.