Chapitre 5 : Les automates cellulaires (AC) / Cellular Automata (CA)

B. Chopard et M. Droz : Cellular Automata Modeling of Physical Systems, Cambridge University Press, 1998.

http://cui.unige.ch/~chopard/CA/Animations/img-root.html

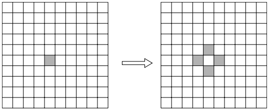
5.1 Définition et concepts de bases

What is a Cellular Automata ?

- Mathematical abstraction of the real world, modeling framework
- Fictitious Universe in which everything is discrete
- But, it is also a mathematical object, new paradigm for computation
- Elucidate some links between **complex systems**, **universal computations**, **algorithmic complexity**, **intractability**.

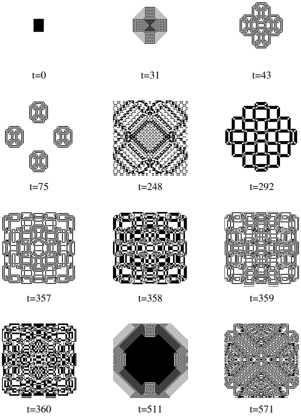
Example : the Parity Rule

- Square lattice (chessboard)
- Possible states $s_{ij} = 0, 1$
- Rule : each cell sums up the states of its 4 neighbors (north, east, south and west).
- If the sum is even, the new state is $s_{ij} = 0$; otherwise $s_{ij} = 1$



Generate "complex" patterns out of a simple initial condition.

Pattern generated by the Parity Rule

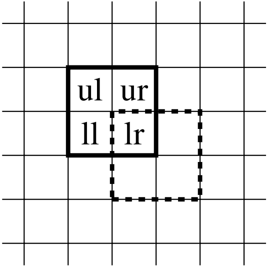


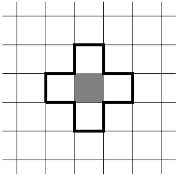
CA Definition

- Discrete space A : regular lattice of cells/sites in d dimensions.
- Discrete time
- Possible states for the cells : discrete set S
- Local, homogeneous **evolution rule** Φ (defined for a neighborhood $\mathcal{N}$).
- Synchronous (parallel) updating of the cells
- Tuple : $< A, S, \mathcal{N}, \Phi >$

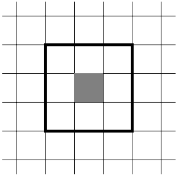
Neighborhood

- von Newmann
- Moore
- Margolus
- ...





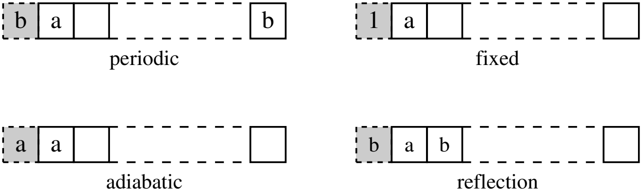
(a)



(b)

Boundary conditions

- periodic
- fixed
- reflexive
- ....



The inital condition is also to be specified

Generalization

- Stochastic CA
- Asynchronous update : loss of parallelism, but avoid oscillations
- Non-uniform CA

Stochastic CA

We apply different rules with given probabilities. For instance

$$s_i(t+1) = \begin{cases} \Phi_1(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{with probability } p \\ \Phi_2(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{with probability } 1-p \end{cases}$$

This could be reformulated as a standard CA as

$$\begin{aligned} s_i(t+1) &= \begin{cases} \Phi_1(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{if } b_i(t) = 1 \\ \Phi_2(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{if } b_i(t) = 0 \end{cases} \\ b_i(t+1) &= \Phi_3(b_{i-1}(t), b_i(t), b_{i+1}(t)) \end{aligned}$$

where Φ_3 is a CA rules that produces **random bits** (see examples below)

Asynchronous CA

The update of the cells are not done all at the same time as done in the norlal case (parallel or synchronous updating)
Sequential updating :

$$\begin{aligned} s_1(t+1) &= \Phi(s_n(t), s_1(t), s_2(t)) \\ s_2(t+1) &= \Phi(s_1(t+1), s_2(t), s_3(t)) \\ s_3(t+1) &= ..... \end{aligned}$$

A random sequential updating is desirable, though.
A random sequential updating can be reformulated as a standard CA

$$\begin{aligned} s_i(t+1) &= \begin{cases} \Phi(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{if } b_i(t)=1 \\ s_i(t) & \text{if } b_i(t)=0 \end{cases} \\ b_i(t+1) &= \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p \end{cases} \end{aligned}$$

non-uniform CA

The rule is not unique for all cells. For instance, some cells are updated with rule Φ_1, others with rule Φ_2.

$$s_i(t+1) = \begin{cases} \Phi_1(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{if } i \in A \\ \Phi_2(s_{i-1}(t), s_i(t), s_{i+1}(t)) & \text{otherwise} \end{cases}$$

can be expressed as a standard CA as

$$\begin{aligned} s_i(t+1) &= b_i(t)\Phi_1 + (1 - b_i(t))\Phi_2 \\ b_i(t+1) &= b_i(t) \end{aligned}$$

And $b_i(t=0)$ are such that $b_i(0) = 1$ if $i \in A$ and $b_i(0) = 0$ otherwise.

Implementation of the evolution rule

On-the-fly calculation

$$
\begin{aligned}
s_{ij}(t+1) &= s_{i-1,j}(t) + s_{i+1,j}(t) \\
&\quad + s_{i,j-1}(t) + s_{i,j+1}(t) \bmod 2
\end{aligned}
$$

Lookup table

0000	0
0001	1
0010	1
0011	0
0100	1
0101	0
0110	0
0111	1
1000	1
1001	0
1010	0
1011	1
1100	0
1101	1
1110	1
1111	0

Number of possible rules

There is a finite number of “possible universes :”

- Let m be the number of states per cell and k the number of neighbors.
- There are m^k configurations of the neighborhood (16 in the previous paage)
- For each of them, there are m possible values for the rule (2 in the previous example).
- Therefore there are m^{m^k} possible rules

Historical notes

- Origin of the CA's (1940s) : John von Neumann and S. Ulam
- Design a better computer with self-repair and self-correction mechanisms
- Simpler problem : find the logical mechanisms for self-reproduction :
- Before the discovery of DNA : find an algorithmic way (transcription and translation)
- Formalization in a fully discrete world
- Automaton with 29 states, arrangement of thousands of cells which can self-reproduce
- Universal computer

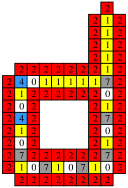
Langton's CA

Christopher G. Langton, Physica 10D (1984) 135-144

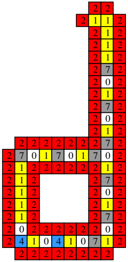
- Simplified version (8 states).
- Not a universal computer
- Structures with their own fabrication recipe
- Using a reading and transformation mechanism

Demo: C++/Langton/langton 300 300 800 801

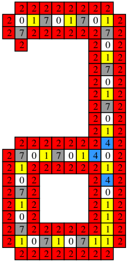
Langton's CA : basic cell replication



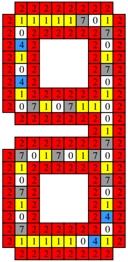
(time 0)



(time 35)



(time 75)

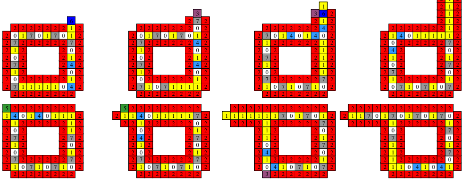


(time 125)

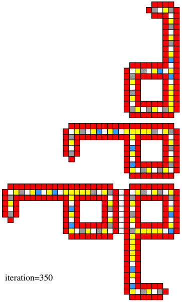
[Animation step by step]

[Lookup table]

Langton's Automaton : spatial and temporal evolution

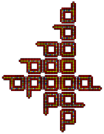


iteration=137 iteration=140 iteration=145 iteration=150



iteration=350

iteration 600



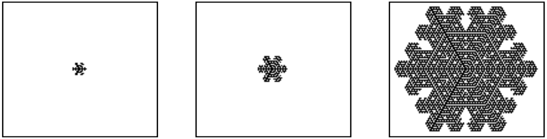
Langton's CA : some conclusions

- Not a biological model, but an algorithmic abstraction
- Reproduction can be seen from a mechanistic point of view (Energy and matter are needed)
- No need of a hierarchical structure in which the more compicated builds the less complicated
- Evolving Hardware.

5.2 CA as a mathematical abstraction of reality

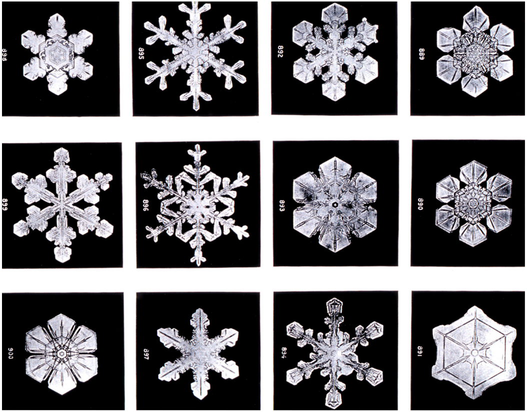
- Several levels of reality : macroscopic, mesoscopic and microscopic.
- The macroscopic behavior depends very little on the details of the microscopic interactions.
- Only “symmetries” or conservation laws survive. The challenge is to find them.
- Invent a fictitious world, with its own microscopy, and particularly easy to simulate on a (parallel) computer, with the desired macroscopic behavior.
- Simple, flexible, intuitive, efficient

A Caricature of reality



What is this ?

The real thing

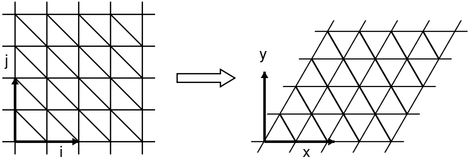


Wilson Bentley, From Annual Summary of the "Monthly Weather Review", 1902.

Snowflakes model

- Very rich reality, many different shapes
- Complicated true microscopic description
- Yet a simple growth mechanism can capture some essential features

• A vapor molecule solidifies ($\rightarrow$ice) if one and only one already solidified molecule is in its vicinity
• Growth is constrained by 60^o angles



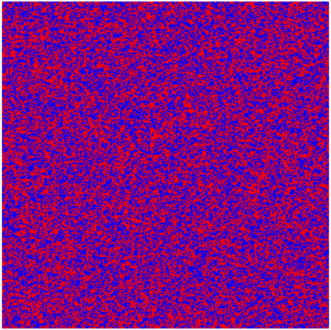
5.3 Examples of CA rules

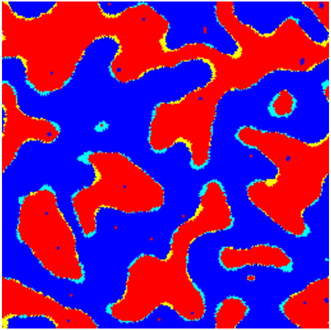
Cooperation models : annealing rule

- Growth model in physics : droplet, interface, etc
- Biased majority rule : (almost copy what the neighbors do)

Rule :

$\text{sum}_{ij}(t)$	0	1	2	3	4	5	6	7	8	9
$s_{ij}(t+1)$	0	0	0	0	1	0	1	1	1	1





The rule sees the curvature radius of domains

Cells differentiation in drosophila

In the embryo all the cells are identical. Then during evolution they differentiate

- slightly less than 25% become neural cells (neuroblasts)
- the rest becomes body cells (epidermioblasts).

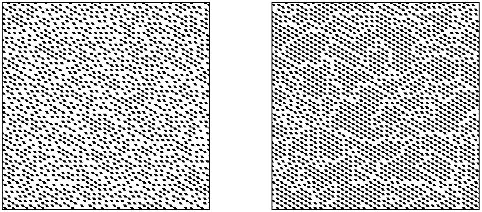
Biological mechanisms :

- Cells produce a substance S (protein) which leads to differentiation when a threshold S_0 is reached.
- Neighboring cells inhibit the local S production.

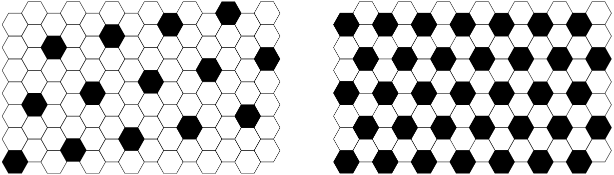
CA model for a competition/inhibition process

- Hexagonal lattice
- The values of S can be 0 (inhibited) or 1 (active) in each lattice cell.
- A $S = 0$ cell will grow (i.e. turn to $S = 1$) with probability p_{grow} provided that all its neighbors are 0. Otherwise, it stays inhibited.
- A cell in state $S = 1$ will decay (i.e. turn to $S = 0$) with probability p_{anihil} if it is surrounded by at least one active cell. If the active cell is isolated (all the neighbors are in state 0) it remains in state 1.

Differentiation : results



(a) (b)



The two limit solutions with density 1/3 and 1/7, respectively.

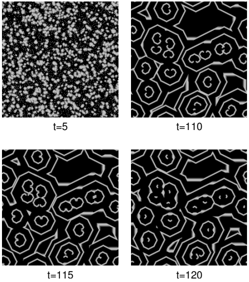
- CA produces situations with about **23%** of active cells, for almost any value of p_{anihil} and p_{growth}.
- Model robust to the lack of details, but need for hexagonal cells

Excitable Media, contagion models

- 3 states : (1) normal (resting), (2) excited (contagious), (3) refractory (immuned)
    1. excited $\rightarrow$ refractory
    2. refractory$\rightarrow$ normal
    3. normal $\rightarrow$ excited, if there exists excited neighbors (otherwise, normal $\rightarrow$ normal).

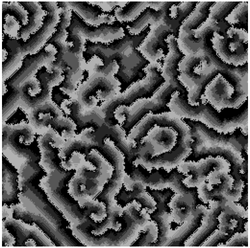
Greenberg-Hastings Model

- $s \in \{0, 1, 2, ..., n-1\}$
- normal : $s = 0$; excited $s = 1, 2, ..., n/2$; the remaining states are refractory
- contamination if at least k contaminated neighbors.



Belousov-Zhabotinski (tube worm)

The state of each site is either 0 or 1 ; a local timer with values 0, 1, 2 or 3 controls the 0 period.



(i) where the timer is zero, the state is excited ;

(ii) the timer is reset to 3 for the excited sites which have two, or more than four, excited sites in their Moore neighborhood.

(iii) the timer is decreased by 1 unless it is 0 ;

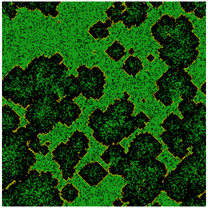
Forest fire

(1) a burning tree becomes an empty site ;

(2) a green tree becomes a burning tree if at least one of its nearest neighbors is burning ;

(3) at an empty site, a tree grows with probability p ;

(4) A tree without a burning nearest neighbor becomes a burning tree during one time step with probability f (lightning).



Other rules :

Random

$$C(t+1) = [S(t) \text{ and } E(t)] \text{ xor } W(t) \text{ xor } N(t) \text{ xor } C(t)$$

time-tunnel

$$\mathrm{Sum}(t) = S(t) + E(t) + W(t) + N(t) + C(t)$$

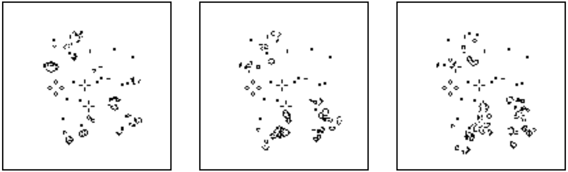
$$C(t+1) = \begin{cases} C(t-1) & \text{if } \mathrm{Sum}(t) \in \{0,5\} \\ 1 - C(t-1) & \text{if } \mathrm{Sum}(t) \in \{1,2,3,4\} \end{cases}$$

http://cui.unige.ch/~chopard/CA/Animations/img-root.html

5.4 Complex systems

Rule of the Game of Life :

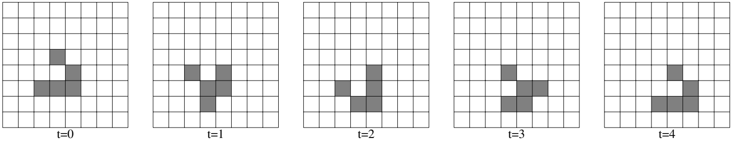
- Square lattice, 8 neighbors
- Cells are dead or alive (0/1)
- Birth if exactly 3 living neighbors
- Death if less than 2 or more than 3 neighbors



t t+10 t+20

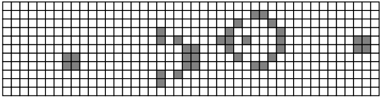
Complex Behavior in the game of life

Collective behaviors develop (beyond the local rule)
“Gliders” (organized structures of cell) can emerge and can **move** collectively.

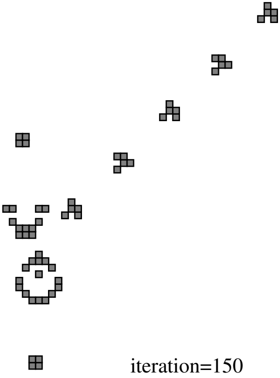


Complex Behavior in the game of life

A glider gun



- A *glider gun* is a structure that keeps creating gliders
- There are more complex structures with more complex behavior : a zoology of organisms.
- The game of life is a *Universal computer*



Animation step by step

Universal computer

Being a universal computer, the game of life can be used to program any calculable function. There are many examples on Internet, including the calculation of prime numbers.
The following link gives another example :

The following link gives another example :

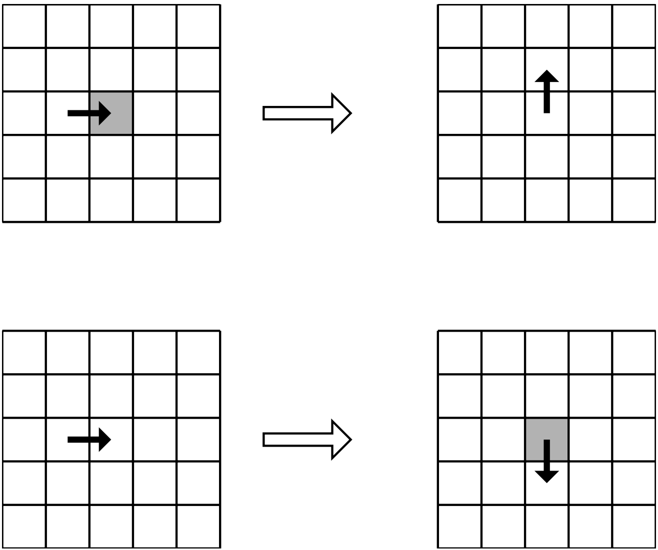
A clock made out of the game of life rules

https://codegolf.stackexchange.com/questions/88783/build-a-digital-clock-in-conways-game-of-life

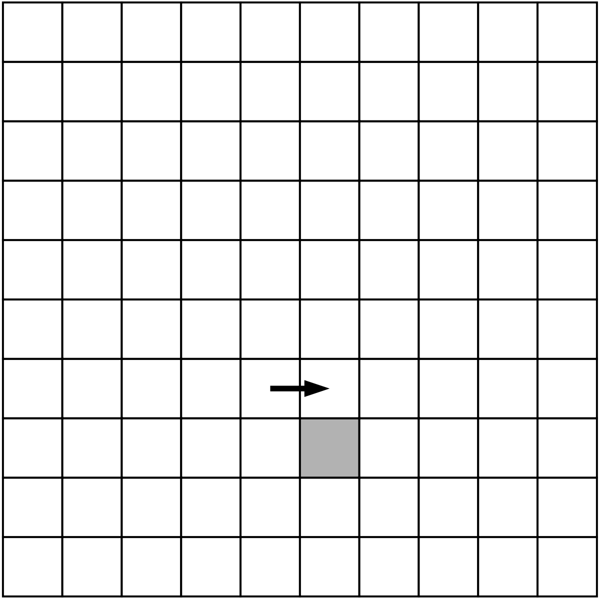
La fourmi de Langton

C'est un animal hypothétique qui se déplace sur un réseau en 2D, en fonction de règles simples, qui dépendent de la «couleur» des cases sur laquelle elle se trouve.

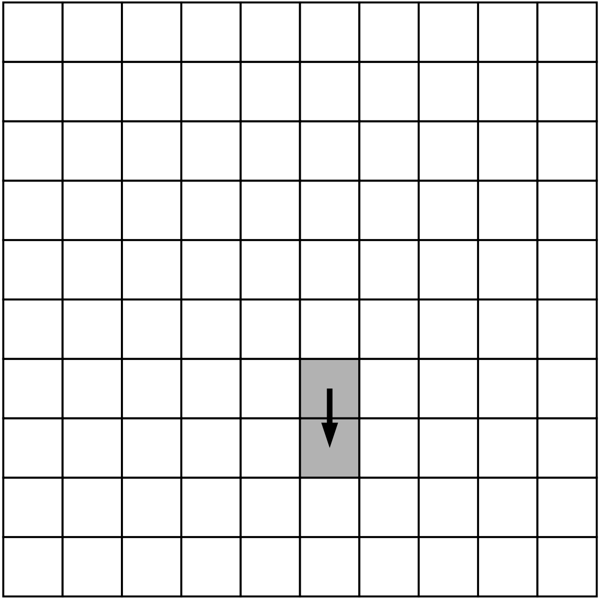
La fourmi de Langton : quelques étapes



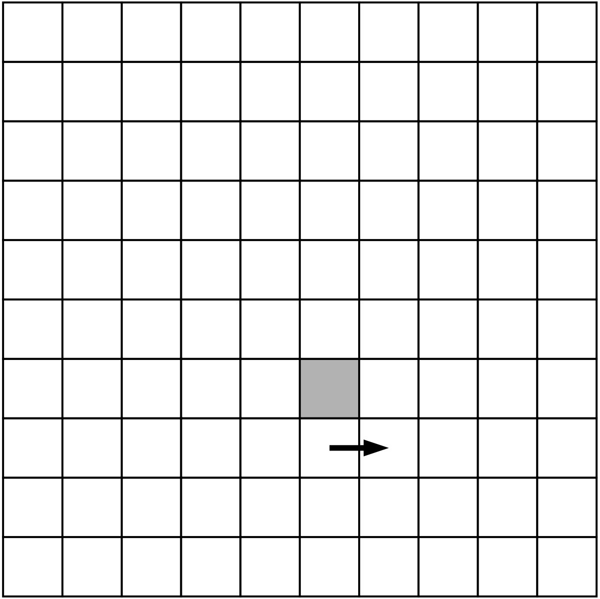
La fourmi de Langton : quelques étapes



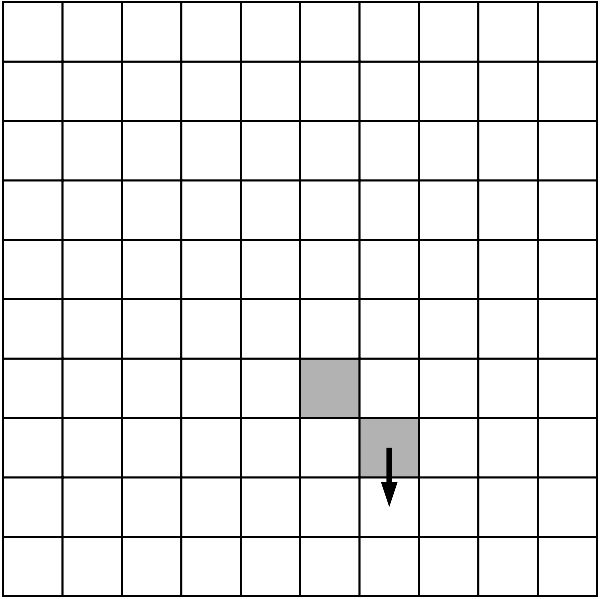
La fourmi de Langton : quelques étapes



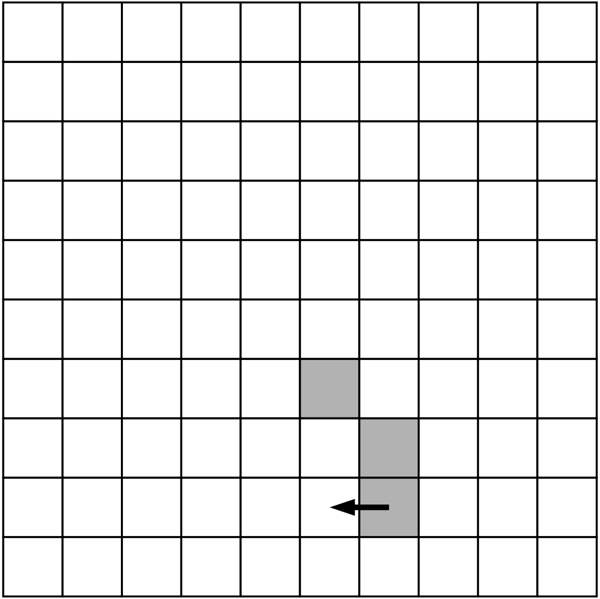
La fourmi de Langton : quelques étapes



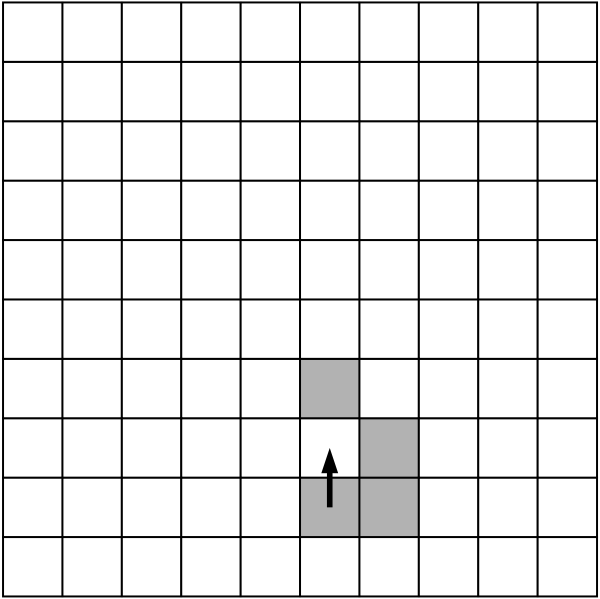
La fourmi de Langton : quelques étapes



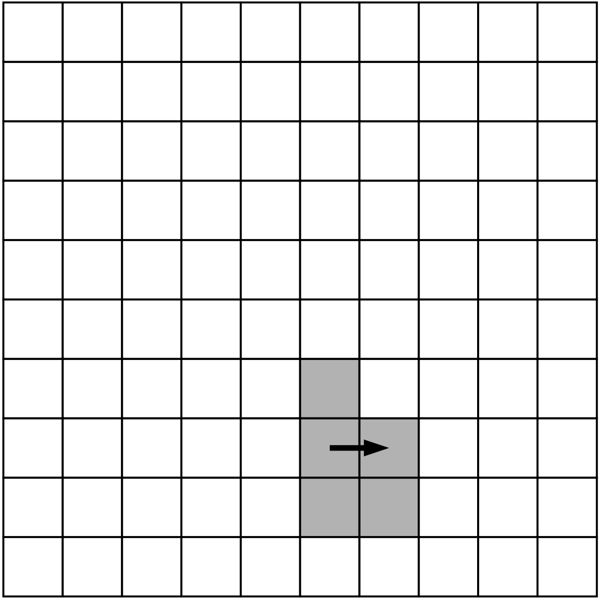
La fourmi de Langton : quelques étapes



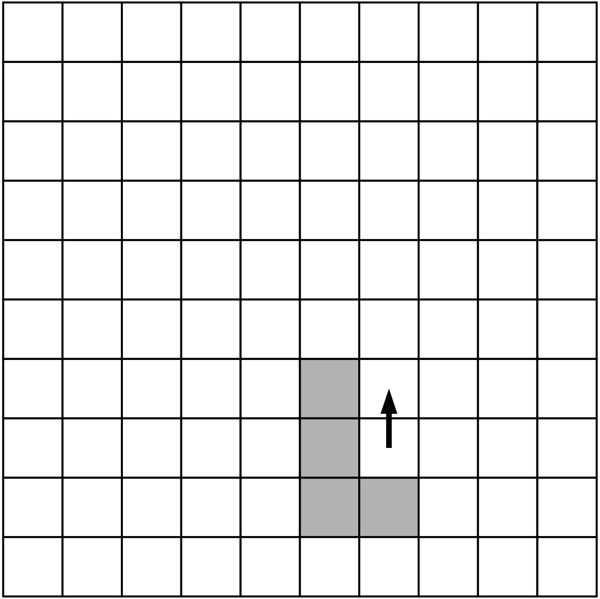
La fourmi de Langton : quelques étapes



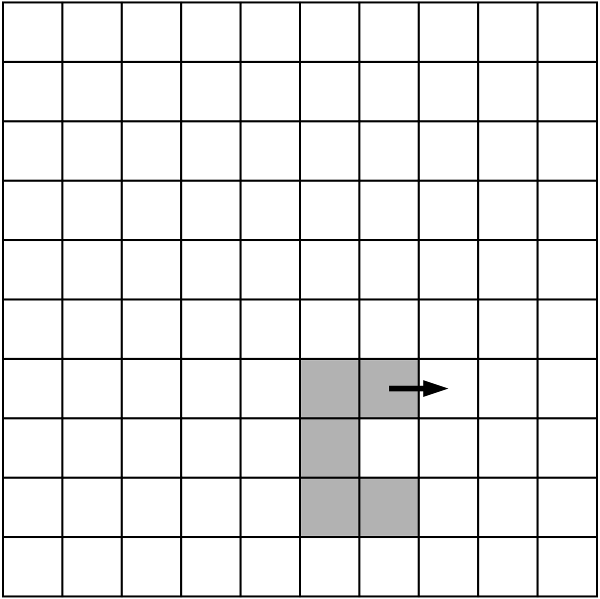
La fourmi de Langton : quelques étapes



La fourmi de Langton : quelques étapes



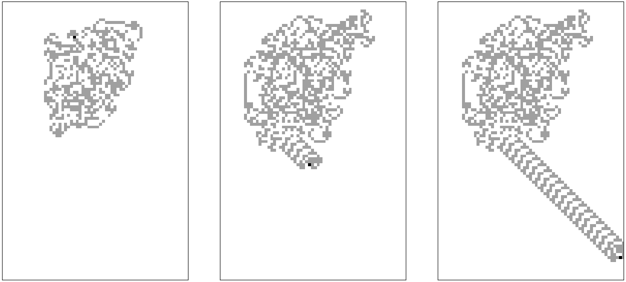
La fourmi de Langton : quelques étapes



Où va la fourmi, finalement ?

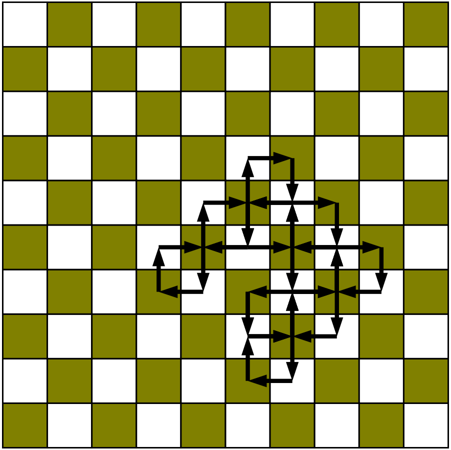
- Animation `C++/Ant-Langton ./ant 100 100 11600`

Où va la fourmi, finalement ?



t=6900 t=10431 t=12000

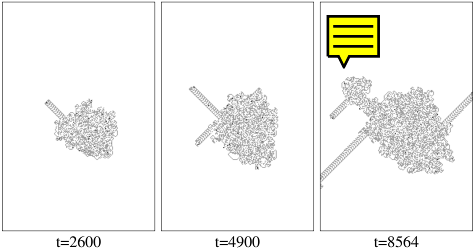
La fourmi s'échappe toujours à l'infini



Quelque soit la distribution des cases blanches et grises.

What about many ants?

- Adapt the "change of color" rule
- Cooperative and destructive effects
- The trajectory can be bounded or not
- Past/futur symmetry explains periodic motion



Impact sur la démarche scientifique

- On connait parfaitement la loi fondamentale du système

Impact sur la démarche scientifique

- On connait parfaitement la loi fondamentale du système
- ...car on l'a construit nous même !

Impact sur la démarche scientifique

- On connait parfaitement la loi fondamentale du système
- ...car on l'a construit nous même !
- Mais on ne sait pas pour autant prédire les détails du mouvement (p. ex. à quel moment une autoroute apparaît)

Impact sur la démarche scientifique

- On connait parfaitement la loi fondamentale du système
- ...car on l'a construit nous même !
- Mais on ne sait pas pour autant prédire les détails du mouvement (p. ex. à quel moment une autoroute apparaît)
- Le microscopic n'est pas toujours capable de prédire le macroscopique.

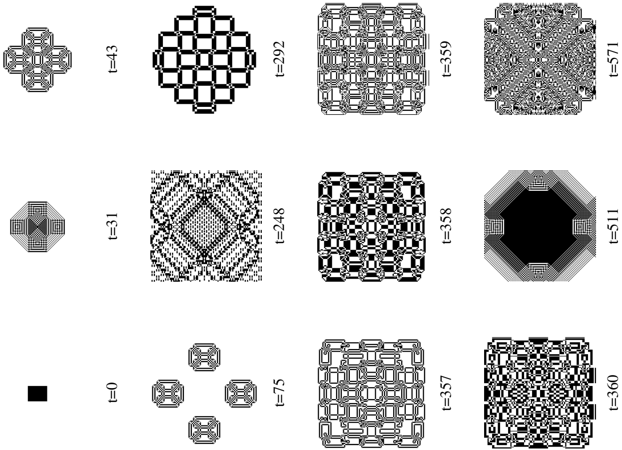
Impact sur la démarche scientifique

- On connait parfaitement la loi fondamentale du système
- ...car on l'a construit nous même !
- Mais on ne sait pas pour autant prédire les détails du mouvement (p. ex. à quel moment une autoroute apparaît)
- Le microscopic n'est pas toujours capable de prédire le macroscopique.
- Seule solution : **observer** le système.

Impact sur la démarche scientifique

- On connait parfaitement la loi fondamentale du système
- ...car on l'a construit nous même !
- Mais on ne sait pas pour autant prédire les détails du mouvement (p. ex. à quel moment une autoroute apparaît)
- Le microscopic n'est pas toujours capable de prédire le macroscopique.
- Seule solution : **observer** le système.
- Les seules informations sur la trajectoire globale sont le reflet des symétries de la règle

Parfois on peut être plus efficace que simplement observer



Parfois on peut être plus efficace que simplement observer

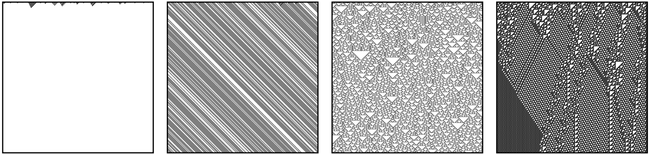
(a) (b) (c)

After 2^k iterations, the pattern is the superposition of the initial pattern shifted by k cells in each of the four directions. Therefore one can compute the pattern at iteration t in $\mathcal{O}(n^2 \log t)$ operations instead of $\mathcal{O}(n^2 t)$ for the simulation.

Wolfram's rules

256 one-dimensional, 3 neighbors Cellular Automata :

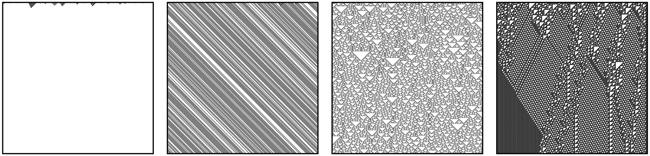
$$\underbrace{111}_{1} \quad \underbrace{110}_{1} \quad \underbrace{101}_{1} \quad \underbrace{100}_{1} \quad \underbrace{011}_{1} \quad \underbrace{010}_{0} \quad \underbrace{001}_{0} \quad \underbrace{000}_{0}$$



(a) (b) (c) (d)

Coombs, Stephen 2009, The Geometry and Pigmentation of Seashells

Wolfram's rules : complexity classes

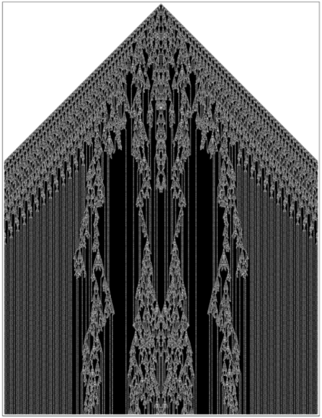


(a) (b) (c) (d)

- **Class I** Reaches a fixed point
- **Class II** Reaches a limit cycle
- **Class III** self-similar, chaotic attractor
- **Class IV** unpredicable persistent structures, irreducible, universal computer

Note : it is **undecidable** whether a rule belongs or not to a given class.

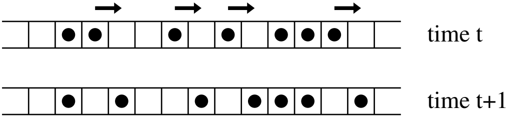
Wolfram's rules : 1D, 5 neighbors

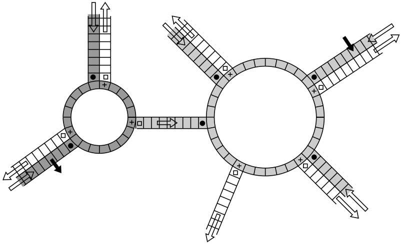


code 1635

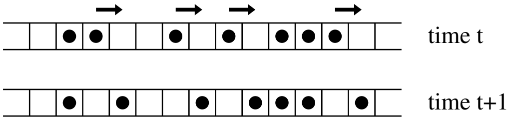
5.5 Traffic Models

A vehicle can move only when the downstream cell is free.





Evolution rule (rule 184)



$$\underbrace{111}_{1} \quad \underbrace{110}_{0} \quad \underbrace{101}_{1} \quad \underbrace{100}_{1} \quad \underbrace{011}_{1} \quad \underbrace{010}_{0} \quad \underbrace{001}_{0} \quad \underbrace{000}_{0}$$

$$n_i(t+1) = n_i(t) \times n_{i+1} + (1 - n_i(t)) \times n_{i-1}(t)$$

There are also multi-speed models.

Physical quantities : car density

The car density at time t on a road segment of length L is defined as

$$\rho(t) = \frac{N(t)}{L}$$

where N is the no of cars along L

Physical quantities : velocity and flow

The average velocity $< v >$ at time t on this segment is defined as

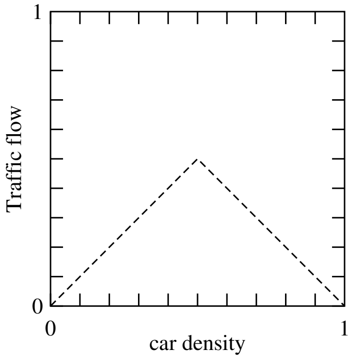
$$< v >= \frac{M(t)}{N(t)} = \frac{1}{N} \sum_{i=1}^{N} v_i$$

where $M(t)$ is the number of car moving at time t

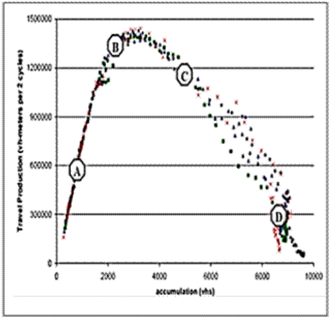
The traffic flow j is defined as

$$j = \rho < v >$$

Fundamental Flow diagram



rule 184



Macroscopic Fundamental Diagram for San-Francisco

N. Geroliminis, and C. Daganzo,"Existence of urban - scale macroscopic fundamental diagrams : Some experimental findings," Transportation Research Part B42, issue 9, (2008). pp 759 - 770.

Fundamental Flow diagram : another case

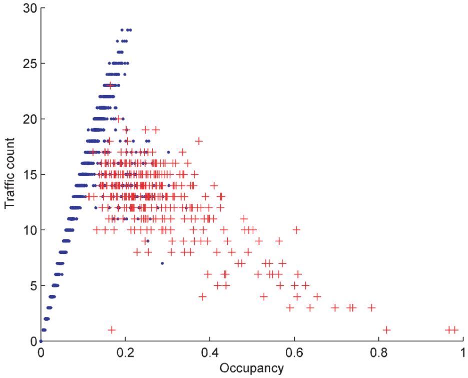


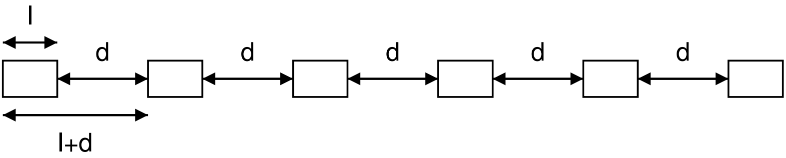
FIGURE 5 Split of observations. Dots indicate free flow; plus signs indicate congested flow.

J. Li, 1001 Ghausi Hall, and H. M. Zhang (2001). Transportation Research Record : Journal of the Transportation Research Board,No. 2260,Transportation Research Board of the National Academies,

DOI : 10.3141/2260-0

A more general numerical model

Let us assume cars of lenght ℓ, each separated by a distance d (head-tail distance).



The density ρ is defined as the number of cars per meter occupied on the road, namely

$$\rho = \frac{1}{\ell + d} \quad \left[\frac{1}{m}\right]$$

The maximum density is $\rho_{max} = 1/\ell$, when $d = 0$.

Car speed depends of d

Let is further assume that the car velocity is proportional to d

$$v = \gamma d$$

($\gamma = 0.5$ to have 2 seconds between cars, with v and d expressed in m/s and m, respectively).
We assume that there is a maximum velocity v_{max} (speed limit for instance) and that cars try to drive as fast as possible, only limited by v_{max} and d.
At speed v_{max} the distance d should be larger than d_{max} defined as

$$d_{max} = \frac{v_{max}}{\gamma}$$

Free flow : $\rho \leq \rho_c$

If d is large, $d \geq d_{max} = v_{max}/\gamma$, cars will drive at maximum speed. This corresponds to a car density

$$\rho = \frac{1}{\ell + d} \leq \rho_c \equiv \frac{1}{\ell + d_{max}} = \frac{1}{\ell + \frac{v_{max}}{\gamma}}$$

where ρ_c is the critical density

Thus, for $\rho \leq \rho_c$ cars can drive at speed v_{max}, and the flow j is

$$j = \rho v = \rho v_{max}$$

and

$$j_{max} = \rho_c v_{max} = \frac{v_{max}}{\ell + \frac{v_{max}}{\gamma}} \quad \to \quad \gamma$$

Congested regime : $\rho \geq \rho_c$

For car densities $\rho \geq \rho_c$ the velocity v is limited by the distance d which is obtained from

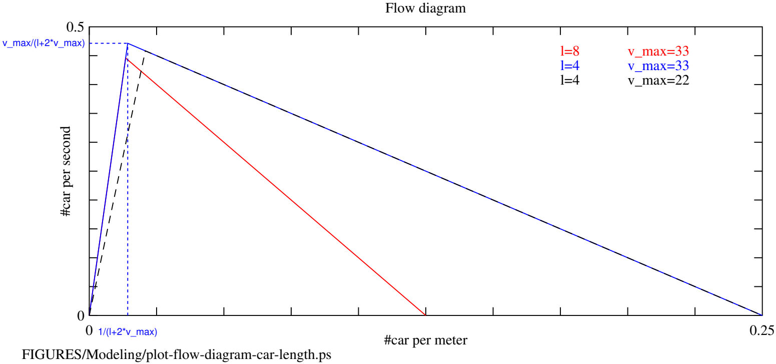
$$\rho = \frac{1}{\ell + d} \qquad \Rightarrow \qquad d = \frac{1 - \ell\rho}{\rho}$$

Then the flow is

$$j = \rho v = \rho\gamma d = \rho\gamma\frac{1 - \ell\rho}{\rho} = \gamma(1 - \ell\rho)$$

Fundamental flow diagram

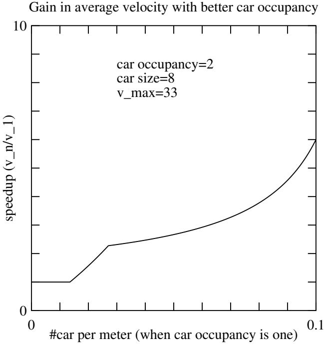
Let us now take $v_{max} = 33m/s = 120km/h$, $\gamma = 0.5$ and consider cars of lenght $\ell = 8m$ and $\ell = 4m$.



Increase the number of passengers per car

Instead of 1 passenger per car, let us consider the case of car pooling.
With n passengers per car, one reduces ρ by a factor n.
The velocity for single occupation is

$$v_1(\rho) = \begin{cases} v_{max} & \text{if } \rho \leq \rho_c = \frac{1}{\ell + v_{max}/\gamma} \\ \gamma \frac{1-\ell\rho}{\rho} & \text{otherwise} \end{cases}$$



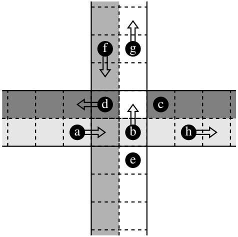
FIGURES/Modeling/plot-flow-versus-car-occupation.ps

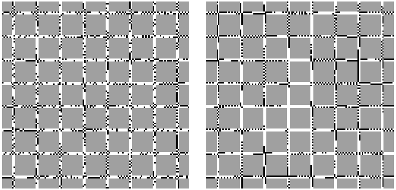
With

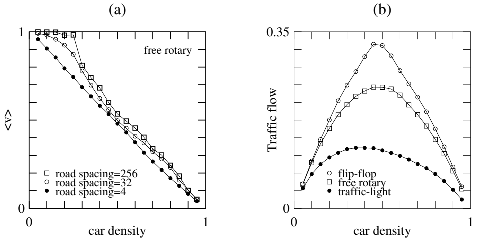
$$v_n(\rho) = v_1(\rho/n)$$

the speedup is defined as $v_n(\rho)/v_1(\rho)$ and it indcates the reduction of the travel time.

Traffic in a Manhattan-like city



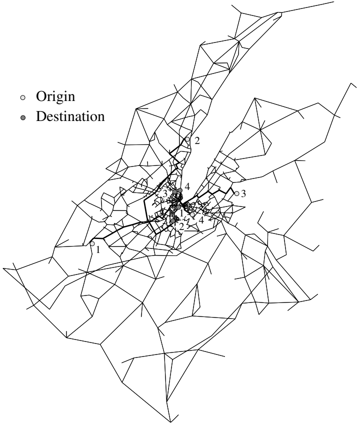

(a) (b)



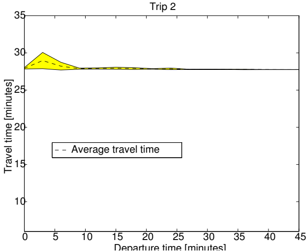
http://cui.unige.ch/~chopard/CA/Animations/img-root.html

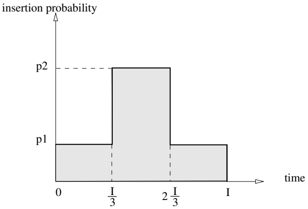
Case of the city of Geneva

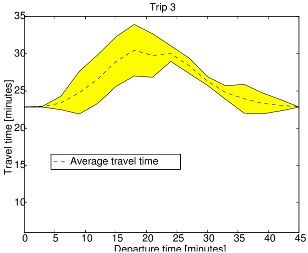
- 1066 junctions
- 3145 road segments
- 560886 road cells
- 85055 cars



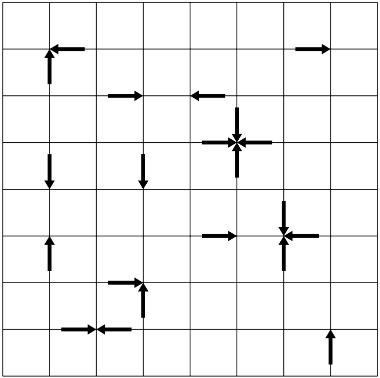
Travel time during the rush hour





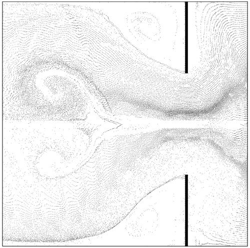


5.6 Lattice gases – Gaz sur Réseau



Gaz sur réseau

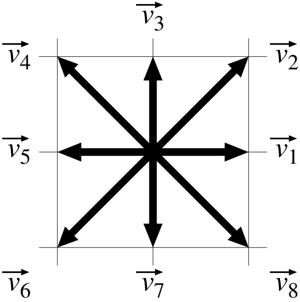
- C'est un AC qui modélise un fluide (ou un gaz) par la dynamique de particules discretes se déplaçant sur un réseaux
- LGA : Lattice Gas automata
- Dynamique moléculaire entièrement discrètes
- Particules idéalisée, à une échelle **mésoscopique** : les détails microscopiques sont simplifiés, mais effet collectif d'un fluide



Gaz sur réseau

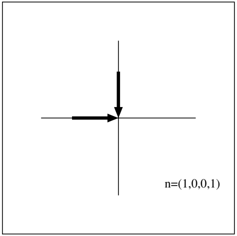
- On peut montrer rigoureusement l'équivalence de ce type de modèle avec le comportement d'un vrai fluide.
- On parle de CA-fluid ou LGA-fluid
- On peut aussi représenter des processus de diffusion, réaction chimique, advection, etc.
- Les modèles de Boltzmann sur réseau vont généraliser cette méthode

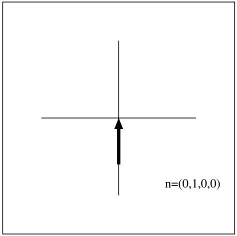
Description



- les particules ont un nombre fini de vitesses possibles, v_i
- Elle sont telles qu'en un pas de temps Δt de l'AC, elles sautent sur un autre site du réseau, parcourant typiquement une distance Δx.
- le choix des v_i est fortement lié au choix du réseau sous-jacent puisque $\mathrm{r} + \Delta t \mathrm{v}_i$ doit être un site du réseau

Description





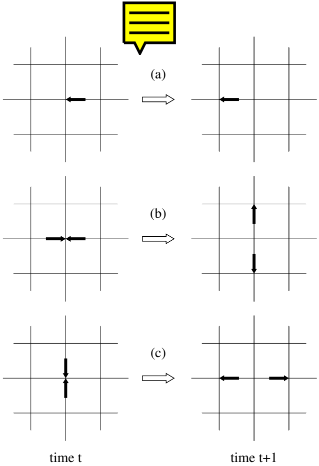
- L'état de chaque site (cellule) r est décrit par des «nombres d'occupation» $n_i(\mathrm{r}, t)$
- $n_i(\mathrm{r}, t) = 1$ signifie qu'une particule *entre* dans le site r au temps t avec vitesse v_i.
- $n_i = 0$ signifie l'absence d'une particule entrante par ce canal
- $n = (n_1, n_2, n_3, n_4)$

Principe d'exclusion

- $n_i \in \{0, 1\}$ est un booléen : au max 1 particule par site et par direction.
- Un nombre fini de bit est donc suffisant pour décrire l'état du système

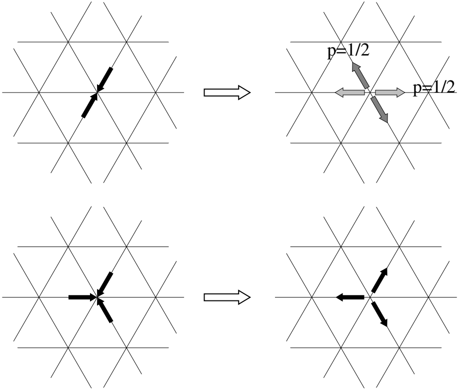
Example : HPP model collision rules

- HPP : Hardy, Pomeau, de Pazzis, 1971 : kinetic theory of point particles on the D2Q4 lattice
- FHP : Frisch, Hasslacher and Pomeau, 1986 : first LGA reproducing a (almost) correct hydrodynamic behavior (Navier-Stokes eq.)



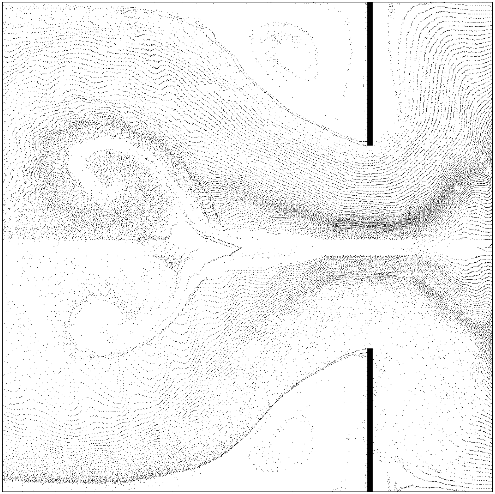
Exact mass and momentum conservation : that is what really matters for a fluid ! ! !

FHP model



Stochastic rule with Conservation of mass and momentum.

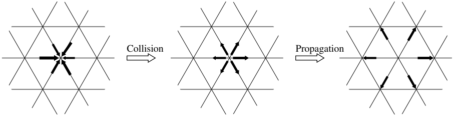
Flow past an obstacle (FHP)



Microdynamics of LGA

It consists of two steps. We define $n_i^{in} = n_i$ and n_i^{out} to better specify them

- **Collision step :** The quantities n_i^{in} "collide" locally. Particles are deviated and new values n_i^{out} are computed at each lattice site, according to a pre-defined collision operator $\Omega_i(n)$
- **Propagation step :** The quantitiy $n_i^{out}(\mathrm{r})$ is sent to the neighboring site along lattice direction v_i.



Microdynamics of LGA

In formula, we get

- collision : $n_i^{out}(\mathrm{r}, t) = n_i^{in}(\mathrm{r}, t) + \Omega_i(n^{in}(\mathrm{r}, t)$
- propagation : $n_i^{in}(\mathrm{r} + \mathrm{v}_i \Delta t, t + \Delta t) = n_i^{out}(\mathrm{r}, t)$

where Δt carries the time units and v_i has the unit of a velocity.

Particles traveling in direction v_i will reach lattice site $\mathrm{r} + \mathrm{v}_i$, still with the same velocity v_i.

Microdynamics of LGA

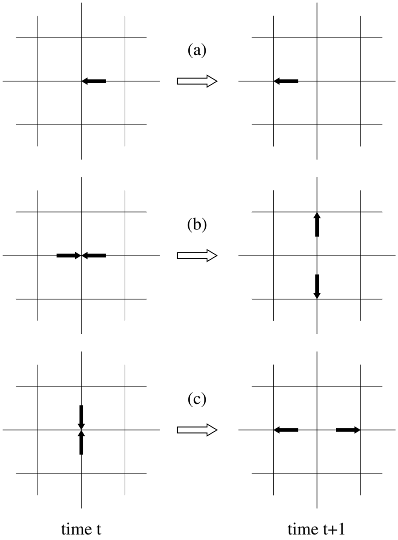
The above formula reflects how the LGA microdynamics is implemented in a computer. But mathematically, one can combine the collision and propagation steps into :

$$n_i(\mathrm{r} + \mathrm{v}_i \Delta t, t + \Delta t) = n_i(\mathrm{r}, t) + \Omega_i(n(\mathrm{r}, t)$$

where $n_i \equiv n_i^{in}$

Note : if $\Omega_i = 0$, we obtain a free particle motion

HPP model



HPP model : Computer Implementation

Admitted velocities

$$\mathrm{v}_1 = (1,0), \quad \mathrm{v}_2 = (0,1), \quad \mathrm{v}_3 = (-1,0) \quad \mathrm{v}_4 = (0,-1)$$

Microdynamics :

$$n_i^{out} \quad = \quad n_i$$

HPP model : Computer Implementation

Admitted velocities

$$\mathrm{v}_1 = (1,0), \quad \mathrm{v}_2 = (0,1), \quad \mathrm{v}_3 = (-1,0) \quad \mathrm{v}_4 = (0,-1)$$

Microdynamics :

$$\begin{aligned} n_i^{out} &= n_i \\ &\quad -n_i n_{i+2}(1-n_{i+1})(1-n_{i+3}) \end{aligned}$$

HPP model : Computer Implementation

Admitted velocities

$$\mathrm{v}_1 = (1,0), \quad \mathrm{v}_2 = (0,1), \quad \mathrm{v}_3 = (-1,0) \quad \mathrm{v}_4 = (0,-1)$$

Microdynamics :

$$\begin{aligned} n_i^{out} &= n_i \\ &\quad -n_i n_{i+2}(1-n_{i+1})(1-n_{i+3}) \\ &\quad +n_{i+1} n_{i+3}(1-n_i)(1-n_{i+2}) \end{aligned}$$

HPP model : Computer Implementation

Admitted velocities

$$\mathrm{v}_1 = (1,0), \quad \mathrm{v}_2 = (0,1), \quad \mathrm{v}_3 = (-1,0) \quad \mathrm{v}_4 = (0,-1)$$

Microdynamics :

$$\begin{aligned} n_i^{out} &= n_i \\ &\quad - n_i n_{i+2}(1-n_{i+1})(1-n_{i+3}) \\ &\quad + n_{i+1} n_{i+3}(1-n_i)(1-n_{i+2}) \end{aligned}$$

and

$$n_i(\mathrm{r}) = n_i^{out}(\mathrm{r} - \mathrm{v}_i)$$

Mass and momentum conservation

The incoming mass is

$$\rho^{in}(\mathrm{r}, t) = \sum_i n_i^{in}(\mathrm{r}, t)$$

the outgoing mass is

$$\rho^{out}(\mathrm{r}, t) = \sum_i n_i^{out}(\mathrm{r}, t)$$

It is easy to check that the HPP collision rule is such that

$$\rho^{in}(\mathrm{r}, t) = \rho^{out}(\mathrm{r}, t)$$

Similarly, **momentum** is the defined as

$$j(\mathrm{r}, t) \equiv \rho(\mathrm{r}, t)\mathrm{u}(\mathrm{r}, t) = \sum_i \mathrm{v}_i n_i(\mathrm{r}, t)$$

and it is easy to show that HPP conserves it during collision

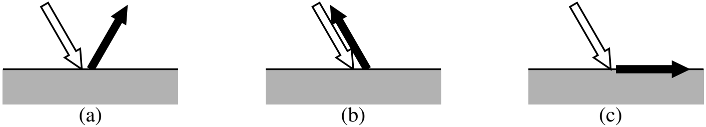
Demos

- Pressure/density wave : aniotropy
- Reversibility
- Spurious invariants : momentum along each line and column, checkerboard invariant
- Diffusion, DLA, reaction-diffusion models

Some demos

- Sound wave propagation for FHP
- Snow transport by wind

Boundary conditions



(a) Specular reflection, (b) bounce back condition and (c) trapping wall condition

The **Bounce Back rule** implements a no-slip condition. It is the most common choice :

$$n_i^{out} = n_{-i}^{in}$$