Complexité de la fourmi de Langton 

A. Gajardo, A. Moreira, E. Goles (2002)

Cédric Bérenger

AMU, Option MOCANA, 2016

La fourmi de Langton : des règles simples...

- Une fourmi évolue sur une grille 2D.
- Elle est représentée par une flêche $\uparrow\downarrow\leftarrow\rightarrow$.
- Elle se déplace en suivant la flêche.
- Lorsqu'elle se déplace :
  - la couleur de la case courante s'inverse.
  - la flèche est tournée de 90° :
    - ★ Vers la gauche si la case est blanche,
    - ★ Vers la droite si la case est noire.

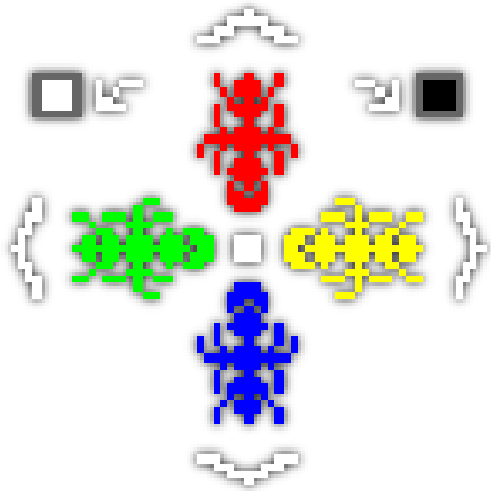

FIGURE – Orientations.

- Simulateur : https://github.com/CedricBerenger/LangtonsAnt

... Un Comportement complexe.

La fourmi présente un comportement complexe :

- Initialement Symétrique,
- Puis Chaotique,
- Et Finalement Périodique : la fourmi s'enferme dans une « Autoroute ».

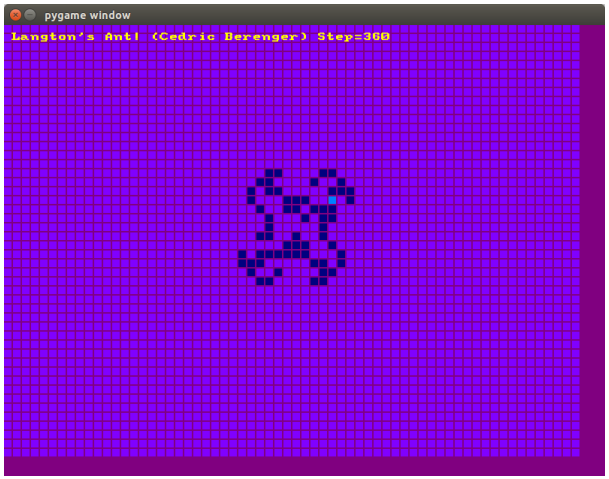

FIGURE – Symétrique (étape<500)

FIGURE – Chaotique (500< étape<10000)

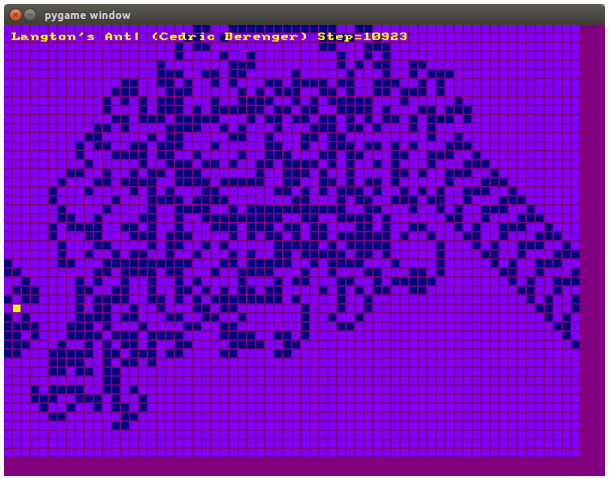

FIGURE – Périodique (étape>10000)

Folklore : Un résultat principal.

Théorème (Cohen-Kung Theorem)

La trajectoire de la fourmi de Langton est non bornée.

Preuve (Par l'absurde).

Si la trajectoire est bornée, alors le comportement de la fourmi est périodique et est localisé dans une zone fixe. Pour que le comportement soit périodique, un état doit se répéter. Or s'il on considère les coins de la zone, on aboutit à une contradiction : la zone ne peut être fixe. □

Conjecture de l'Autoroute

Si le support est fini, on conjecture que la fourmi finit toujours par construire l'autoroute.

Objectif : La Fourmi de Langton, un système P-Difficile et Universel

- Nous allons montrer que la fourmi de Langton est un système P-Complet et capable de Calcul Universel, pour cela :
- Nous allons simuler une machine de Turing (Universalité) ...
  - ... Via la simulation d'Automates Cellulaires 1D ...
  - ... Via l'assemblage de Circuits Booléen (P-Difficulté) ...
  - ... Via la construction de *Portes Logiques*.

Vers la Construction de portes logiques : Les Chemins.

- Les Chemins permettent de contrôler les déplacements de la fourmi.

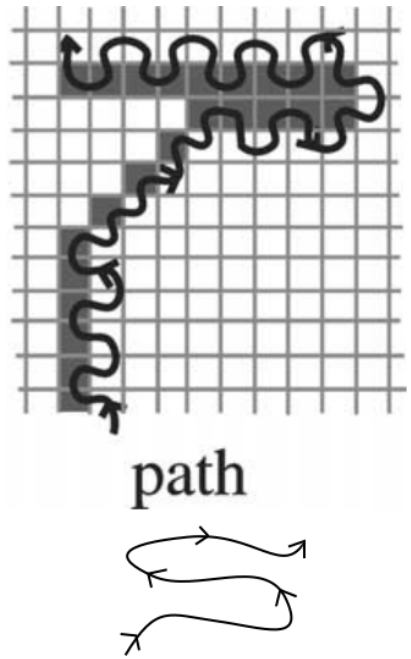

FIGURE – Exemple de chemin.

Vers la Construction de portes logiques : La graine de L'Autoroute.

- Lorsque la fourmi a terminé son calcul, on l'emprisonne dans l'autoroute pour contrôler sa trajectoire à l'infini.

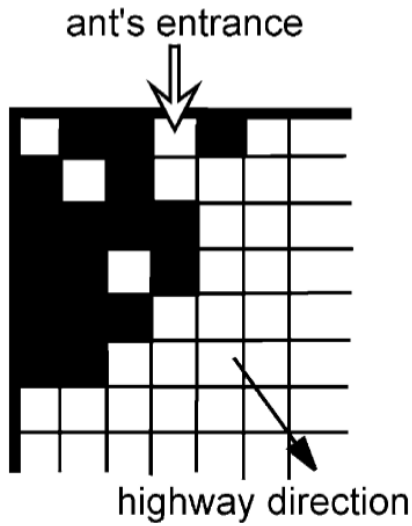

FIGURE – Graine permettant de générer l'autoroute.

Construction de portes logiques : Principe.

- Etats d'entrées/sorties codés par les états de cellules.
- La fourmi calcule en se déplaçant :
  - L'état des cellules d'entrée modifient la trajectoire de la fourmi.
  - Suivant la trajectoire adoptée, la fourmi écrit ou non sur les cellules de sortie par défaut à *False*.

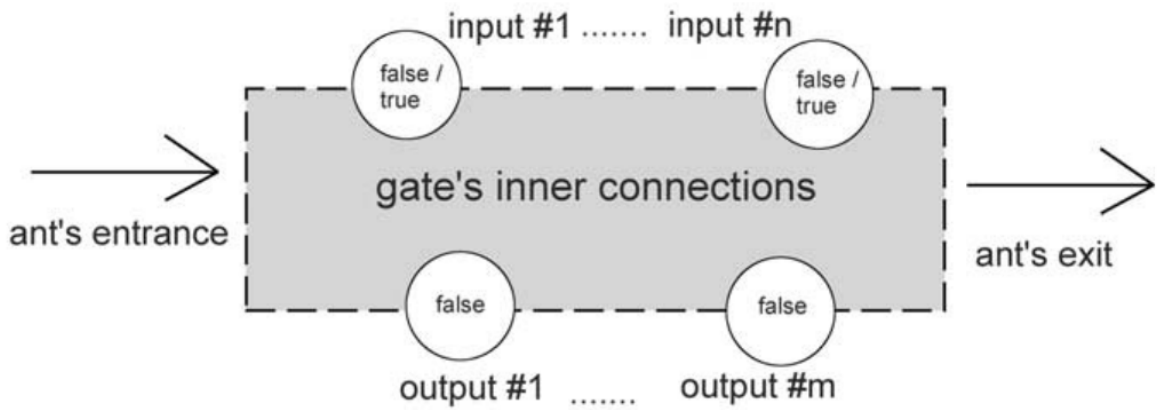

FIGURE – Schéma de fonctionnement d'une porte logique.

Construction de portes logiques : Les entrées / sorties.

- Les Entrées se comportent comme des *commutateurs*, orientant la fourmi sur des chemins différents en fonction de l'entrée.
- Les sorties sont à *False* par défaut. Si la fourmi suit le chemin d'une sortie, elle y écrit *True*.

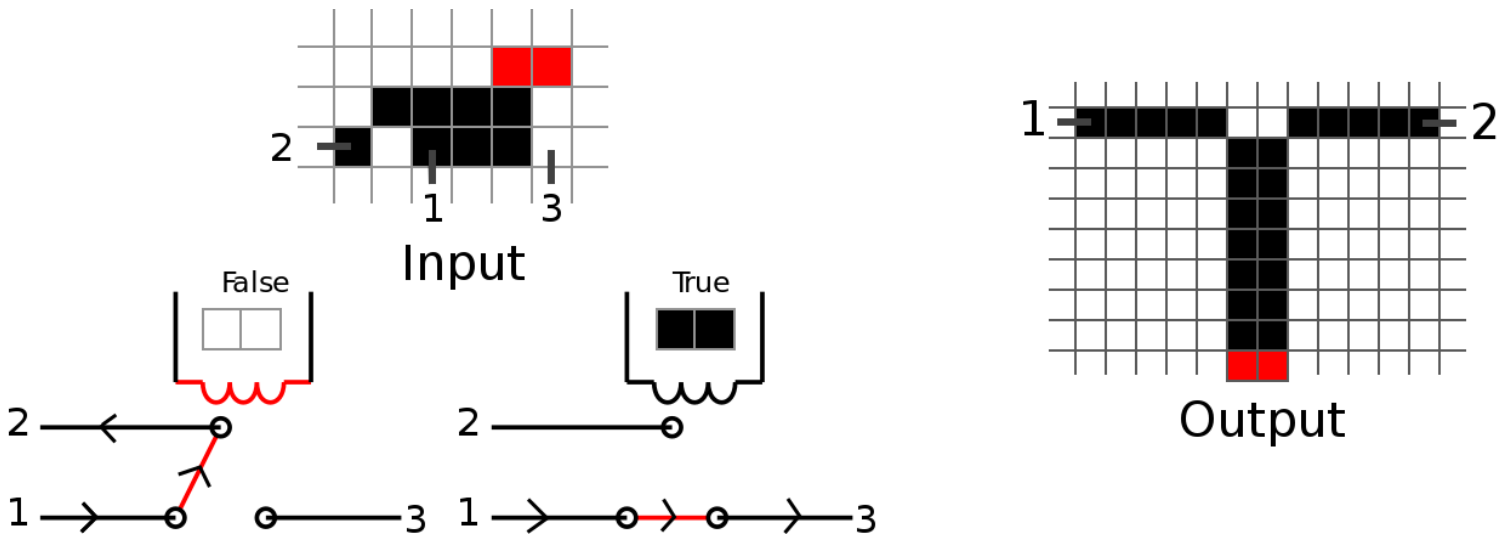

FIGURE – Schéma de fonctionnement des Entrées / Sorties.

Construction de portes logiques : Les briques de bases.

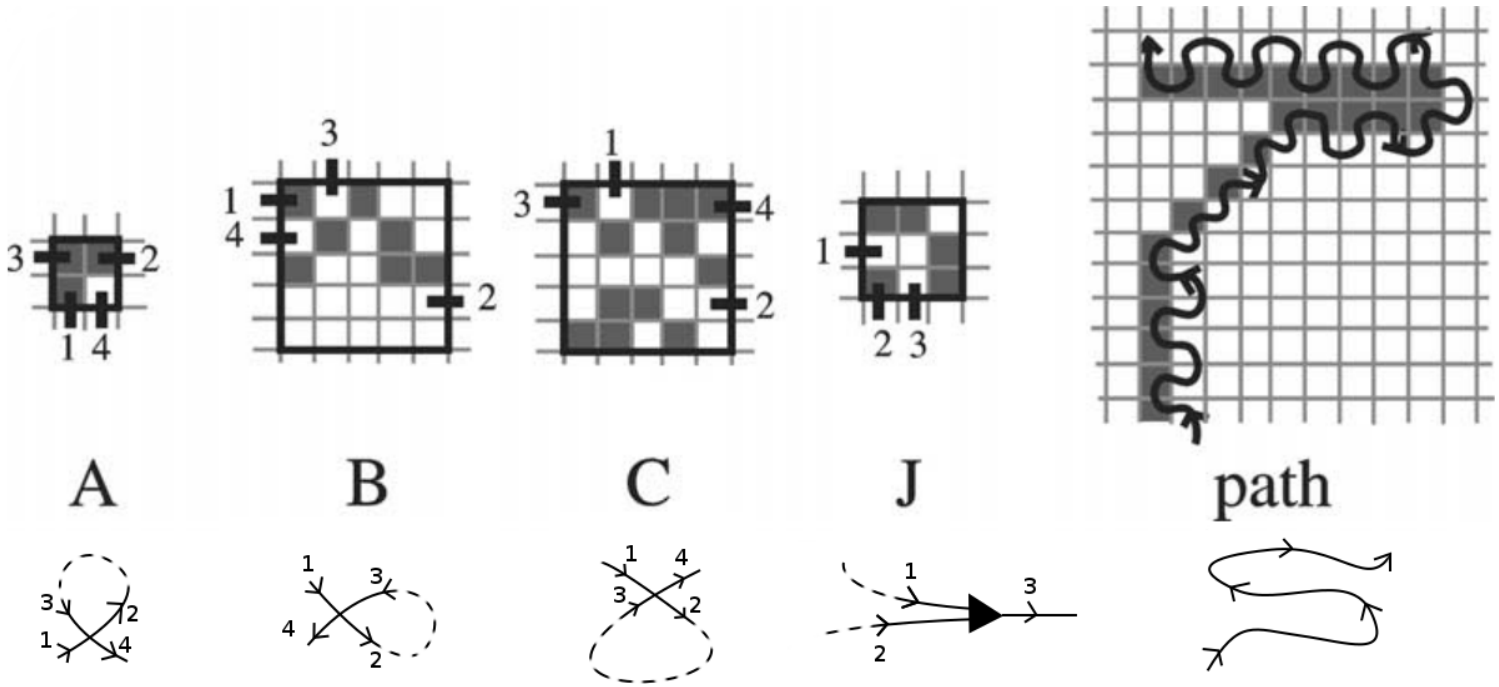

FIGURE – Les croisements A,B,C, la jonction J et les chemins sont les briques de bases pour la création de portes logiques. *Remarque :* B et C permettent l'entrée en 3 et la sortie en 4 sans passer d'abord par 1 et 2.

Construction des portes logiques : Schémas.

- Les portes sont « cablées » par des chemins.
- Les croisements et les entrées/sorties utilisent les briques de bases.

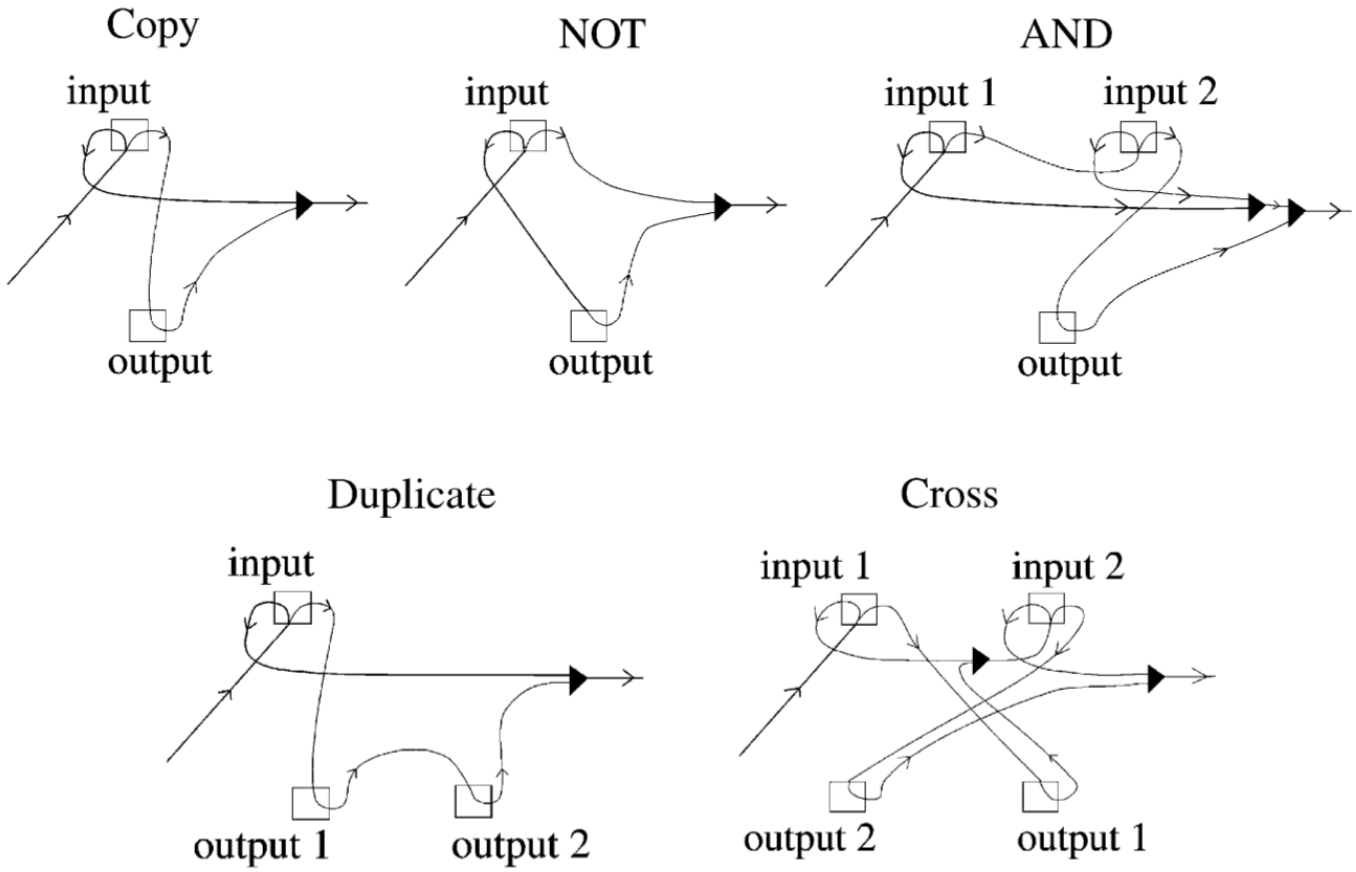

FIGURE – Représentation schématique de portes logiques.

Construction de portes logiques : AND et NOT.

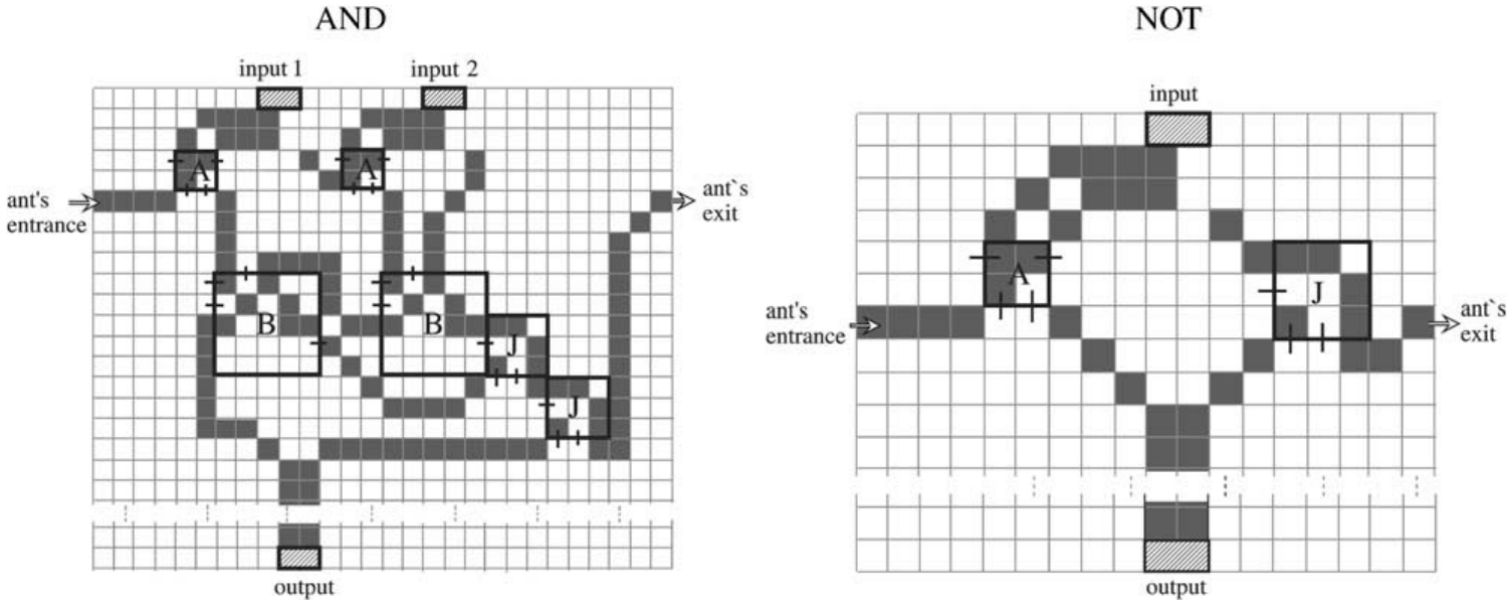

FIGURE – Construction des portes AND et NOT.

Construction de portes logiques : Cross, Copy, Duplicate.

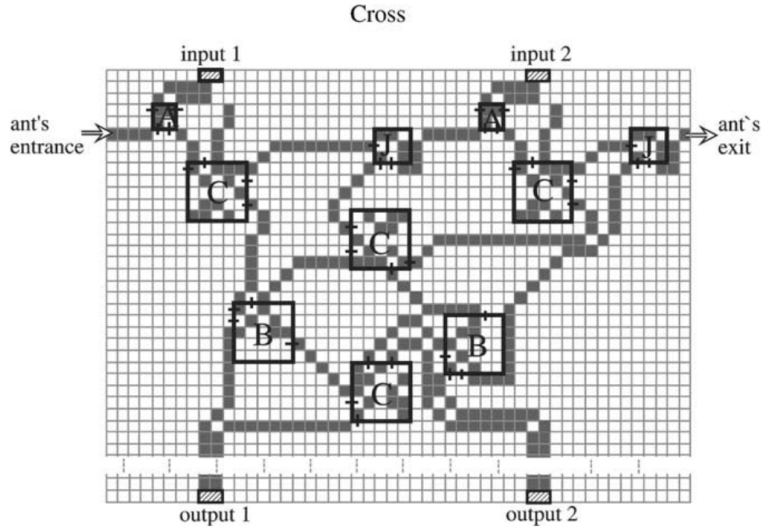

Fig. 7. Cross gate.

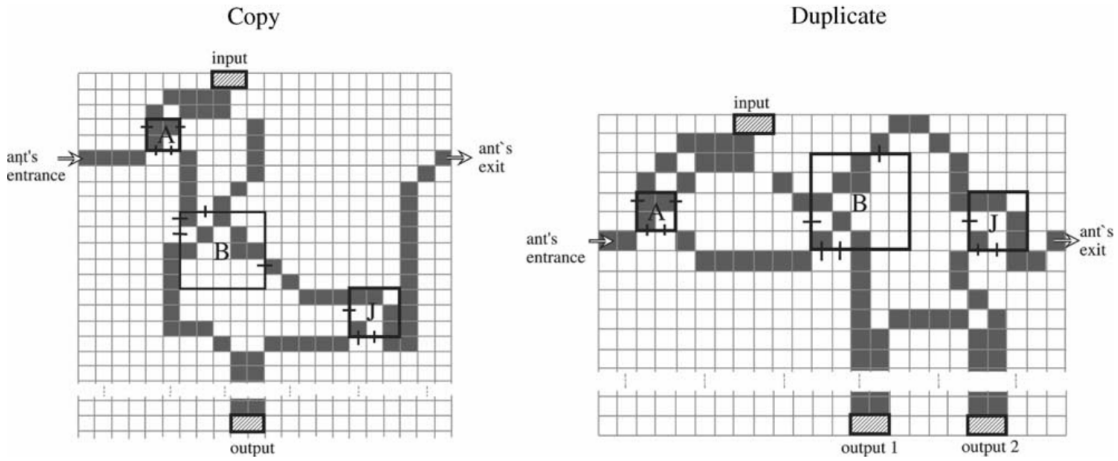

FIGURE – Construction des portes Cross, Copy (Yes) et Duplicate.

Assemblage de Circuits Booléens.

- Les portes logiques précédentes permettent de coder et de calculer nimporte quel circuit booléen fini.

Calcul d'un XOR

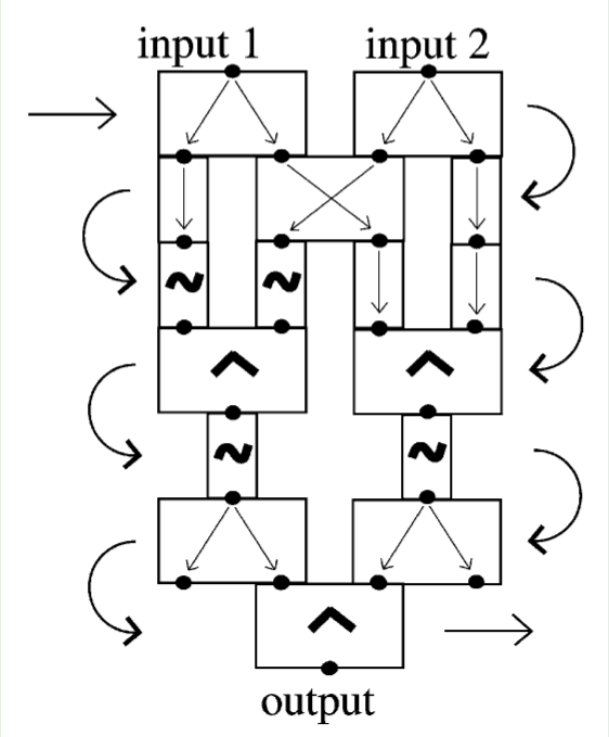

FIGURE – Schéma d'assemblage d'une porte XOR : $\neg(\neg a \wedge \neg b) \wedge \neg(a \wedge b)$.

Simulation d'automates cellulaires 1D.

- Fonction de transition calculée par un circuit booléen fini R.
- Calcul d'un état de l'automate par une cascade de circuits R.
- Chaque ligne calcul un état et augmente le champ d'action (trapèze).

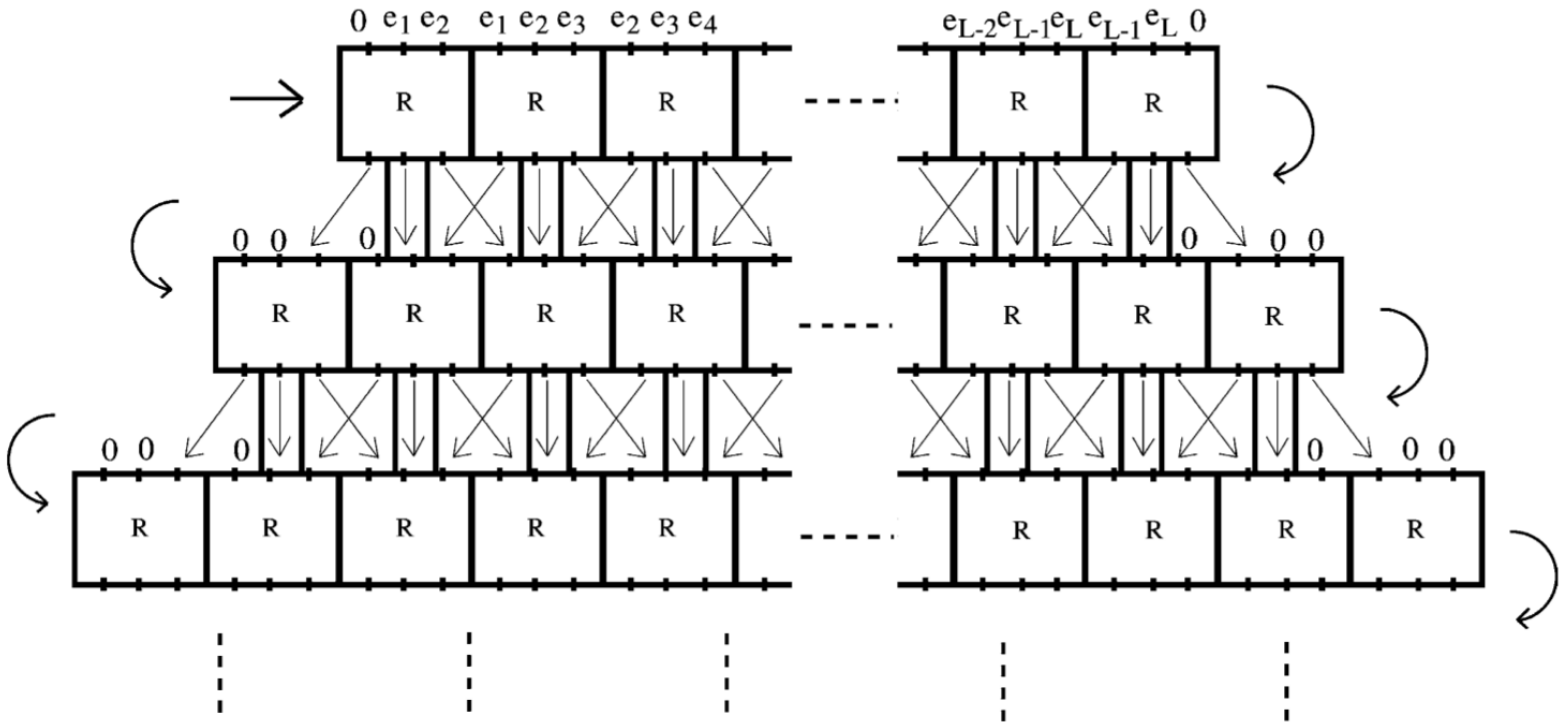

FIGURE – Simulation d'un automate cellulaire 1D.

Conclusion : La Complexité de la fourmi de Langton

- La fourmi de Langton admet des problèmes P-Difficiles :
  - Le calcul d'un circuit booléen est un problème P-Complet,
  - Ce problème se réduit polynomialement à la fourmi de Langton.
- La fourmi de Langton est un Modèle de Calcul Universel (Nécessite un support infini) :
  - Le modèle est capable de simuler des automates cellulaires 1D (à etats quiescents),
  - Il existe des automates cellulaires 1D à état quiescent capable de simuler une machine de turing universelle.