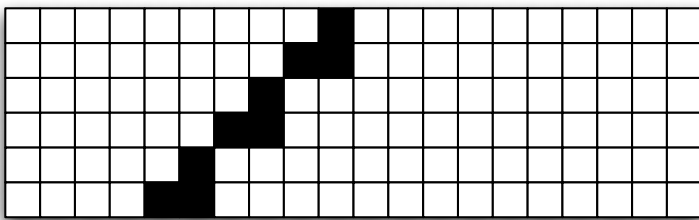


Vie Artificielle

Automates Cellulaires 1D

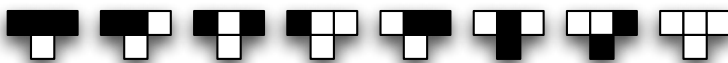
Dernière mise à jour: janvier 2015

Les automates cellulaires ont été largement popularisés dans les années 50 par John Von Neumann au cours de ses travaux pour construire un automate à 29 états dit “constructeur universel”, c’est à dire capable de se dupliquer ainsi que de dupliquer tout automate lui étant rattaché. Une version très simplifiée des automates cellulaires a été largement étudiée par Stephen Wolfram, entre autres, au début des années 90: les automates cellulaires à deux états sur une dimension. En voici un exemple typique:



La première ligne est initialisé ici par l'utilisateur, avec une seule cellule noire. La ligne suivante va ensuite être mise à jour en fonction *exclusivement* de la ligne précédente à l'aide d'une règle de mise à jour à définir, et ainsi de suite. Ainsi, l'automate se développe sur plusieurs lignes, la ligne du bas étant toujours l'état courant (i.e. on visualise en 2D la mise à jour d'un automate 1D).

Un règle de mise à jour est en réalité définie comme un ensemble de N sous-règles définissant l'état d'une nouvelle cellule selon l'état des trois cellules précédentes. Cet ensemble se présente sous la forme suivante (il s'agit de l'ensemble de sous-règles utilisé pour générer l'automate ci-dessus):



Pour chaque “T”, la forme du haut détermine tous les états possibles des cellules qui précèdent la cellule à mettre à jour. La forme du bas détermine l'état de la nouvelle cellule étant donnée les cellules qui la précède. Lors de la détermination d'un état d'une nouvelle cellule, il faut donc trouver, parmi les états possibles des cellules précédentes, celui qui correspond à la situation courante, puis initialiser la nouvelle cellule à l'état correspondant. Dans le cas particulier de cet exemple, quelque soit les états des cellules précédentes, la nouvelle cellule aura l'état 0, sauf si la cellule *soit* au dessus à droite, *soit* au dessus à gauche, est active, à l'exclusion de tout autre.

Tous ensembles de sous-règles ont un numéro unique. Ce numéro est déterminé en lisant, de droite à gauche, l'état résultant pour chaque configuration des cellules précédentes. Pour l'ensemble de sous-règles utilisé ci-dessus, on lit de droite à gauche: 00000110 (en binaire, on écrit 00000110b). Il nous reste à convertir ce numéro depuis la représentation binaire vers une représentation en décimal. Le bit de poids fort étant (toujours) à gauche, on aura:

$$\begin{aligned} &00000110b \\ &= 0 * 2^7 + 0 * 2^6 + 0 * 2^5 + 0 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 \\ &= 0 + 0 + 0 + 0 + 0 + 2^2 + 2^1 + 0 \\ &= 6d \end{aligned}$$

Il s'agit donc de la **règle 6**.