

REGARDS

LOGIQUE & CALCUL

Le royaume du *Jeu de la vie*

L'univers plan du Jeu de la vie est l'exemple même d'un système simple susceptible d'engendrer de la complexité, du calcul et... certaines formes de beauté.

Jean-Paul DELAHAYE

De même qu'il existe un monde du *Seigneur des Anneaux* créé par John Tolkien, un monde de *Harry Potter* créé par Joanne Rowling, et que toutes les grandes œuvres littéraires engendrent des doubles imaginaires de notre univers réel, il existe aussi un monde du *Jeu de la vie* de John Conway. Ce qui se passe dans ce monde-là est une aventure combinatoire et mathématique d'un genre unique. J. Conway, qui fixa les lois « physiques » de cette fiction informatique en 1970, a créé l'équivalent d'une grande page blanche sur laquelle des milliers de joueurs écrivent en proposant des constructions de plus en plus complexes et merveilleuses. L'exploration de ce monde et la mise au point des dispositifs mécanico-logiques qu'on y dépose demandent une grande patience, car ils sont soumis à une contrainte stricte qui s'exprime en dix mots : « naissance si trois voisins, survie si deux ou trois voisins ».

Depuis cinq ans, des découvertes et des avancées remarquables ont profondément renouvelé son intérêt. Le but de cette rubrique est d'en donner une idée.

Le *Jeu de la vie* de Conway est un automate cellulaire qui fonctionne sur une grille infinie à cases carrées – les cellules – dont l'ensemble constitue l'espace plan où tout se déroule. Chaque cellule est vide ou occupée, morte ou vivante. Le temps s'y écoule de manière discrète : il y a un instant 0, un instant 1, un instant 2, etc. Chaque cellule n'a connaissance que de l'état de ses huit voisins directes, et la règle qui

détermine l'évolution des cellules d'une génération à la suivante est la règle rappelée plus haut.

Toutes les histoires qui se racontent et se raconteront concernant ce monde du *Jeu de la vie* sont définitivement fixées par ce mécanisme évolutif d'une absolue concision. C'est aux joueurs d'en faire quelque chose. Tout doit survenir à partir de cette physique déterministe réduite à l'extrême ne tolérant aucune exception, et Dieu sait qu'il en est survenu des miracles dans cet univers contraint... dont on découvre qu'il permet presque tout.

Pour jouer, vous pouvez vous contenter d'un papier, d'un crayon et d'une gomme, mais vous ne serez pas très efficace : depuis son origine, la recherche de configurations intéressantes se fait avec l'aide d'ordinateurs. C'est eux qui calculent l'évolution des configurations que vous proposerez, c'est eux qui vous aideront à en dessiner de nouvelles, ou même, qui vous guideront dans l'exploration d'immenses ensembles de configurations à la recherche d'interactions nouvelles entre composants déjà identifiés.

Les outils que les explorateurs du monde du *Jeu de la vie* ont mis au point sont d'une grande efficacité. Il y a quelques mois, après une longue période où je n'ai pas suivi ce qui se passait au pays de Conway, j'ai téléchargé le programme gratuit *Golly* (dû principalement à Andrew Trevorrow et Tomas Rokicki, voir <http://golly.sourceforge.net/>) et l'ai fait fonctionner en partant de configurations assez complexes proposées avec le programme. J'ai été stupéfait de la rapi-

dité avec laquelle le calcul et l'affichage des évolutions se faisaient. Le calculateur de nombres premiers [décrit dans la figure 4] dont la configuration initiale comporte 5 225 cellules produit des centaines de nombres premiers en quelques secondes, malgré l'augmentation de la taille de la configuration au cours des calculs. On obtient ainsi tous les nombres premiers inférieurs à 100 000 en cinq minutes avec une configuration qui atteint alors 10 millions de cellules.

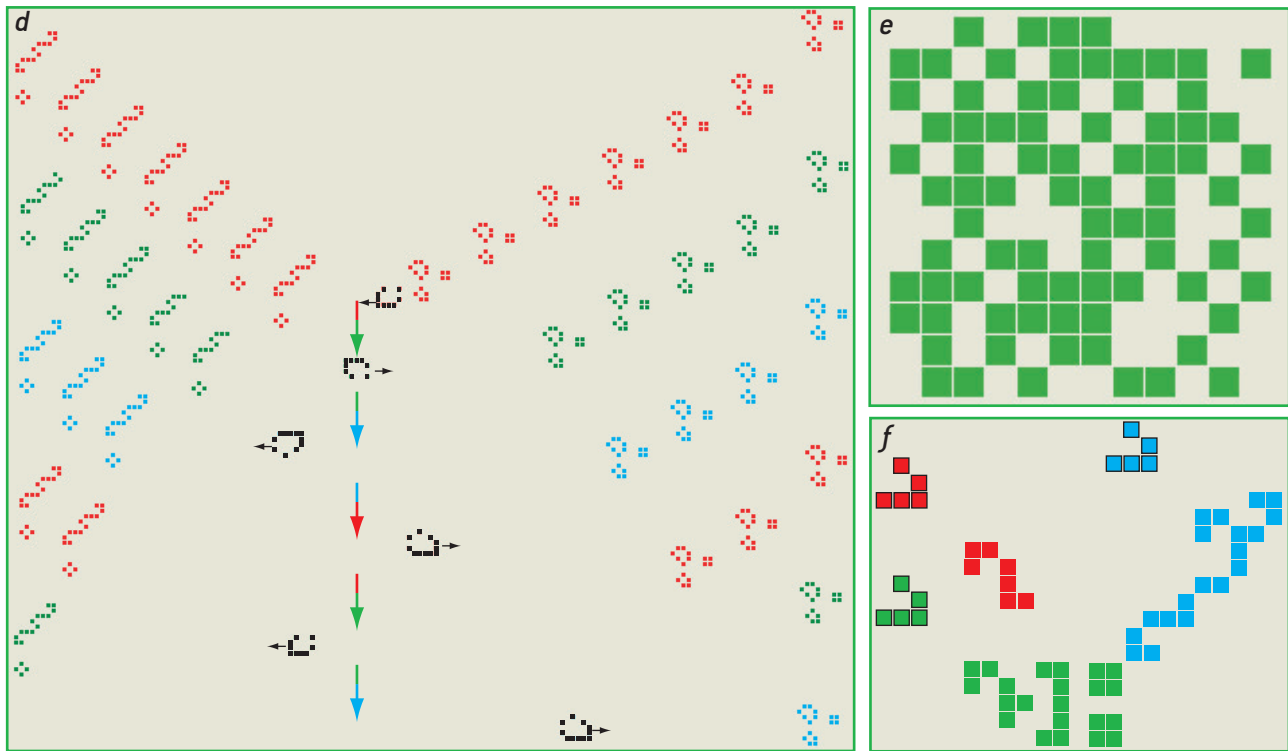
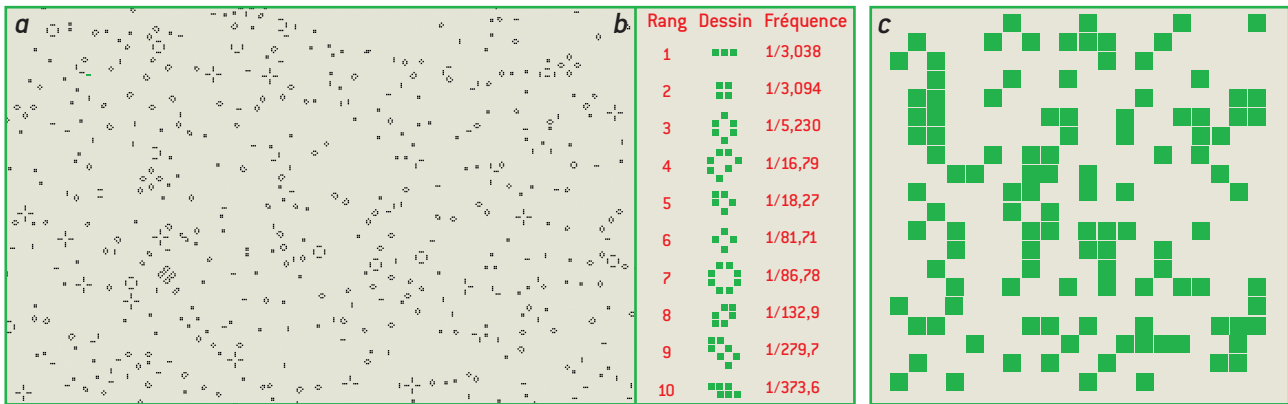
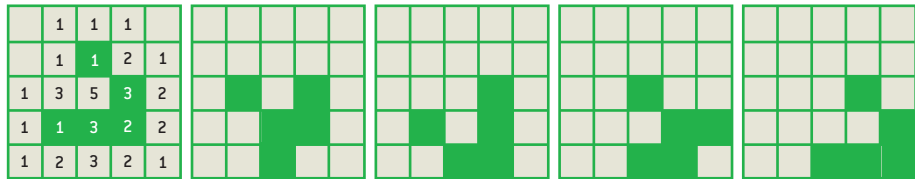
Réfléchir avant de programmer

La vitesse inouïe du programme *Golly* résulte d'une série de développements qui méritent d'être détaillés. William Gosper, découvreur du premier lance-glisseurs, le *Gosper-gun*, avait décrit au début des années 1980 les principes d'un algorithme nommé *Hashlife* fondé sur une série d'idées originales. Cet algorithme est très différent des algorithmes classiques pour exécuter des configurations du *Jeu de la vie* qui tentent d'en coder les règles en associant le calcul nécessaire à des primitives de bas niveau des ordinateurs, voire en le parallélisant.

L'algorithme *Hashlife*, lui, est fondé sur une analyse abstraite qui permet le repérage des régularités spatiales et temporelles des configurations : on ne fait pas plusieurs fois le même calcul si des configurations sont présentes en plusieurs exemplaires sur le plan, ou si le calcul a déjà été fait. *Hashlife* utilise de plus le

Regards

1. DANS LE JEU DE LA VIE DE CONWAY, les cellules évoluent sur une grille en suivant la règle : naissance si 3 voisins, survie si deux ou trois voisins. Ce « glisseur » (*cases vertes*) est une configuration de cinq cellules qui se déplace d'une case en diagonale en quatre générations.



2. CENDRES, CONFIGURATIONS STABLES, ABSORBEURS. Une configuration aléatoire qu'on laisse évoluer conduit à un ensemble d'objets isolés stables ou périodiques qu'on nomme des cendres (a). On a indiqué la fréquence des 10 premières configurations stables sur le tableau (b). Certaines figures disparaissent totalement en quelques générations. La configuration c, petite, mais dense, disparaît seulement au bout de 658 générations. Une autre (d) disparaît très lentement, mangée par un navire (en noir) qui va et vient entre deux séries d'objets fixes et alignés et qui lui-

même s'annihile une fois son travail terminé. Les différentes étapes du va-et-vient sont indiquées en couleurs différentes. Un jardin d'Éden est une configuration qui ne peut pas résulter de l'évolution d'une autre. Le plus petit jardin d'Éden connu (e) a été découvert par Nicolay Beluchenko en 2006. Les configurations stables capables de résister au choc d'un glisseur qui se précipite sur elles sont rares. Trois d'entre elles ont été représentées en couleurs différentes avec, à gauche, pour chaque absorbeur, les glisseurs (encadrés de noir) qu'il est capable d'engloutir (f).

Regards

calcul groupé des générations, d'abord par 8, puis par 64 ($= 8^2$), puis par 8^3 , etc. Un peu délicat dans sa conception et difficile à programmer, car utilisant systématiquement la technique de la récursivité, il illustre le pouvoir des mathématiques qui écrasent même les plus obstinés des tâcherons programmeurs, lesquels s'occupent d'un problème sans trop réfléchir en s'obstinant à ne voir que des mémoires et des instructions élémentaires là où il faut voir des structures partiellement répétitives dont la simplification exige astuce et abstraction.

À l'époque de son invention, une première version de l'algorithme *Hashlife* a été programmée au centre de recherche Xerox de Palo Alto, mais le programme était destiné à un type de machines spécialisées dans l'exécution d'applications en intelligence artificielle, les machines *Lisp*, qui disparaissent très vite. On ignore aujourd'hui si le programme de Gosper a fonctionné de manière satisfaisante sur ces architectures matérielles aujourd'hui dépassées. L'idée de l'algorithme *Hashlife*, plus ou moins oubliée, mais heureusement décrite par Gosper dans un article de 1984, a été reprise par Tomas Rokicki à partir de 2004, puis intégrée au programme *Golly* qui fonctionne sur tout type d'ordinateurs. Le résultat est proprement hallucinant et même les bons programmeurs sont surpris par les folles performances du simulateur.

Pour montrer comment des idées mathématiques soigneusement mises au point et affinées conduisent à des miracles de programmation, examinons un exemple. La configuration nommée *Breeder* (« éleveur ») est la première configuration connue possédant une croissance quadratique : le nombre de cellules vivantes augmente comme le carré du numéro de la génération. C'est une configuration complexe qui, au départ, possède 2 069 cellules vivantes. Elle comporte une tête qui avance en laissant derrière elle un alignement de lance-glisseurs dont chacun crée une nouvelle file de glisseurs. Le nombre de lance-glisseurs en action augmente linéairement en fonction du temps, et donc, le nombre de glisseurs croît quadratiquement.

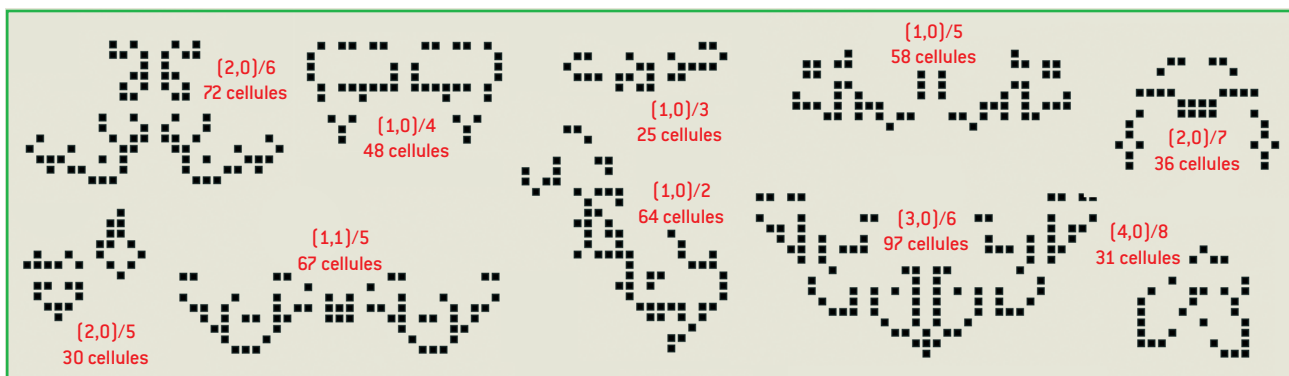
10⁵⁰ cellules et plus !

Au bout d'une minute de fonctionnement, sur un micro-ordinateur ordinaire, la configuration représentée à l'écran par *Golly* est la génération 10²⁸ et elle comporte 10⁵³ cellules vivantes. Cette configuration est présentée en petit, mais un zoom permet d'en observer tous les détails, qui ont donc bien été calculés. Pour mesurer l'exploit, il faut savoir qu'on évalue que la mémoire cumulée de tous les disques durs qui existent dans le monde représente entre 10²⁰ et 10²¹ octets et que la puissance cumulée de tous les microprocesseurs sur Terre aujourd'hui est elle aussi

située entre 10²⁰ et 10²¹ instructions par seconde : même dans 100 ans, un programme traditionnel ne fera pas ce que fait *Hashlife*. Le cœur de *Hashlife* est constitué d'astuces de compressions spatiales et temporelles. L'utilisation des méthodes mises en œuvre par *Hashlife* améliorerait sans doute le traitement d'autres problèmes de simulation et en accélérerait le fonctionnement. Les programmeurs qui ont à traiter des problèmes de ce genre doivent y regarder de près.

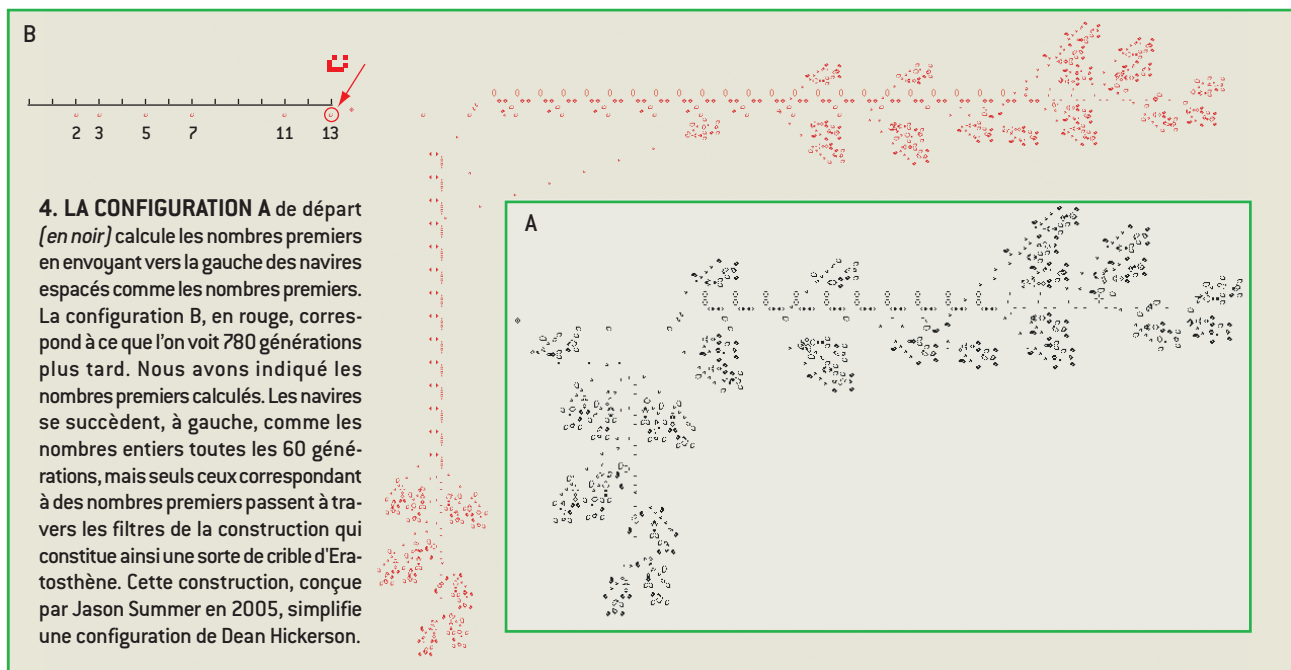
La réalisation d'un programme efficace d'exécution du *Jeu de la vie* a eu pour conséquence d'accélérer les découvertes de nouvelles configurations aux propriétés étonnantes.

D'abord, l'évolution des configurations aléatoires est assez bien connue. Une zone du plan remplie au hasard – on tire à pile ou face l'état initial de chaque cellule – qu'on laisse évoluer jusqu'à ce qu'il ne reste plus que des « cendres » (des configurations isolées stables ou de petites périodes), a toujours, à peu de choses près, le même aspect bien connu de tous ceux qui ont joué au *Jeu de la vie*. Ces cendres ont une densité moyenne de 0,02871 cellule vivante par cellule. La configuration isolée la plus souvent retrouvée dans des cendres (plus d'une fois sur trois) est le feu clignotant (*blinker*) composé de trois cellules côte à côte, verticales ou horizontales ; ensuite vient le bloc (un carré stable de 4 cellules). En tout, Achim



3. UN NAVIRE est une configuration qui se déplace d'un certain nombre de cases en restant identique à elle-même après un nombre caractéristique de générations. Le glisseur de la figure 1, à 5 cellules, est le plus petit des navires. Par définition, un navire de type $(n,m)/k$

retrouve sa forme initiale toutes les k générations après avoir avancé de n cellules dans un sens et de m cellules dans la direction perpendiculaire. Ainsi, le glisseur cité est noté $(1,1)/4$. Les navires représentés ici sont les plus petits connus pour chaque type de déplacement.



Flammenkamp a identifié 3 798 « objets naturels » et les a classés par probabilité d'apparition décroissante.

Cendres, annihilation, jardins d'Éden, etc.

Quand une configuration évolue, elle peut disparaître complètement : quel que soit le nombre n , il existe des configurations dont la durée de vie est supérieure à n , mais qui disparaissent entièrement. Une petite configuration qui ne meurt qu'au bout de 658 générations est montrée sur la figure 2c, ainsi que l'évolution d'une configuration 2d, aussi grande qu'on le veut, et qui s'annihile toute seule si on attend assez longtemps. Les résultats de J. Conway sur la possibilité de simuler tout calcul avec des configurations du *Jeu de la vie* démontrent l'indécidabilité du devenir d'une configuration : il n'existe aucun algorithme capable d'indiquer, pour toute configuration finie du *Jeu de la vie*, si elle s'annihilerait ou non.

Certaines configurations ne peuvent pas être le produit direct d'une autre configuration : on les nomme des jardins d'Éden. Le plus petit connu aujourd'hui a été décou-

vert en 2006 par Nicolay Beluchenko, il s'inscrit dans un carré 12×12 (voir la figure 2e).

Une question non résolue depuis 40 ans est : peut-on construire une configuration possédant un père (un antécédent direct), mais pas de grand-père (un père du père) ? On ne sait pas non plus s'il existe une configuration stable ne possédant qu'elle comme prédécesseur.

Noam Elkies a publié en 1997 un résultat conjecturé pendant près de 30 ans : la densité maximale que peut avoir une population infinie stable de cellules est $1/2$. Cette densité est atteinte par la configuration composée de lignes vides alternant avec des lignes pleines. Marcus Moore a démontré qu'une configuration stable confinée dans un rectangle $m \times n$ possède au plus $(mn+m+n)/2$ cellules.

Certaines configurations stables ne changent en rien d'une génération à l'autre, ce sont des *still-life* (natures mortes). On en connaît des milliers et même des séries infinies. Certaines ont, de plus, la propriété de se reconstituer quand un glisseur (la configuration la plus simple qui se déplace, voir la figure 1) interagit avec elle. Ces configurations « absorbantes » permettent de garder le contrôle de ce qui

se passe lors de constructions complexes (voir la figure 2f).

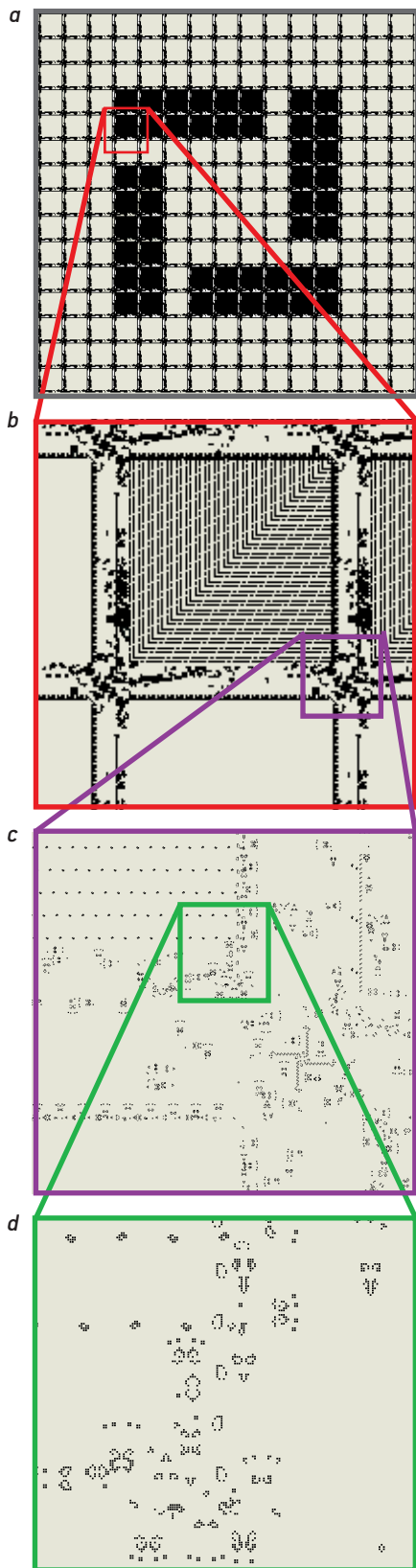
Configurations oscillantes et navires

Les configurations oscillantes (après une évolution qui dure quelques générations, elles retrouvent leur forme initiale au même endroit) peuvent avoir pour période 2 (le feu clignotant), mais aussi 3, 4, 5. On a découvert que tous les entiers sauf peut-être 19, 23, 31, 37, 38, 41, 43 et 53 sont des périodes possibles d'oscillateurs. Pour les huit entiers manquants, des recherches se poursuivent, mais les explorations déjà menées ont été très approfondies et les réponses (positives ou négatives) concernant les nombres restants seront certainement difficiles à obtenir.

La recherche de navires (après une évolution qui dure quelques générations, un navire retrouve sa forme initiale déplacée de quelques cases) est un sujet de recherche central pour le *Jeu de la vie*, car ces navires permettent la construction de machines complexes comme le *Breeder* ou le calculateur de nombres premiers.

Jusqu'en 1989, on ne connaissait que deux types de navires : des navires

Regards



5. UNE MÉTACELLULE est une configuration de forme carrée. Lorsqu'elle est disposée en plusieurs exemplaires pour constituer une grille [a], la métacellule se comporte vis-à-vis des autres métacellules comme dans le *Jeu de la vie* : selon que les métacellules ont leur intérieur vide ou rempli de navires, la cellule correspondante est morte ou vivante. La configuration représentée ici fonctionne parfaitement et rapidement malgré les 7 408 795 cellules de base qui la constituent. En zoomant [b, c, d], on aperçoit les détails de l'incroyable construction qui est l'imbrication d'une multitude de mécanismes subtils, fonctionnant avec la règle du *Jeu de la vie*.

diagonaux avançant à la vitesse d'une case toutes les quatre générations et des navires avançant à la vitesse d'une case toutes les deux générations horizontalement ou verticalement. De récentes découvertes ont enrichi l'armada de ces navires.

Le navire de 11 880 063 cellules, construit en 2004 par Gabriel Nivasch, se déplace de 102 cases horizontalement toutes les 270 générations à la vitesse de $17/45^\circ$ de cellule par génération. Sa taille est si grande que sa construction a été réalisée en utilisant un programme d'assemblage particulier chargé de combiner 51 morceaux conçus séparément. Contrairement aux autres navires qui se déplacent rapidement sur l'écran, au début il faut à ce navire géant plusieurs secondes pour changer de génération. Puis, grâce à *Hashlife*, quand les premières étapes sont calculées et mémorisées, le calcul s'accélère et l'avancée de l'énorme navire devient perceptible.

Parmi les belles découvertes récentes concernant les navires, mentionnons les navires gris et les bulles. Les navires gris comportent une partie centrale composée de bandes verticales (une bande pleine, une bande vide). Leur allure qui évoque des insectes ou des animaux marins est amusante et leur déplacement rapide sur le plan de Conway constitue un spectacle étrange (<http://pentadecathlon.com/lifeNews/index.php>). Les bulles sont aussi une très belle découverte récente : ce sont des navires qui se déplacent non pas sur le plan vide, mais sur le plan recouvert de la construction stable de densité maximale (une bande pleine, une bande vide, etc.). Ces navires-bulles sont particulièrement délicats et le moindre incident de parcours (par exemple créé en changeant une cellule) non seulement casse le navire, comme cela se produit pour les autres navires, mais embrase tout l'espace qui s'autodétruit donnant au bout de plusieurs mil-

liers de générations des cendres analogues à celles évoquées plus haut.

La première configuration à croissance quadratique fut considérée dans les années 1970 comme un fantastique exploit. Les progrès faits sur cette question sont totalement inattendus. Le premier *Breeder* était une configuration composée de plusieurs milliers de cellules au départ. Il a été simplifié plusieurs fois pour conduire à la fin des années 1990 à une jolie configuration de 200 cellules qui, en s'agrandissant dans quatre directions à la fois, recouvre le plan du motif rayé de densité maximale déjà évoqué. Ce « recouvrement du plan » (*space filler*) semblait impossible à battre. Pourtant, au fur et à mesure que de nouvelles expériences ont pu être menées, on a découvert que la taille d'une configuration à croissance quadratique pouvait être bien plus réduite.

Concentration cellulaire

La configuration à croissance quadratique la plus petite connue aujourd'hui a été découverte par Nick Gotts le 17 mars 2006. Elle ne comporte que 26 cellules au départ. Elle a deux caractéristiques remarquables. D'abord les 26 cellules du départ sont composées de deux parties éloignées l'une de l'autre de plus de 15 000 cellules vides : il est presque impossible de voir les 26 cellules simultanément, car il faut utiliser une grande échelle de représentation pour faire tenir toute la configuration sur l'écran.

La seconde propriété remarquable est que le mécanisme de croissance quadratique ne se met en place qu'après environ un million de générations ; le nombre de cellules atteint alors un million. *Golly* grâce à *Hashlife* permet tous les calculs nécessaires en quelques secondes. En attendant un peu plus, on constate qu'à la génération 10^9 , la population comporte

Regards

10^{11} cellules, puis qu'à la génération 10^{12} la population est de 10^{17} cellules. En quelques minutes, on arrive à la génération 10^{23} qui possède alors 10^{40} cellules.

La possibilité de construire et de faire fonctionner de grands assemblages de cellules a permis deux autres exploits : la construction explicite de systèmes universels de calcul, et la définition d'une métacellule fonctionnelle.

Calcul universel et métacellule

L'automate cellulaire du *Jeu de la vie* est « computationnellement universel », ce qui signifie qu'on a la possibilité de mener tout calcul faisable par programme à l'aide d'une configuration du *Jeu de la vie*. Ce résultat a été établi dans les années 1970. Cependant, entre la démonstration de « Il existe une configuration qui programme f » et la construction explicite d'une configuration qui calcule f , le pas est souvent grand, surtout si f est une fonction compliquée. Deux méthodes générales sont maintenant proposées pour programmer toute fonction programmable f : une construction de machine de Turing par Paul Rendel en 2000 et une machine à registres (c'est un mécanisme de calcul dont la mémoire est constituée par des nombres entiers) par Paul Chapman en 2002.

Ces constructions, qui possèdent chacune plusieurs dizaines de milliers de cellules, rendent assez facile aujourd'hui la programmation à l'aide de configurations du *Jeu de la vie* de n'importe quelle fonction programmable f . Le résultat théorique est devenu pratique... ce qui ne veut pas dire utile, car ces constructions sont des défis de programmation qui n'ont pas d'applications directes.

Le dernier exploit est celui de la mise au point d'une métacellule. Il s'agit d'une configuration d'environ 30 000 cellules dont l'ensemble occupe une forme à peu près carrée. Lorsque la configuration fonctionne, elle peut se remplir de navires, ce qui lui donne un aspect grisé, ou au contraire rester vide, auquel cas on n'en voit que le pourtour. La métacellule a été

conçue pour qu'en juxtaposant plusieurs exemplaires de cellules, celles-ci interagissent exactement comme interagissent les cellules du *Jeu de la vie*.

Le plus étonnant est que ces métacellules fonctionnent et qu'en se plaçant à la bonne échelle, on voit le méta-automate cellulaire fonctionner à toute vitesse grâce à l'algorithme *Hashlife* de Golly. Un de mes collègues en découvrant cela s'est écrié : « C'est incroyable, jamais je n'aurais pensé possible un truc pareil. »

Le *Jeu de la vie* montre que le continu utilisé par les physiciens pour modéliser l'espace et le temps est peut-être superflu. Le fini local et déterministe des automates cellulaires, même en ne considérant que les plus simples d'entre eux, autorise le déploiement d'une phénoménologie d'une infinie variété. Elwyn Berlekamp, John Conway et Richard Guy vont plus loin. Pour eux : « Il est probable qu'en remplissant une partie assez grande du plan infini du *Jeu de la vie* par une configuration aléatoire, alors, après un long moment, émergeront des êtres autoreproducteurs intelligents qui peupleront l'espace. »

Intérêt philosophique

Cette idée qu'un monde déterministe régi par des lois d'une simplicité absolue est suffisant pour la physique et même la vie et l'intelligence est discutée par le philosophe Daniel Dennett. Il considère que de tels modèles d'univers n'excluent pas ce que nous appelons la conscience et le libre arbitre. Pour lui, tout philosophe devrait étudier soigneusement le *Jeu de la vie* et ce n'est qu'en réussissant à penser les idées de conscience et de libre arbitre dans un tel monde que nous en comprendrons la véritable nature.

Il paraît évident aujourd'hui qu'au-delà des philosophes, ce sont tous les mathématiciens, physiciens, informaticiens et biologistes qui disposent avec le *Jeu de la vie* d'un sujet d'expérimentation et de réflexion d'une exceptionnelle profondeur : bien plus qu'un jeu, c'est un modèle simplifié abstrait de l'univers dont la connaissance et la maîtrise enrichissent notre compréhension générale du monde dans sa totalité. ■

L'AUTEUR

Jean-Paul DELAHAYE est professeur à l'Université de Lille et chercheur au Laboratoire d'informatique fondamentale de Lille (LIFL).

✓ SUR LE WEB

<http://golly.sourceforge.net/Hashlife>
<http://en.wikipedia.org/wiki/Hashlife>

<http://tomas.rokicki.com/hlife/>
<http://www.ddj.com/hpc-high-performance-computing/184406478>

http://en.wikipedia.org/wiki/Conway's_Game_of_Life

✓ BIBLIOGRAPHIE

William Gosper, *Exploiting Regularities in Large Cellular Spaces*, Physica 10D, 1984.

Elwyn Berlekamp, John Conway, Richard Guy *Winning Ways for your Mathematical Plays*, Academic Press, New York, 1982. Nouvelle édition : A. K. Peter, 2003.

C. Corge, *Machines de Turing et automates cellulaires. Du trait gravé au très animé*, Ellipses Marketing, 2008.

Daniel Dennett, *Freedom Evolves*, Allen Lane, 2003.

N. Elkies, *The still-Life Density Problem and its Generalizations in Voronoi's Impact on Modern Science*, P. Engel, H. Syta, eds. Institute of Math., Kyiv, pp. 228-253, 1998.

M. Gardner, *Wheels, Life and other Mathematical Amusements*, W. H. Freeman, 1983.

William Poundstone, *The Recursive Universe. Cosmic Complexity and the Limit of Scientific Knowledge*, Oxford University Press, 1985.