LOGIQUE & CALCUL

Le problème du sudoku

La donnée de 16 chiffres dans une grille de sudoku est insuffisante pour que la solution du problème soit unique. Pour le prouver, il a fallu énumérer toutes les grilles, en usant d'astuces afin de raccourcir les calculs.

Jean-Paul DELAHAYE

C'est une illusion de croire que la puissance des machines nous dispense de réfléchir. À propos du problème des données minimales au sudoku, nous allons voir que la limite du faisable est vite atteinte, et que pour la repousser, il faut de l'ingéniosité, des mathématiques et quand même beaucoup de patience.

Rappelons les règles du sudoku et fixons quelques termes. Une « grille de sudoku complète » est un tableau carré de $9 \times 9 = 81$ cases contenant chacune un chiffre de 1 à 9 et tel que :
(a) chaque ligne et chaque colonne contiennent chaque chiffre une fois exactement ;
(b) chaque sous-tableau 3×3 (résultat du découpage de la grille en neuf carrés de neuf cases) contient chaque chiffre une fois exactement.

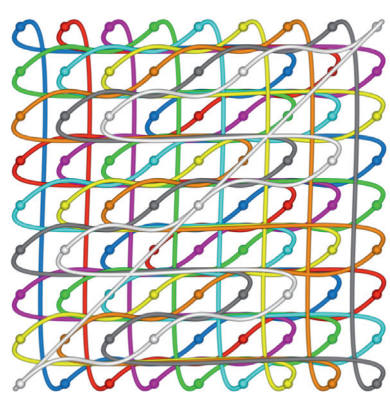
Une grille partielle est un « problème de sudoku correct » s'il existe une façon unique de compléter la grille en une grille de sudoku complète, ce qui est le but du jeu.

Précisons qu'au sudoku, l'ordinateur est gagnant sur l'humain : il existe un grand nombre de programmes permettant à l'ordinateur de battre de vitesse le meilleur humain. Les bons programmes trouvent la solution d'un problème, aussi difficile soit-il, en moins d'un millième de seconde. On joue donc pour le plaisir de l'exercice.

Quand on cherche à formuler un énoncé, si l'on spécifie trop peu de cases, la grille partielle ne constitue pas un problème correct (c'est évident si on ne retient qu'une case), car plusieurs solutions seront possibles. Dans les journaux, les grilles proposées comportent environ 25 cases remplies. On sait cependant qu'il existe des grilles correctes de sudoku ne comportant que 17 données *(voir l'encadré ci-contre)*.

Le problème des données minimales est : *Quel est le plus petit nombre de données d'un problème de sudoku correct ?*

L'exemple à 17 données montre que ce minimum est 17 ou moins. On a longtemps cherché, sans succès, des problèmes corrects à 16 données ; on a donc conjecturé que 17 est la réponse.



VISUALISATION D'UN SUDOKU par Jean-François Colonna. Chaque couleur est associée à un chiffre.

Au moins 16 ou 17 cases ?

Le problème est resté ouvert jusqu'à ce que, en décembre 2011, la réponse soit fournie par Gary McGuire, de l'Université de Dublin, et son équipe. À la suite d'un calcul étalé sur une année et correspondant à 800 années de calcul d'un processeur unique, la conclusion fut qu'il n'existe aucune grille correcte de sudoku à 16 données et donc que le nombre minimal de données d'un problème correct de sudoku est 17.

Il s'agit d'un théorème mathématique, et puisque c'est la question mathématique concernant le sudoku qui a demandé le plus d'effort, on l'appellera « Le théorème du sudoku ». Le travail qui a conduit à l'énoncer est une démonstration. Comme cela se produit maintenant de plus en plus souvent, il s'agit d'une démonstration avec un ordinateur (*voir l'encadré page 78 pour d'autres exemples*). Le détail des étapes du calcul n'a pas été publié, car cela donnerait un document d'une longueur colossale. L'article rendant compte de la preuve n'est donc que la description de la méthode utilisée, comportant, entre autres, les énoncés et démonstrations des propositions purement mathématiques utiles, mais pas assez d'éléments pour que l'on puisse vérifier le théorème sans tout reprogrammer et recalculer.

Des versions préliminaires de l'article étaient disponibles depuis un moment, mais il n'a été publié dans sa forme définitive que le 12 juin 2014, dans la revue *Experimental Mathematics*. Malgré cette publication officielle, essentielle pour qu'un résultat soit considéré comme sérieux, et une vérification du calcul menée par une autre équipe (en septembre 2013), certains chercheurs, dont Joshua Cooper de l'Université de Caroline du Sud, doutent du statut de ce théorème dont la « démonstration », d'un type particulier, exclut difficilement le risque d'erreur. Nous y reviendrons.

Dans un premier temps, présentons la méthode de preuve utilisée par G. McGuire et les autres équipes qui ont abordé le problème. Nous insisterons sur l'utilité d'une analyse mathématique préliminaire, la collaboration d'autres équipes (ayant mené d'autres calculs volumineux et transmis des programmes à G. McGuire et son équipe), la rigueur méthodologique indispensable, le délicat savoir-faire nécessaire à la programmation et à la réalisation des calculs et, enfin, l'incroyable quantité de calculs effectués.

Existe-t-il une grille complète G dont on peut extraire 16 données qui, quand on les complète (en respectant les règles du sudoku), donnent nécessairement la grille G ?

Des grilles intéressantes

La grille *a* comporte 17 données et n'a qu'une seule solution, *b*. Ce type de grille a posé la question : peut-on faire plus économique et ne fixer que 16 données dans une grille tout en préservant l'unicité de la solution ?

Le calcul de Gary McGuire et de son équipe montre que pour avoir une solution unique, il faut toujours spécifier au moins 17 cases. Mais parfois, 17 données ne suffisent pas. La grille *c* fournit un exemple où 18 données sont nécessaires (aucune sous-grille à 17 données n'est correcte). Cette grille permet aussi d'illustrer la puissance de la méthode des *sous-ensembles inévitables* au cœur des programmes de G. McGuire.

Considérons les deux colonnes verticales de trois chiffres en jaune en haut à gauche (1, 4, 7) et au centre (4, 7, 1). Si on les échange, on obtient une nouvelle grille complète de sudoku. Cela prouve que tout problème conduisant à cette grille doit comporter une donnée parmi les six cases concernées. Il en va de même pour les deux colonnes jaunes en haut au centre (4, 7, 1) et à droite (7, 1, 4), ainsi que pour les colonnes (1, 4, 7), et (7, 1, 4) en jaune en haut. Il en résulte que parmi les neuf cases jaunes en haut, deux doivent être données dans tout problème ayant cette grille comme solution unique.

C'est aussi vrai pour les autres paquets de neuf cases (en bleu, en rouge) de la zone du haut, de la zone du milieu et de la zone du bas. Cela fait en tout neuf paquets de neuf cases, et pour chacun deux données au moins à retenir. Tout problème conduisant à *c* doit donc comporter au moins 18 données. Il n'y aura donc, dans ce cas particulier, aucun énoncé correct à 17 données.

Pour toute grille, un raisonnement montre qu'un problème correct comporte au moins huit données. S'il ne comporte que sept données ou moins et doit conduire à une grille G fixée, ces données ne mentionnent pas deux chiffres, par exemple le 8 et le 9, et donc les sept données ne peuvent pas distinguer la grille complète visée G et la même grille G' où l'on a interverti les 8 et les 9. Les sept données ne constituent donc pas un problème correct.

De tels raisonnements mathématiques ne permettent pas aujourd'hui d'aller plus loin : on ne sait même pas prouver sans machine que huit données ne suffisent jamais à composer un problème correct.

a

1						5	2	
				7	8			
						6		
	9			4				
			5			1		
	7							
		6	2					
	4						7	8
								3

b

1	6	8	4	9	3	5	2	7
4	5	2	6	7	8	3	9	1
9	3	7	1	5	2	6	8	4
6	9	3	8	4	1	7	5	2
8	2	4	5	6	7	1	3	9
5	7	1	3	2	9	8	4	6
7	8	6	2	3	4	9	1	5
3	4	5	9	1	6	2	7	8
2	1	9	7	8	5	4	6	3

c

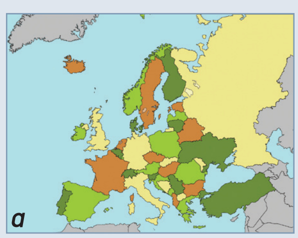
1	2	3	4	5	6	7	8	9
4	5	6	7	8	9	1	2	3
7	8	9	1	2	3	4	5	6
2	3	4	5	6	7	8	9	1
5	6	7	8	9	1	2	3	4
8	9	1	2	3	4	5	6	7
3	4	5	6	7	8	9	1	2
6	7	8	9	1	2	3	4	5
9	1	2	3	4	5	6	7	8

Les démonstrations par ordinateur

Le recours à l'ordinateur pour démontrer des résultats mathématiques est de plus en plus fréquent. Souvent, l'ordinateur n'intervient qu'associé à des résultats purement mathématiques démontrés à la main et qui posent les bases des algorithmes confiés à la machine. C'est le cas pour le théorème des 17 données du sudoku.

Les démonstrations de théorèmes dépendant de calculs informatiques ont souvent été jugées insatisfaisantes, car d'une part, aucun humain seul ne peut les contrôler, d'autre part, les programmes et leurs calculs sont susceptibles d'erreurs de toutes sortes. Pour limiter ce risque d'erreurs, on a développé une méthode de « preuve formelle » par ordinateur, qui diminue considérablement ce risque et qui permet même une meilleure garantie de justesse que les preuves humaines dans certains cas. (voir cette rubrique de juin 2011, *Du rêve à la réalité des preuves*). On met en œuvre des « assistants de preuves », des logiciels spécialisés que les mathématiciens utilisent pour écrire pas à pas toutes les étapes d'une démonstration, sans en oublier aucune (oubli qui est possible quand la preuve est menée par un humain en utilisant de nombreux raccourcis).

La preuve complète mise au point est vérifiée par une procédure indépendante et très pointilleuse ; éventuellement, cette vérification est confiée à plusieurs autres logiciels de contrôle. Les logiciels de contrôle peuvent eux-mêmes avoir été soumis à des contrôles très stricts « prouvant » leur bon fonctionnement. On parvient ainsi à une certitude quasi parfaite qu'une preuve est correcte.

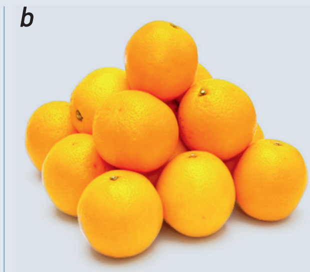


a

Parmi les théorèmes soumis à cette vérification par la méthode des assistants de preuves, deux ont retenu particulièrement l'attention. L'un est le théorème des quatre couleurs, qui indique que « toute carte peut être coloriée avec quatre couleurs sans que deux pays voisins soient colorés de la même façon » *(a)* ; il a été démontré en 1976 et vérifié formellement en 2005 par une équipe de l'Inria.

L'autre est le théorème de Thomas Hales qui résout la conjecture de Kepler sur l'empilement le plus dense des sphères *(b)*, démontré en 1998 et formellement vérifié par T. Hales et son équipe en 2014.

b



Malheureusement, dans le cas du résultat de Gary McGuire sur le sudoku, la méthode des assistants de preuve ne pourra sans doute pas s'appliquer avant longtemps : le calcul effectué est énorme et, à moins d'avancées mathématiques nouvelles, une preuve formelle ne pourra que refaire le calcul dans un cadre plus rigoureux pour éviter les erreurs, ce qui demande encore plus de calculs, chose impossible à envisager aujourd'hui.

La méthode de résolution se décompose en trois étapes : (A) énumérer toutes les grilles complètes de sudoku ; (B) pour chacune, considérer toutes les façons d'en extraire 16 données ; et (C) pour chaque jeu de 16 données ainsi extrait, faire fonctionner une méthode de résolution de manière à trouver au moins deux solutions différentes.

Si toutes les grilles complètes passent ce test, alors on aura démontré que 16 données ne sont jamais suffisantes et que 17 est donc la bonne réponse. Le faire brutalement comme je viens de le décrire demanderait des siècles de calcul.

Trois remarques économisent du temps de calcul et, bien employées, donnent une durée de calcul raisonnable.

Point (A) : limiter l'énumération des grilles complètes qu'on fait dérouler.

Point (B) : trouver des méthodes évitant d'énumérer tous les sous-ensembles de 16 éléments d'un ensemble de 81 éléments, car il y en a trop !

Point (C) : disposer d'un algorithme de test le plus efficace possible pour tester si une grille partielle est un problème correct.

Un catalogue intelligent et complet des grilles

Remarquons qu'on aurait pu procéder autrement : (A') énumérer toutes les grilles partielles possibles de 16 données, puis (B') montrer que chacune a plusieurs solutions. Toutefois, la théorie des « ensembles inévitables », qui permet de réduire considérablement les calculs de (B) (expliquée plus loin), rend la méthode en trois étapes (A-B-C) plus efficace que la méthode en deux étapes (A'-B').

Le nombre de grilles complètes différentes semble avoir été connu par Mark Brader dès 2003, mais c'est en 2006 qu'un document de Bertram Felgenhauer et Frazer Jarvis l'a justifié. Ce nombre est énorme et égal à $9! \times 2^{13} \times 3^{4} \times 27\,704\,267\,971 = 6\,670\,903\,752\,021\,072\,936\,960 = 6{,}6... \times 10^{21}$.

C'est, en ordre de grandeur, la mémoire cumulée numérique de tous les dispositifs informatiques (téléphones compris), et ces dispositifs opèrent tous ensemble environ 10^{21} instructions par seconde. Ces chiffres sont imprécis, en particulier parce qu'on ne peut pas savoir quelles machines ont cessé de fonctionner, mais à un facteur 10 près, ils indiquent où nous en sommes.

Traiter une grille de sudoku pour savoir si elle possède des sous-grilles correctes à 16 données est un gros travail. Procéder naïvement, même en mettant tous les dispositifs de calcul terrestres à la tâche, n'aurait même pas permis de résoudre le problème. Réduire le nombre de grilles à étudier est le premier point où les mathématiques ont aidé l'équipe de G. McGuire.

Si, dans une grille complète de sudoku, on change tous les 1 en 9 et tous les 9 en 1, on obtient une autre grille complète de sudoku. Si la première admet des problèmes corrects à 16 données, c'est aussi le cas de la seconde : partant d'un problème correct pour la première, on change les 1 en 9 et les 9 en 1, et on obtient un problème correct pour la seconde. Il suffit donc d'étudier l'une des deux. L'échange des 1 et des 9 n'est pas la seule transformation qui conduit à trouver des grilles équivalentes. En fait, toute permutation des neuf chiffres 1, 2,..., 9 permet, à partir d'une grille complète, d'en construire une autre équivalente pour notre problème. Pour chaque grille, ce procédé donne $9\,! = 9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 =$ 362 880 grilles équivalentes. Voilà déjà une belle réduction du calcul à entreprendre.

Cependant, la permutation des chiffres n'est pas la seule façon d'obtenir des grilles équivalentes. Voici la liste de quelques autres opérations qui, appliquées à une grille complète, donnent des grilles équivalentes.

– Choisir deux bandes horizontales et les échanger. Une bande horizontale est une série de trois carrés 3×3 placés les uns à côtés des autres (il y en a trois : une en haut, une au milieu, une en bas).

– Échanger deux bandes verticales. Une bande verticale est une série de trois carrés 3×3 placés les uns au-dessus des autres (il y en a trois : une à droite, une au centre, une à gauche).

– Échanger deux colonnes d'une même bande verticale.

– Échanger deux lignes d'une même bande horizontale.

– Transposer la grille (la ligne 1 devient la colonne 1, la ligne 2 devient la colonne 2, etc.).

Finalement, une grille est équivalente à un très grand nombre d'autres. Les grilles se regroupent en classes et il suffit de traiter une grille de chacune des classes.

9	3	7	8	5	6	2	4	1
5	6	2	1	9	4	3	8	7
4	8	1	2	7	3	5	6	9
8	2	3	6	4	7	9	1	5
6	1	5	9	3	2	4	7	8
7	4	9	5	8	1	6	2	3
3	7	8	4	6	9	1	5	2
1	9	6	7	2	5	8	3	4
2	5	4	3	1	8	7	9	6

DEUX « ENSEMBLES INÉVITABLES » sont indiqués *(en bleu et en orange)* dans cette grille. Chacun nécessite une donnée au moins pour que la solution au sudoku soit unique.

L'utilisation d'outils de la théorie des groupes, dont le lemme de Burnside *(voir Le miraculeux « lemme de Burnside » dans la rubrique de décembre 2006)*, a permis de connaître le nombre exact de grilles complètes non équivalentes : il y en a 5 472 730 538 $\approx 5{,}5 \times 10^9$.

Glenn Fowler, chercheur aux Laboratoires AT&T, a constitué un catalogue complet des grilles en ne retenant qu'une seule grille par classe d'équivalence. Un algorithme de compression spécialement conçu pour les grilles de sudoku a permis de stocker tout le catalogue de G. Fowler sur six gigaoctets, ce qui est assez remarquable puisque cet algorithme permet de n'utiliser qu'un octet (8 bits) environ par grille.

Chercher les « ensembles inévitables »

Si l'on s'y prend sans finesse, pour chacune des grilles retenues, il faut envisager toutes les façons d'extraire 16 données. Or il y a 33 594 090 947 249 085 $\approx 3{,}4 \times 10^{16}$ façons de choisir 16 cases parmi 81. Cela conduirait à résoudre un grand nombre de problèmes de sudoku pour découvrir à chaque fois qu'il y a deux solutions ou plus. Ce nombre serait énorme, car égal à : 5 472 730 538 $\times$ 33 594 090 947 249 085 = 183 851 407 423 359 414 572 057 730 $\approx 1{,}8 \times 10^{26}$.

Nous savons que ce nombre est bien au-delà de ce qui est envisageable, même en utilisant toute la puissance de calcul disponible sur la planète. Les mathématiques à nouveau ont dû proposer une idée intéressante qui s'applique à d'autres problèmes, la « méthode des ensembles inévitables ». Expliquons cette idée sur l'exemple de la grille complète G ci-contre.

Pour construire un problème correct ayant pour solution cette grille G, il faut que l'une des données du problème soit l'une des quatre cases en bleu, en haut de la grille. En effet, il existe une autre grille complète G' très proche de G obtenue en échangeant seulement, dans les quatre cases en bleu, les 9 et les 5. Si les données de la grille partielle n'indiquent rien concernant ces quatre cases, il sera impossible

L'histoire de la conjecture

L'étude du problème du sudoku minimal par Gary McGuire et son équipe a commencé en 2005, par la mise au point d'un premier programme recherchant, pour une grille complète fixée, toutes les sous-grilles partielles correctes.

Cependant, c'est seulement en 2009, voyant que la solution du problème était peut-être atteignable, que G. McGuire et ses collègues ont entrepris des travaux systématiques. En effet, Gordon Royle avait à cette époque réuni une collection importante de problèmes corrects différents à 17 données. Il recherchait en même temps, mais en vain, un problème ne demandant que 16 données. Cette situation rendait plausible la conjecture que 16 données ne suffisent jamais. Une étonnante grille ayant 29 sous-grilles correctes à 17 données attira l'attention *(ci-dessous)*. On ne connaît aujourd'hui encore aucune grille complète ayant 30 sous-grilles correctes à 17 données. Encouragé par ce grand nombre de sous-grilles correctes à 17 données, on espéra un moment en extraire une sous-grille correcte à 16 données. En vain, ce qui fut considéré comme un indice de plus en faveur de la conjecture.

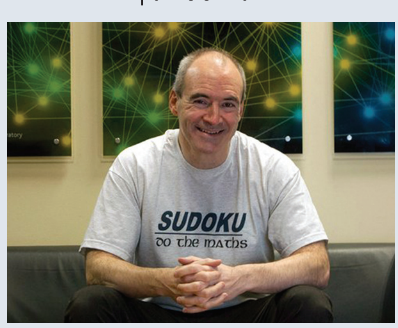
Avant d'en arriver au résultat que 16 données sont insuffisantes, un calcul mené par une équipe de l'Université de Graz en Autriche montra, en 2008, que 11 ne suffisent pas. Le calcul entrepris par cette équipe pour montrer que 12 ne suffisent pas fut abandonné sans aboutir. On était donc loin du but. Rappelons toutefois que le raisonnement mathématique pur ne réussit pas mieux que 7 *(voir l'encadré page 77)* ; la démonstration par ordinateur était donc déjà en avance.

En 2008, une jeune fille de 17 ans proposa une preuve mathématique que 16 données sont insuffisantes, mais on découvrit rapidement une erreur dans son raisonnement, que personne n'a su corriger.

En 2010, une équipe de l'Université Chiao Tung à Taïwan entreprit de s'attaquer au problème des 16 données en utilisant une méthode de calcul distribué. Elle évalua alors qu'avec les outils dont elle disposait, le problème serait résolu en menant un calcul de 2 400 années-processeurs, par exemple, en faisant calculer 2 400 ordinateurs de table chacun pendant un an. Le calcul a abouti en septembre 2013 (après celui de G. McGuire). C'est la seule vérification à ce jour du théorème des 16 données.

Le calcul de G. McGuire et de son équipe se déroula sur toute l'année 2011, opérant l'équivalent de 800 années-processeurs de calculs. Pour chaque grille de sudoku complète, le temps de traitement a été enregistré. Il a fallu en moyenne 3,6 secondes par grille. L'article rendant compte du résultat a été publié en 2014.

6	3	9	2	4	1	7	8	5
2	8	4	7	6	5	1	9	3
5	1	7	9	8	3	6	2	4
1	2	3	8	5	7	9	4	6
7	9	6	4	3	2	8	5	1
4	5	8	6	1	9	2	3	7
3	4	2	1	7	8	5	6	9
8	6	1	5	9	4	3	7	2
9	7	5	3	2	6	4	1	8



LE MATHÉMATICIEN GARY MCGUIRE *(ci-dessus)* et un sudoku ayant 29 sous-grilles correctes à 17 données *(ci-contre)*.

de savoir si la solution est G ou G', et donc le problème sera incorrect. Il en va de même avec les six cases en orangé : permuter les deux rectangles de trois données (7, 2, 1) et (1, 7, 2) donne une grille G'' qui n'est exclue que si l'on spécifie au moins une des six cases.

De tels ensembles se dénomment des *ensembles inévitables*. Si, pour une grille complète G, on en trouve *k*, et qu'ils sont disjoints, alors on est certain que toute grille partielle conduisant à G comportera au moins une donnée dans chaque ensemble inévitable et donc en comportera au moins *k*. Lorsque certains ensembles inévitables ont une intersection commune, la situation, un peu plus difficile, impose tout de même un nombre minimum de données à toute grille partielle conduisant à G.

La lutte finale

Le travail pour traiter une grille complète G et montrer qu'il faut toujours au moins 17 données se ramène alors à la recherche et à l'exploitation d'ensembles inévitables aussi nombreux que possible. Après divers essais, les chercheurs ont adopté la stratégie suivante, qui a été appliquée à chacune des 5,5 milliards de grilles complètes du catalogue de G. Fowler.

(a) Déterminer (en mettant au point les algorithmes nécessaires) tous les ensembles inévitables de moins de 13 cases. Trouver tous les ensembles inévitables sans limitation de taille, ce qui *a priori* semble meilleur, conduit à plus de calculs au total.

(b) En déduire (toujours en mettant soigneusement au point les algorithmes nécessaires) toutes les sous-grilles de 16 données convenables pour cette famille d'ensembles inévitables, c'est-à-dire ayant au moins un élément dans chaque ensemble inévitable. Cette partie retient considérablement moins de grilles partielles à 16 éléments que les $3{,}4 \times 10^{16}$ grilles partielles *a priori* envisageables. En moyenne, une grille complète de sudoku a 360 ensembles inévitables de moins de 13 cases.

(c) Pour les grilles partielles retenues, utiliser un programme de test qui indique si oui ou non il y a une solution unique.

Pour mettre tout cela en place et le programmer aussi efficacement que possible, il a fallu développer quelques résultats mathématiques sur les ensembles inévitables, et surtout être soigneux dans la programmation des algorithmes.

Cette qualité des programmes de l'équipe de G. McGuire a permis de gagner la compétition engagée avec une équipe chinoise (parvenue au même résultat, mais plus d'un an après). Le calcul des Chinois n'a cependant pas été vain, puisqu'en aboutissant au même résultat, il appuie l'affirmation que 17 est bien le nombre minimum de données d'une grille partielle correcte de sudoku.

La partie finale de la méthode exige l'utilisation d'un programme pour résoudre les sudokus, programme qui doit être capable d'indiquer qu'une grille partielle est incorrecte car elle autorise plusieurs solutions.

Là encore, le partage gratuit de programmes entre chercheurs a été utile. Plutôt que de développer leur propre programme de résolution de sudokus, l'équipe a préféré rechercher le meilleur de ceux déjà développés. Leur choix s'est porté sur un programme *open-source* (donc libre d'utilisation, contrôlable et modifiable) dû à Brian Turner, un ingénieur de *Google* qui développe des logiciels de jeux pour son plaisir. Ce programme rapide a cependant été encore accéléré et peut tester en une seconde 50 000 problèmes à 16 données.

Une fois tous les éléments en place et vérifiés, l'exploration du catalogue à la recherche d'une grille partielle à 16 données a été engagée. Après un an de calcul, la réponse est arrivée : il n'en existe pas !

Définitivement résolu ?

Ce formidable travail collectif et informatique pour réussir à démontrer « Le théorème du sudoku » pose cependant un problème : peut-on faire confiance à ce calcul fondé sur une série de programmes délicats fonctionnant une année entière sur des centaines de processeurs différents, et qui n'a été vérifié qu'une fois depuis par une méthode assez proche ?

Le point qui rend difficile une certitude absolue est dû à la nature du calcul, qui n'est que la recherche infructueuse d'une grille correcte à 16 données. Lorsqu'un ordinateur trouve une solution à un problème et que cette solution est contrôlable par une autre méthode plus rapide que la phase de recherche (comme dans la chasse aux grands nombres premiers), on a peu de raisons de douter de l'exactitude de la solution trouvée. Le premier programme peut comporter des erreurs ou dysfonctionner à cause d'un bug interne ou dans le système d'exploitation, il se peut aussi que le programme oublie une partie du calcul que les programmeurs prévoyaient, etc. Rien de tout cela n'est grave si, à l'issue du calcul, on propose quelque chose de rapidement et indépendamment contrôlable par d'autres programmes ; on se moquera de la phase préliminaire éventuellement très longue et qui n'a plus d'importance pour l'affirmation mathématique.

Ici, la situation est opposée : le programme a recherché quelque chose pendant un an et n'a rien trouvé, et il faut être certain que rien n'a été oublié. Il faut donc être certain que les idées mathématiques sont chacune parfaitement maîtrisées et mises en place, que les programmes ont été écrits sans la moindre erreur, qu'il n'y a pas de bug dans le système d'exploitation de la machine (qui aurait pu faire sauter l'exécution d'une partie des instructions) et qu'à chaque instant le calcul a bien été mené comme il devait l'être sans être perturbé par une panne de disque dur ou par un rayon cosmique. Les craintes sont compréhensibles.

Seule la poursuite du travail sur le sujet, la mise au point de nouvelles méthodes, mathématiques ou techniques, conduisant à d'autres confirmations (si possible plus rapides et avec des programmes indépendants) fournira la certitude attendue. Un assistant de preuve *(voir l'encadré page 78)* ne serait guère utile, car il est à parier qu'il serait beaucoup trop lent (valider les calculs oblige ici à les refaire, même pour l'assistant de preuve). L'ordinateur est utile, mais restons conscients des erreurs possibles dans les preuves informatiques de non-existence, qui ne doivent pas être considérées avec le même œil que les preuves d'existence. ■

L'AUTEUR

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

BIBLIOGRAPHIE

G. McGuire *et al.*, **There is no 16-clue sudoku : Solving the sudoku minimum number of clues problem *via* hitting set enumeration**, *Experimental Mathematics*, vol. 23(2), pp. 190-217, 2014.

J. Cooper et A. Kirkpatrick, **Critical sets for sudoku and general graph colorings**, *Discrete Mathematics*, vol. 315, pp. 112-119, 2014.

J. Rosenhouse et L. Taalman, **Taking Sudoku Seriously : The Math Behind the World's Most Popular Pencil Puzzle**, Oxford University Press, 2012.

H.-H. Lin et I-C. Wu, **An efficient approach to solving the minimum sudoku problem**, *ICGA Journal*, vol. 3, pp. 191-208, 2011.

Références supplémentaires sur le site www.pourlascience.fr

Retrouvez la rubrique Logique & calcul sur www.pourlascience.fr

pls_0447_p076081_delahaye.indd 81 01/12/14 18:06