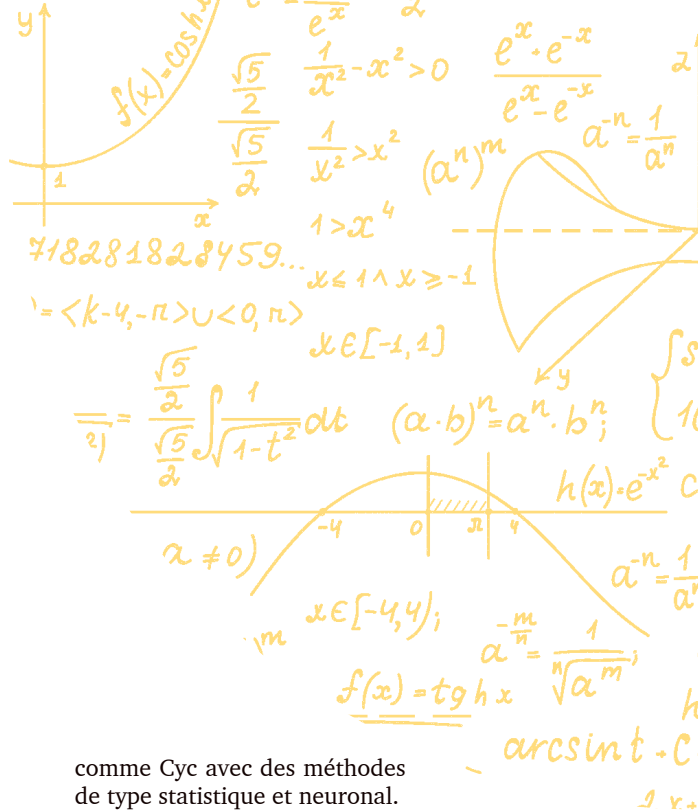


## FAIRE DES CONJECTURES INTÉRESSANTES

Pour concevoir des systèmes menant une recherche mathématique, il faut faire en sorte qu'ils effectuent des tris. Un tel système ne doit retenir que les propriétés et conjectures les plus intéressantes. Le Britannique Simon Colton, qui a développé le système *HR* (voir le texte principal), propose six critères pour repérer les conjectures intéressantes :

- 1) Les conjectures doivent être plausibles. Il faut éliminer celles qui sont fausses parce qu'il existe des exemples simples qui les contredisent.
- 2) Les conjectures doivent être nouvelles. Le système doit disposer des résultats connus sur le sujet étudié et savoir les exploiter pour ne pas proposer les conséquences immédiates de résultats connus.
- 3) Les résultats envisagés doivent être « étonnants », une notion qui échappe aux programmes. Le plus souvent, on se contente d'éliminer les énoncés qui sont des conséquences immédiates d'énoncés connus ou qui sont des cas particuliers de formules logiques toujours vraies, comme  $\text{non}(\text{non-}A) \Leftrightarrow A$ . Une propriété simple qui exige une longue démonstration est étonnante puisqu'on ne peut pas comprendre aisément ce qui la rend vraie. Malheureusement, la notion « exige une longue démonstration » est impossible à programmer en pratique, car, pour l'affirmer d'un énoncé, il faudrait explorer tous les chemins de démonstrations possibles pouvant y conduire.
- 4) Les conjectures doivent concerner de nombreux cas.
- 5) Les conjectures doivent être facilement compréhensibles et, donc, doivent avoir des énoncés assez courts et, si possible, ne faire intervenir que les concepts de base de la théorie étudiée.
- 6) Les conjectures doivent être fécondes et, par exemple, avoir beaucoup de conséquences simples. En mathématiques, il est difficile d'anticiper cette qualité d'une conjecture. Le grand théorème de Fermat n'est considéré comme fécond que parce qu'il a résisté pendant des siècles. Si Fermat l'avait démontré en quelques lignes, personne n'aurait estimé l'énoncé de ce théorème comme intéressant.



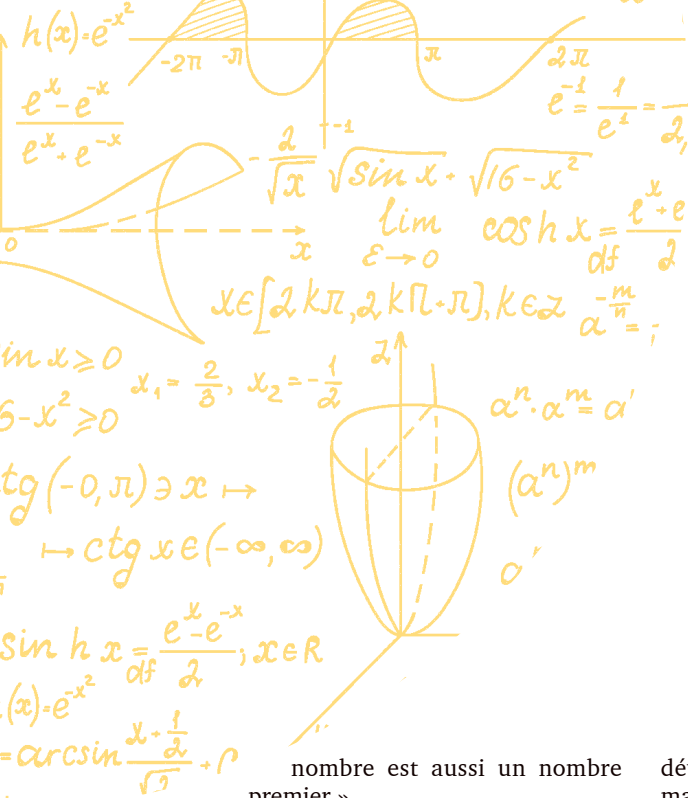
comme Cyc avec des méthodes de type statistique et neuronal.

Quelques années après Eurisko, Simon Colton, de l'université d'Édimbourg, a réussi à faire « imaginer » des notions mathématiques nouvelles à son programme, dénommé *HR* en l'honneur des mathématiciens Godfrey Hardy et Srinivasa Ramanujan, génie ayant découvert une multitude de formules extraordinaires.

Le programme *HR* s'est intéressé à l'arithmétique et a tenté à la fois de créer des concepts nouveaux – des définitions de suites numériques nouvelles –, de formuler des hypothèses à leur sujet et parfois de les démontrer à l'aide de programmes de démonstration automatique qu'il mettait à son service. Le programme disposait de critères pour ne se concentrer que sur des définitions potentiellement intéressantes. Il a pu ainsi identifier 14 suites numériques nouvelles, qui ont été acceptées en 2000 par l'Encyclopédie des suites numériques de Neil Sloane (<http://oeis.org>), où ne figurent que les suites ayant de l'intérêt aux yeux des mathématiciens.

L'une des suites proposées par *HR* est étonnamment simple et a conduit Simon Colton à publier un article dans le *Journal of Integer Sequences* en 1999. Il s'agit de la suite des « nombres refactorisables ». Un entier  $n$  est refactorisable s'il est divisible par le nombre de ses diviseurs. L'entier 9 est par exemple refactorisable, car il a 3 diviseurs (1, 3 et 9) et que justement il est divisible par 3. Plus généralement, si  $p$  est premier,  $p^{p-1}$  est refactorisable.

Notons que le programme de Simon Colton a découvert tout seul cette jolie propriété : « Si la somme des diviseurs d'un entier est un nombre premier, alors le nombre de diviseurs de ce



nombre est aussi un nombre premier.»

Le programme *HR* a aussi inventé des notions concernant les groupes, les anneaux finis et les graphes. Il dispose de diverses mesures évaluant l'intérêt d'un concept, se concentre sur le plus prometteur et cherche alors à énoncer des propriétés à son sujet pour lesquelles il tente de trouver des démonstrations ou des contre-exemples. Cette façon de procéder est proche de ce que font souvent les chercheurs en mathématiques, mais partiellement, car un chercheur prend de la distance vis-à-vis de son sujet et tente d'établir des liens entre concepts appartenant à des champs éloignés.

## UN VRAI SUCCÈS SUR LES GRAPHS

Le principe général de *HR* est particulièrement efficace quand il s'agit de structures finies, car le programme peut disposer d'une liste variée d'exemples et les utiliser pour tester si les propriétés qu'il envisage sont vérifiées par chaque élément de sa base d'exemples. Si c'est le cas, la propriété devient plausible et le système l'envisage comme conjecture. Si l'un des exemples contredit la propriété examinée, alors il est certain qu'elle est fautive, et il l'abandonne.

La théorie des graphes se prête bien à cette approche, et c'est pourquoi les méthodes automatiques de recherche de conjectures en théorie des graphes ont donné les premiers exemples incontestables de découvertes mathématiques faites par l'ordinateur.

Le programme *Graffiti* est la première grande réussite de cette approche. Il a été

développé et utilisé de 1984 à 2002 par le mathématicien polonais Siemion Fajtlowicz.

*Graffiti* s'intéresse particulièrement aux invariants des graphes. Ce sont, par exemple, le diamètre du graphe (la distance – en nombre d'arêtes – maximale séparant deux nœuds), le nombre chromatique d'un graphe (le nombre minimal de couleurs qu'il faut pour colorer les nœuds du graphe en évitant que deux nœuds voisins soient de même couleur), le degré maximal d'un graphe (nombre maximal de voisins d'un nœud), etc.

En menant des recherches automatisées de relations entre ces invariants, *Graffiti* a proposé de nombreuses conjectures intéressantes, dont certaines ont été démontrées par la suite ou invalidées par des chercheurs de la discipline, donnant lieu à plusieurs publications scientifiques et thèses universitaires.

Siemion Fajtlowicz a d'ailleurs été amené à travailler avec les meilleurs spécialistes de la théorie des graphes comme Paul Erdős, Bela Bollobas ou Paul Spencer. Erdős et ses coauteurs, dans un article de 1988 où ils démontrent l'une des conjectures du programme *Graffiti*, reconnaissent son apport. La conjecture en question affirme que  $dm_G \leq r_G$ , où  $dm_G$  est la distance moyenne entre deux nœuds du graphe  $G$ , et  $r_G$  est la somme des inverses du nombre de voisins des nœuds de  $G$ . Aujourd'hui encore, en théorie des graphes et dans certains autres domaines, l'utilisation de générateurs de conjectures aide les chercheurs.

Un événement remarquable s'est produit en 1995 dans le domaine de l'aide informatique à la recherche mathématique. Simon Plouffe a programmé sa machine pour qu'elle explore des formules d'un certain type donnant le nombre  $\pi$