

du problème. Par exemple, un célèbre algorithme de tri, le tri à bulles, fait remonter progressivement les plus grands éléments d'une liste par une succession de comparaisons deux à deux. Ce processus prend un temps qui augmente comme le carré du nombre d'éléments de la liste. Même si l'on doublait la longueur de la liste, l'algorithme mettrait dans le pire des cas quatre fois plus longtemps (ce « pire cas » se produit quand la liste est dans l'ordre inverse).

Au-dessus du niveau NP de complexité figurent les problèmes extrêmement difficiles sur le plan calculatoire. Il existe même des problèmes pour lesquels notre modèle de calcul standard, celui que tous les ordinateurs implémentent,

■ L'AUTEUR



Toby WALSH est chercheur au NICTA (National Information Communications Technology Australia) et professeur d'intelligence artificielle à l'université de Nouvelle-Galles du Sud, en Australie.

Article publié avec l'aimable autorisation de la revue *American Scientist*.

est inadéquat. Pour de tels problèmes, il n'existe pas de programme informatique dont on ait l'assurance qu'il va s'arrêter de tourner et donner un résultat. Ces exemples s'inscrivent dans la classe des problèmes dits indécidables. Cette classe inclut des questions telles que décider si un programme informatique va s'arrêter plutôt que de continuer à tourner indéfiniment en boucle, ce que les informaticiens nomment le « problème de l'arrêt ». Alan Turing, pionnier de l'informatique, a prouvé que le problème de l'arrêt est indécidable : il n'existe aucun programme informatique qui puisse, pour tout programme, indiquer s'il s'arrêtera ou non.

La classe NP se situe juste à la frontière entre « facile » et « difficile ». On y trouve, notamment, de nombreux problèmes représentant un défi, tels que l'organisation des tournées de livraison de colis, du tableau de service pour un hôpital ou de l'emploi du temps d'un lycée. Il s'avère que gagner à Candy Crush s'inscrit dans ce sous-ensemble de NP. L'un quelconque de ces problèmes peut être réduit à n'importe quel autre d'entre eux. En ce sens, ils sont tous d'égale difficulté.

Un calcul plus long que l'âge de l'Univers ?

Malheureusement, les meilleurs programmes informatiques dont nous disposons pour les problèmes NP ont un temps d'exécution qui augmente énormément avec la taille des données du problème. Sur mon ordinateur de bureau, j'ai un programme qui met quelques heures à trouver les tournées optimales pour $N = 10$ camionnettes de livraison et à démontrer que cette solution est la meilleure possible. Mais pour $N = 100$ camionnettes, le même programme devrait tourner pendant plus de temps que l'âge de l'Univers. En fait, le temps d'exécution de mon programme est une fonction exponentielle de la taille N du problème, et les exponentielles ont une croissance extrêmement rapide.

Bien que les informaticiens soient généralement d'accord avec moi pour dire que les problèmes NP sont à la limite entre facile et difficile, il n'y a aucun moyen de savoir avec certitude, pour un problème spécifique, de quel côté il se situe. Les meilleurs programmes informatiques actuellement à notre disposition mettent un temps exponentiellement long à résoudre



UNE BOMBE DE COULEUR DÉTRUIT TOUS LES BONBONS VIOLETS de la grille sur cette capture d'écran d'une partie de Candy Crush Saga. L'une des raisons pour lesquelles le jeu est tellement prenant est qu'il est en fait vraiment difficile, au sens mathématique du terme.



les problèmes de la classe NP. Mais nous ignorons s'il existe un algorithme (à découvrir) qui saurait résoudre les problèmes NP efficacement, en un temps polynomial. Les mathématiciens expriment cette question en la résumant : « P est-il égal à NP ? » En fait, c'est l'un des problèmes ouverts les plus importants des mathématiques d'aujourd'hui. L'institut Clay de mathématiques a même promis, depuis 2000, une récompense d'un million de dollars à quiconque le résoudra.

Dans le dernier sondage interrogeant les informaticiens pour savoir s'ils pensaient que $P = NP$ est vrai, 83 % déclaraient que non. Autrement dit, ils pensent qu'il n'existe pas d'algorithme efficace pour résoudre les problèmes NP et qu'il n'y en aura jamais. On a également demandé à des informaticiens par quel vocable il convenait de désigner les problèmes qui sont aussi difficiles à résoudre que ceux de NP, qu'ils soient ou non dans cette classe. Le mot qui a finalement remporté les suffrages est le plutôt prosaïque « NP-difficile ». Mais la consultation a également mis en évidence des suggestions plus hautes en couleur telles que NP-impractical (NP-pas pratique), NP-tricky (NP-épineux) et NP-hard-ass (NP-dur à cuire)...

L'idée de la réduction de problème est centrale à la question « $P = NP$? ». Si nous trouvions un algorithme capable de résoudre efficacement un problème NP-difficile, alors nous pourrions aussi résoudre efficacement tous les autres problèmes de la

ELEMENTARY, une série télévisée formant une adaptation moderne de Sherlock Holmes, illustre comment, bien au-delà de Candy Crush, les problèmes NP se sont fait une place dans la culture populaire. Dans l'épisode dont cette image est extraite, deux mathématiciens ont caché à leurs rivaux leurs recherches sur la question « $P = NP$? » en faisant leurs calculs avec des marqueurs ultraviolets, comme le découvre Holmes (ci-dessus) après leur assassinat lié à leurs résultats pionniers.

classe NP. Le monde serait assez différent si ce résultat était un jour atteint.

Du côté des avantages, nous pourrions mener nos vies en gérant plus efficacement notre temps, tout serait magnifiquement optimisé, depuis les tournées de livraison jusqu'aux horaires de vols en passant par les emplois du temps des personnels hospitaliers (et l'on gagnerait régulièrement en jouant à Candy Crush).

**La complexité
calculatoire est parfois
un bienfait**

En revanche, il est important que certaines tâches restent difficiles, par exemple le déchiffrement des messages cryptés, si nous voulons que nos mots de passe et nos comptes bancaires restent sûrs. La complexité calculatoire peut être un bienfait autant qu'un fléau. Nous voulons rendre difficile (d'une difficulté prouvable) le décryptage des messages par des pirates informatiques. Et tout autant, nous devons être capables de crypter ces messages facilement. Cet exemple rappelle la définition de la classe NP-difficile : des problèmes dont la solution est facile à vérifier, mais difficile à trouver.

Pour montrer que Candy Crush est un problème mathématiquement difficile, nous pouvions procéder en partant de n'importe quel problème NP-difficile et en le réduisant à Candy Crush. Pour nous simplifier la

■ BIBLIOGRAPHIE

T. Walsh, **Candy Crush is NP-hard**, prépublication arXiv du 11 mars 2014 (<http://arxiv.org/abs/1403.1911>).

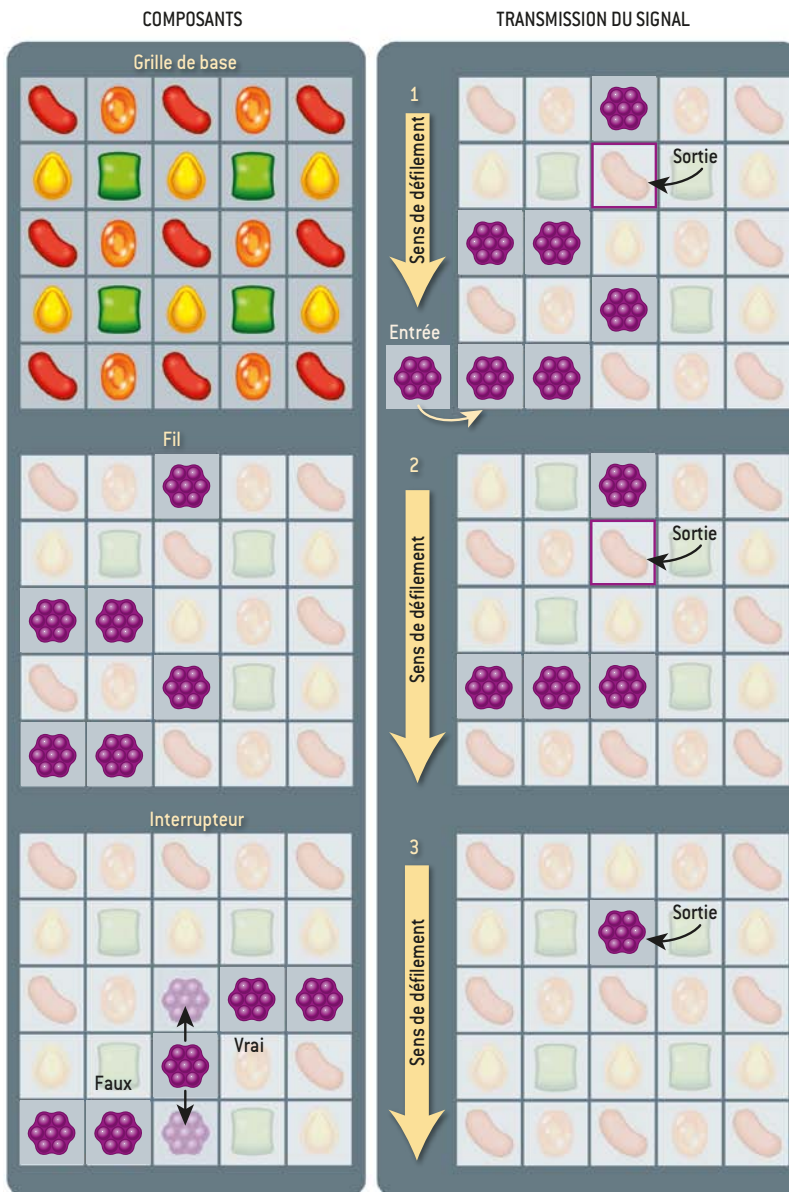
G. J. Woeginger, **The P-versus-NP page** (www.win.tue.nl/~ffgwoegi/P-versus-NP.htm).

J.-P. Delahaye, **Complexités**, Belin/Pour la Science, 2006.

IMITER DES CIRCUITS ÉLECTRIQUES AVEC DES BONBONS

Afin de prouver que Candy Crush appartient à la classe des problèmes NP-difficiles, Toby Walsh a traduit ce jeu en un problème de logique appartenant à cette même classe, en concevant un circuit électrique constitué de bonbons. Le principe consiste à partir d'une grille de base (à gauche en haut) appropriée, dans laquelle on place des bonbons violets en fonction du composant électrique que l'on cherche à imiter.

On peut former ainsi un « fil » (à gauche au milieu) qui transmet un signal, un « interrupteur » (à gauche en bas) qui détermine quel fil sera utilisé, etc. Dans le « fil », par exemple, la propagation du signal (dessins de droite) commence par le placement d'un bonbon violet sur une case « Entrée » (1). Cela crée un alignement de trois bonbons identiques, qui est effacé. Les bonbons situés au-dessus se décalent vers le bas, créant un nouvel alignement de trois bonbons (2). Quand cet alignement est effacé, un bonbon violet tombe dans la case de sortie : le signal a été transmis (3).



tâche, mes collègues et moi sommes partis de l'ancêtre de tous les problèmes NP-difficiles, à savoir trouver une solution à une formule logique. C'est ce qu'on appelle le « problème de la satisfaisabilité ».

Vous avez forcément résolu un tel problème si vous vous êtes un jour attaqué à un casse-tête logique. Vous devez décider quelles propositions rendre vraies, et lesquelles rendre fausses, afin de satisfaire un ensemble de formules logiques. Par exemple, sachant que « l'Anglais vit dans la maison rouge », « l'Espagnol possède le chien » et « le Norvégien habite à côté de la maison bleue », la proposition « l'Espagnol possède le zèbre » doit-elle être déclarée vraie ou fausse ?

Traduire Candy Crush en un problème de logique

Pour réduire un casse-tête logique à un problème de Candy Crush, nous exploitons le lien étroit qui existe entre la logique et les circuits électriques. On peut en effet représenter n'importe quelle formule logique par un circuit électrique. Les ordinateurs ne sont, après tout, qu'un grand ensemble de portes logiques (ET, OU, NON, etc.) connectées par des conducteurs électriques. Par conséquent, tout ce que nous avons besoin de faire est de montrer que l'on pourrait construire un circuit électrique dans un jeu de Candy Crush.

Tout d'abord, nous avons besoin d'une grille sur laquelle le circuit va être construit. La grille doit être un motif neutre de bonbons où l'ordre des types de bonbons les uns par rapport aux autres ne change jamais (voir l'encadré ci-contre). La configuration choisie ressemble à des feux de circulation : dans les colonnes paires, nous alternons les bonbons haricots rouges et les bonbons au citron jaunes, alors que dans les colonnes impaires nous alternons les pastilles orange et les gommes vertes. De cette façon, même en déplaçant des colonnes vers le haut ou vers le bas, nous ne créerons jamais une chaîne de trois bonbons identiques.

Dans ce cadre, nous insérons les composants électriques, qui sont constitués de bonbons violets à la mûre. Les bonbons violets décalent les autres bonbons, ils ne les effacent pas. En groupant des bonbons violets, on crée des « fils » qui acheminent le signal dans le circuit (voir l'encadré ci-contre), ou des configurations plus complexes selon les besoins.

Barbara Aulicino