

Par exemple, lorsqu'on place un bonbon violet sur l'entrée du « fil » en bas à gauche de la grille de départ représentée dans l'encadré, on crée une chaîne de trois bonbons violets. Cette chaîne est effacée, conformément aux principes de base du jeu, ce qui déplace vers le bas les bonbons des deux colonnes concernées. Et ainsi de suite, le signal se propage et est transmis : à la fin, un bonbon violet apparaît dans la case de sortie.

Nous avons également besoin d'interrupteurs que l'utilisateur peut actionner pour décider quels fils sont actifs. Ces interrupteurs représentent le choix des valeurs de vérité (vrai ou faux) d'une proposition logique. L'utilisateur déplace un bonbon violet soit vers le haut, soit vers le bas, ce qui va envoyer un signal soit vers la gauche, soit vers la droite (voir l'encadré).

Enfin, on peut construire des portes logiques telles que ET, OU et NON à partir d'autres bonbons violets en utilisant ces composants de base. Il nous reste alors simplement à connecter les interrupteurs à ces portes logiques avec des fils suffisamment longs pour obtenir un circuit électrique qui simule notre formule logique. Le circuit électrique a alors une sortie qui représente la valeur de vérité de la formule logique.

En termes de circuits logiques électriques, jouer à Candy Crush consiste à décider judicieusement sur quels interrupteurs agir de façon que les portes logiques s'activent convenablement et que la sortie soit « vrai ». De cette manière, le problème consistant à satisfaire une formule logique est réduit à résoudre un problème dans Candy Crush. Et puisque satisfaire une formule logique est un problème difficile, il en va de même de la résolution d'une grille de Candy Crush.

On peut également montrer l'inverse. En d'autres termes, on peut réduire un problème Candy Crush à celui de la satisfaction d'une formule logique. Il faut simplement écrire une séquence de formules qui représentent le jeu sur une grille de Candy Crush. Or pour l'essentiel, une description logique de Candy Crush se trouve dans n'importe quel programme qui y joue. Par conséquent, Candy Crush

n'est pas plus difficile que l'un ou l'autre des problèmes NP-difficiles, et le jeu est exactement aussi difficile qu'il est difficile de résoudre tous les autres problèmes de la classe NP.

Si nous avions une manière efficace de jouer à Candy Crush, nous aurions un moyen efficace (et prouvable) d'organiser les tournées de livraison, de faire les tableaux de service des hôpitaux et les emplois du temps des lycées. Inversement, si nous avions un moyen efficace de préparer les tournées, faire les tableaux de service et les emplois du temps, alors nous disposerions d'un moyen efficace de jouer à Candy Crush. C'est toute la puissance de la réduction de problème.

Pourquoi ne pas profiter de Candy Crush pour que les joueurs effectuent des calculs longs et utiles ?

La prochaine fois que vous calez pour résoudre une grille de Candy Crush avec le nombre de coups impartis, vous pourrez vous consoler en sachant que c'était un problème mathématiquement difficile à résoudre. De fait, cette caractéristique participe sans doute à l'attrait du jeu et à son caractère addictif. S'il était aussi facile à résoudre que le morpion, par exemple, il ne serait sans doute pas aussi captivant.

Au cœur de tout cela, il y a la belle notion de réduction de problème, qui a permis aux informaticiens de simplifier le labyrinthe des différents problèmes calculatoires en un nombre plus petit de classes fondamentales telles que P et NP, que les informaticiens appellent le zoo

de la complexité. Actuellement, ce zoo contient 500 classes de problèmes.

Dans l'hypothèse improbable où l'on démontrerait que $P = NP$, le nombre de classes distinctes du zoo de la complexité chuterait brutalement. De nombreuses classes se révéleraient être identiques. En revanche, si $P \neq NP$, comme le croient la plupart des informaticiens, alors le zoo contient effectivement de nombreuses classes distinctes de problèmes. En fait, le zoo continue à s'élargir. Les zoologistes de la complexité ont récemment introduit de nouvelles classes pour décrire la complexité des problèmes que pourraient résoudre de futurs ordinateurs quantiques.

L'idée de réduction de problème offre une possibilité intéressante pour les mordus de Candy Crush. Peut-être pourrait-on tirer parti des millions d'heures passées par l'humanité à résoudre des grilles de Candy Crush ? En exploitant l'idée de réduction de problème, nous pourrions dissimuler des problèmes calculatoires utiles dans ces casse-tête.

D'autres problèmes calculatoires bénéficient de telles interactions : chaque fois que vous prouvez à un site web que vous êtes une personne et non un robot en résolvant un Captcha (acronyme de *Completely Automated Public Turing test to tell Computers and Humans Apart*, « test de Turing complètement automatisé pour distinguer les ordinateurs des humains » – un tel test vous demande de taper une séquence de lettres et chiffres apparaissant déformés à l'écran), la réponse aide Google à numériser les anciens ouvrages et journaux. Peut-être devrions-nous atteler les casse-tête de Candy Crush à des tâches non moins importantes !

Nos études de Candy Crush nous ont inspiré un profond respect pour ce passe-temps apparemment anodin. Il offre un éclairage original sur l'une des questions ouvertes les plus importantes des mathématiques d'aujourd'hui, et les implications de cette question s'étendent à de nombreuses applications telles que les algorithmes de chiffrement utilisés pour garantir la sécurité des comptes bancaires. Vous pourrez ainsi expliquer ces enjeux supérieurs à votre patron la prochaine fois qu'il vous surprend à essayer de vous hisser un niveau plus haut ! ■