PROBLEMES SOLUBLES ET PROBLEMES INSOLUBLES

By Alan Mathison Turing

Les candidats doivent faire un commentaire du texte faisant appel à leur culture philosophique et historique personnelles. Des illustrations tirées de leur propre discipline, ainsi que de disciplines voisines sont les bienvenues.

« Si l'on donne un *puzzle* à résoudre et qu'il s'avère difficile, on demandera à son détenteur s'il peut être soluble. Une telle question devrait trouver une réponse tout à fait nette, du genre oui/non, dès lors que les règles décrivant ce que vous êtes autorisés à faire sont parfaitement claires. Bien entendu, le détenteur du *puzzle* peut ne pas connaître la réponse. On pourra également demander : “Comment peut-on dire si un *puzzle* est soluble ?”, mais on ne peut répondre à cette question de manière aussi simple. Le fait est qu'il *n*'y a *pas* de méthode systématique pour tester des puzzles et voir s'ils sont résolubles ou non. Si par là on entend simplement que personne n'a encore jamais trouvé un test qui pourrait s'appliquer à n'importe quel *puzzle*, il n'y aurait rien de bien remarquable dans un tel énoncé. C'eût été un véritable exploit que d'inventer un tel test, aussi pouvons-nous difficilement être surpris que cela n'ait jamais été réalisé. Mais ce n'est pas simplement que le test n'a jamais été trouvé. Il a été prouvé qu'un tel test ne pourra jamais l'être.

[…] Les *puzzles* pour lesquels il est demandé de dissocier des corps rigides sont en un sens comme le “*puzzle*” qui consiste à tenter de démêler un enchevêtrement ou, plus généralement, à tenter de transformer un nœud en un autre nœud sans couper la corde. La différence c'est qu'on est autorisé à déformer la corde, mais pas le maillage filaire formant les corps rigides. Dans les deux cas, si l'on veut traiter le problème sérieusement et de manière systématique, il faut remplacer le *puzzle* physique par un équivalent mathématique. Le *puzzle* du nœud s'y prête assez bien. Un nœud n'est qu'une courbe fermée en trois dimensions qui ne se recoupe nulle part ; mais, pour le propos qui nous intéresse ici, tout nœud peut être déterminé assez précisément comme une série de segments orientés suivant les trois axes de coordonnées. Ainsi, par exemple, dans le système de coordonnées habituel (x, y, z), le nœud de trèfle (Figure 1*a*) peut être considéré comme formé d'un certain nombre de segments joignant les points donnés, tels que (1, 1, 1), (4, 1, 1), (4, 2, 1), (4, 2, – 1), (2, 2, – 1), (2, 2, 2), (2, 0, 2), (3, 0, 2), (3, 0, 0), (3, 3, 0), (1, 3, 0), (1, 3, 1) et revenant au point de départ (1, 1, 1) avec le douzième segment. Cette représentation du nœud est montrée en perspective sur la Figure 1*b*.

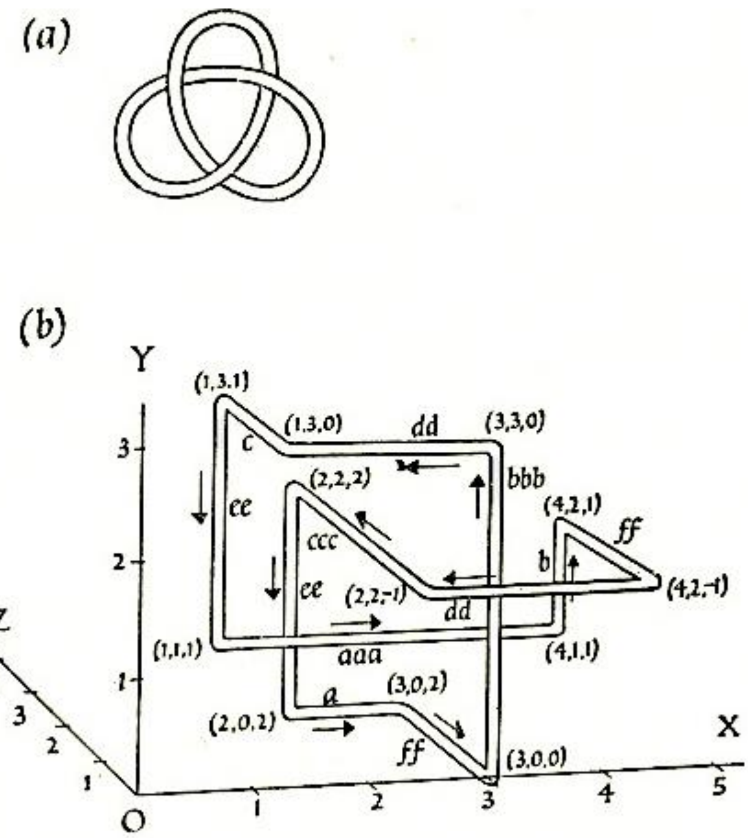

Figure 1. (*a*) Le nœud de trèfle (*b*) une représentation possible de ce nœud comme ensemble de segments joignant des points.

[…] Avant tout nous pouvons supposer que le *puzzle* est d'une façon ou d'une autre réduit à une forme mathématique comme dans le cas des nœuds. La position du *puzzle* peut alors être décrite par des séquences de symboles sur une ligne. Il n'y a généralement pas de difficulté majeure pour réduire d'autres arrangements de symboles à cette forme. La question qui demeure est la suivante : "Quel genre de règles est-on autorisé à appliquer pour réarranger les symboles ou les jetons ?" Pour être en mesure de répondre à cette question, il faut se demander quels genres de processus se reproduisent toujours avec de telles règles et, pour être en mesure d'en réduire le nombre, il faut les décomposer en des processus plus simples. Compter, copier, comparer et substituer sont typiques de tels processus. Quand on réalise de tels processus, surtout s'il y a de nombreux symboles impliqués et si l'on désire éviter de transporter trop d'information dans sa tête, il est nécessaire soit de conserver un certain nombre de notes par ailleurs, soit d'utiliser autant d'objets marqueurs qu'il y a de pièces au *puzzle*. Par exemple, si on est amené à copier une ligne de jetons impliqués dans le *puzzle*, c'est tout aussi bien d'avoir un marqueur qui sépare les pièces copiées de celles qui ne l'ont pas été, et un autre qui marque la limite de la portion à copier. Or, il n'y a aucune raison pour que les règles du *puzzle* lui-même ne puissent s'exprimer de manière à tenir compte de ces marqueurs. Si l'on exprime les règles de cette manière-là, elles peuvent alors se réduire à des substitutions. Ce qui revient à dire que la *forme normale pour des* puzzles *est du type substitution de* puzzle.

[…] Il est clair que la difficulté pour trouver des procédures de décision concernant certains types de *puzzle* [il faut noter qu'un problème de décision ne se pose que quand on a une infinité de questions à poser] tient dans le fait de devoir établir que le *puzzle* est insoluble dans les cas où il *est* insoluble. Ce qui requiert un certain type d'argument mathématique et suggère que nous puissions tenter d'exprimer l'énoncé que le *puzzle* se met sous forme mathématique, puis entreprendre de le prouver par un processus systématique quelconque. Il n'y a pas de difficulté particulière pour la première partie de ce projet : l'expression mathématique de l'énoncé concernant le *puzzle*. Mais la seconde moitié du projet est vouée à l'échec à cause d'un fameux théorème de Gödel qui montre qu'aucune méthode systématique de preuve des théorèmes mathématiques n'est suffisamment complète pour pouvoir régler toute question mathématique par oui ou par non ».

Alan Mathison TURING, *Solvable and Unsolvable Problems,* Science News 31, 1954, pp. 1, 3-4 [*traduction française* de Charles Alunni].

Alan Mathison TURING (23 juin 1912–7 juin 1954) était un mathématicien britannique auteur de l'article fondateur de la science informatique qui allait donner le coup d'envoi à la création de l'ordinateur programmable. Il y présente sa *machine de Turing*, le premier calculateur universel programmable, et invente les concepts de programmation et de programme. Il est également à l'origine de la formalisation des concepts d'*algorithme* et de *calculabilité* qui ont profondément marqué cette discipline. Son modèle a contribué à établir définitivement la *thèse Church-Turing* qui donne une définition mathématique au concept intuitif de fonction calculable. Durant la Seconde Guerre mondiale, il a dirigé les recherches sur les codes secrets générés par la machine *Enigma* utilisée par les nazis. Après la guerre, il a travaillé sur un des tout premiers ordinateurs, puis a contribué de manière provocatrice au débat déjà houleux à cette époque sur la capacité des machines à penser en établissant le *test de Turing*. Vers la fin de sa vie, il s'est intéressé à des modèles de *morphogenèse du vivant* conduisant à ce que l'on appelle les *structures de Turing*.