aléatoire. Leur profondeur logique de Bennett est donc minimale, comme nous venons de le voir: un objet aléatoire n'est pas structuré et a donc une profondeur logique minimale, comparable à celle d'une suite de même longueur composée uniquement de 0.

Le *tout* (composé des deux images A et B) n'est pourtant pas aléatoire, car les deux images sont intimement corrélées. Pour s'en rendre compte, on applique l'opération logique «ou exclusif», notée xor, entre A et B, ce qui donne une image C: quand les deux pixels homologues de A et B sont identiques, on met un 0 pour le pixel homologue de l'image C, sinon on met un 1: 0 xor 0 = 0, 1 xor 1 = 0, 0 xor 1 = 1 et 1 xor 0 = 1.

Les images A et B n'ont pas été produites n'importe comment, et le xor entre A et B fait apparaître un beau dessin géométrique qui est la représentation d'un pavage par des triangles du plan hyperbolique. Le même procédé, que nous allons expliquer, sert de base à ce que l'on dénomme la «cryptographie visuelle», il permet d'obtenir n'importe quelle image aussi structurée qu'on le souhaite en partant de deux objets parfaitement non structurés (mais corrélés).

Le procédé donnant A et B est assez simple. On part d'une image aléatoire A: 50 % de pixels blancs et autant de pixels noirs, tirés au hasard. On choisit une image C quelconque, éventuellement très structurée. L'image B est alors construite en calculant A xor C. En faisant le calcul A xor B, on retombera sur C, car:

A xor B = A xor (A xor C)
= (A xor A) xor C = [0] xor C = C.

La dernière égalité mentionne l'image notée [0], qui désigne celle composée uniquement de pixels blancs. Le fait que [0] xor C = C résulte de ce que 0 xor 0 = 0 et 0 xor 1 = 1.

L'image B est parfaitement aléatoire elle aussi. Celui qui ne dispose que de A seul, ou que de B seul, ne peut strictement rien en tirer. Ni A ni B n'ont le moindre contenu en organisation.

Il résulte de tout cela que le programme le plus court qui donnera le *tout* sera le programme le plus court de A associé avec le programme le plus court de C, suivi d'un calcul du ou exclusif entre A et C, ou sera quelque chose de très proche. Puisque C est structuré de façon non triviale, ce programme minimal pour le *tout* aura un temps de calcul plus long que la somme des temps de calcul des programmes minimaux pour A et minimaux pour B (qui sont des programmes très rapides, puisqu'il n'y a aucune structure dans A, et aucune structure dans B). La profondeur logique du *tout* est donc supérieure à la somme de la profondeur logique de A et de la profondeur logique de B.

On a ainsi ce qu'on voulait: un cas où la maxime aristotélicienne est démontrable. L'énoncé qui permet de prouver l'inégalité est un théorème: «Quelle que soit la profondeur logique d'un objet numérique C de taille $n$, on peut construire deux objets A et B, chacun de taille $n$, de façon que A et B soient chacun de profondeur minimale (parmi les objets de taille $n$), et tels que le *tout* constitué de A et de B soit de profondeur équivalente à celle de C et donc supérieure à la somme de la profondeur de A et de la profondeur de B.»

## LE CAS DES SYSTÈMES COMPLEXES

Il est probable que ceux qui évoquent ce *tout* supérieur à la somme de ses *parties* ont en tête des situations où c'est bien l'organisation (ou encore «la richesse en structures», «la valeur fonctionnelle», «le contenu en calcul») qui sert à mesurer la valeur du *tout* et celle de ses *parties*. L'idée exprimée par la maxime est souvent fausse; elle intéresse d'ailleurs parce qu'on la perçoit comme paradoxale, mais il y a des cas où le paradoxe devient vrai quand ce qui mesure la valeur du tout est vraiment lié à une richesse en organisation. Ces cas font l'intérêt de la maxime.

Croire à la maxime et en faire un pilier philosophique des réflexions sur la complexité sans même chercher à savoir de quoi elle parle, ni si cela peut se mathématiser, est une attitude ridicule. En effet, le plus souvent, c'est l'inégalité inverse qu'on démontre, même quand on envisage la complexité des algorithmes ou la complexité de Kolmogorov. Disposer d'un cas précis où la maxime devient vraie est éclairant, et il faut sans doute y voir une preuve nouvelle de l'intérêt de la définition de Bennett: la complexité structurelle d'un objet fini F se mesure par «le temps de calcul de son programme le plus court», ou, dans la version plus tolérante, par «le temps de calcul des programmes courts qui produisent F».

Il existe peut-être d'autres procédés formels non illusoires donnant un sens à la maxime (l'encadré 3 en présente un), mais celui qui s'appuie sur la profondeur logique de Bennett appliquée à l'association de deux objets corrélés est probablement central, du fait de sa place au sein de la théorie algorithmique de l'information. Dans les systèmes complexes, comme le sont les organismes vivants ou les écosystèmes, les interdépendances font qu'on est assez souvent dans une situation semblable à celle des images A, B et C. Ce qui est apparu dans un premier temps comme une exception y devient la règle. La complexité du *tout* mesurée par la profondeur logique de Bennett est donc, dans de telles structures, supérieure à la somme des complexités des *parties*. ■

## BIBLIOGRAPHIE


M. Li et Paul Vitányi, **An Introduction to Kolmogorov Complexity and its Applications,** Springer, 2014.

J.-P. Delahaye, **Qu'est-ce qu'un objet complexe ?**, *Pour la Science,* pp. 78-83, mai 2013.

J.-P. Delahaye, **La cryptographie visuelle,** *Pour la Science,* pp. 86-91, juin 2012.

C. Bennett, **Logical depth and physical complexity,** dans *The Universal Turing Machine : A Half-Century Survey*, pp. 207-235, Oxford University Press, 1988.

Aristote, **Physique - Livre VI,** http://remacle.org/bloodwolf/philosophes/Aristote/phys6.htm