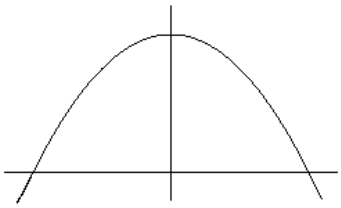
Gilles Dowek

Les démonstrations et les algorithmes

Introduction à la logique et à la calculabilité

L'auteur tient à remercier René Cori, René David, Maribel Fernández, Jean-Baptiste Joinet, Claude Kirchner, Jean-Louis Krivine, Daniel Lascar, Stéphane Lengrand, Michel Parigot, Laurence Rideau et Paul Rozière.

Introduction



Plusieurs méthodes permettent de déterminer l'aire de ce segment de parabole. Une première consiste à le découper en une infinité de petits triangles, puis à déterminer l'aire de chacun d'eux par une démonstration et à faire la somme des aires obtenues. C'est *grosso modo* la méthode qu'Archimède a employée pour démontrer que cette aire était égale à $4/3$. Depuis le XVII$^{\text{e}}$ siècle, toutefois, on peut utiliser une autre méthode, qui donne le même résultat, et qui consiste à déterminer la valeur de l'intégrale $\int_{-1}^{1}(1-x^2)dx$. Intégrer cette fonction polynomiale ne demande pas de construire une démonstration, mais simplement d'appliquer un algorithme.

Construire une démonstration, appliquer un algorithme, ces deux types de méthodes coexistent depuis longtemps au sein des mathématiques, mais les méthodes algorithmiques ont connu un nouvel essor depuis l'apparition des ordinateurs, qui permettent de les utiliser à une toute autre échelle que par le passé.

La coexistence de ces deux méthodes de résolution des problèmes mène à s'interroger sur leurs relations. Jusqu'à quel point peut-on remplacer la construction d'une démonstration par l'application d'un algorithme ? Ce livre

est consacré à un ensemble de résultats, tant négatifs que positifs, qui répondent partiellement à cette question.

Pour parvenir à ces résultats, nous devrons commencer par définir précisément ces notions de démonstration, dans la première partie, et d'algorithme, dans la deuxième. Définir la notion de démonstration permettra de comprendre comment démontrer des théorèmes d'indépendance, qui affirment qu'il n'existe pas de démonstration pour résoudre certains problèmes. Définir la notion d'algorithme permettra de comprendre comment démontrer des théorèmes d'indécidabilité, qui affirment qu'il n'existe pas d'algorithme pour résoudre certains problèmes. Cela nous permettra aussi de comprendre que les algorithmes peuvent se présenter sous de nombreuses formes : ensembles de règles de réécriture, termes du lambda-calcul, machines de Turing, mais que cette diversité masque une profonde unité : l'idée qu'un calcul est une suite de petits pas.

La troisième partie aborde enfin les liens entre les notions de démonstration et d'algorithme. Le résultat central de cette partie est le théorème de Church, qui montre que la démontrabilité dans la logique des prédicats est un problème indécidable, et dont un corollaire est le célèbre théorème de Gödel. Ce résultat négatif sera cependant nuancé par deux résultats positifs. Tout d'abord, s'il est indécidable, ce problème est semi-décidable, ce qui nous mènera à développer des algorithmes de recherche de démonstrations. Ensuite, ajouter des axiomes à la logique des prédicats permet, dans certains cas, de rendre la démontrabilité décidable, ce qui nous mènera à développer des algorithmes de décision pour des théories particulières.

Un dernier chapitre nous montrera un lien différent entre les démonstrations et les algorithmes : certaines démonstrations, que l'on appelle *constructives*, peuvent être utilisées comme des algorithmes.

Au fil des pages, se révélera donc la richesse des liens entre ces notions de démonstration et d'algorithme, mais aussi la complexité qui se cache derrière l'apparente évidence de la notion de vérité.