Calculer avec des algorithmes, calculer avec des machines : un problème philosophique

Résumé long (997 mots)

La thèse de Church, selon laquelle toute fonction calculable est calculable par une machine de Turing, est couramment tenue par la communauté informatique comme la juste expression des limites de la calculabilité. Dans des écrits récents, de nombreux auteurs, tels que B.J. Copeland, W. Sieg, J. Barrow, ou Gilles Dowek, ont néanmoins insisté sur la nécessité de distinguer deux formes de la thèse de Church. La première forme, ou forme *informatique,* affirme que toute fonction calculable *par un algorithme* est calculable par une machine de Turing. La seconde, ou forme physique, affirme que toute fonction calculable *par une machine* est calculable par une machine de Turing. Or, si la première forme est relativement consensuelle, la seconde est plus difficile à définir avec précision, et la question de sa possible démonstration au sein des théories physiques fondamentales est encore ouverte. Il n'est donc pas à l'heure actuelle établi, si les limites du calcul fixées par ces deux propositions coïncident, et la question suivante n'est pas scientifiquement tranchée : *l'ensemble des fonctions calculables par une machine est-il identique à l'ensemble des fonctions calculables par un algorithme ?*

Le présent article vise à dégager l'intérêt philosophique de cette question, et en particulier de la thèse suivante, nommée thèse A : *toute fonction calculable par une machine est calculable par un algorithme.* Même si la notion d' « algorithme » est notoirement difficile à capturer par une définition explicite, il est néanmoins possible d'énoncer certaines conditions nécessaires, qui permettent de faire de la thèse A une proposition philosophique bien définie. Même si on les considère comme des objets abstraits, les algorithmes sont descriptibles par un texte formel fini, nommé *programme* : on dira que les algorithmes sont symboliquement descriptibles. En outre, l'exécution d'un algorithme peut également être décrite par un texte formel, qui explicite l'exécution pas à pas de l'algorithme par des manipulations de suites finies de symboles : on dira que l'algorithme est symboliquement exécutable, ou, pour reprendre une expression informatique, qu'il

est exécutable par un calcul papier-crayon. On peut alors reformuler la thèse A de la manière suivante : toute fonction calculable par une machine est calculable par une procédure symboliquement descriptible, et symboliquement exécutable.

Ainsi reformulée, la thèse A devient une thèse de philosophie du langage non-triviale. Elle affirme la capacité de nos langages formels à exprimer toutes les procédures de manipulation *empirique* de l'information permettant d'implémenter un calcul. Pour le comprendre, imaginons que la thèse A soit fausse. Un physicien pourrait alors construire une machine permettant le calcul d'une fonction, qu'aucun algorithme ne nous permettrait d'évaluer. Pour fixer les idées, admettons la vérité de la thèse de Church informatique, et imaginons que cette machine puisse calculer une fonction non-Turing calculable, comme la fonction d'arrêt. Par le biais de cette machine, nous pourrions connaitre la valeur de la fonction d'arrêt en ses arguments. Aucun algorithme ne décrirait la procédure mathématique menant à ce résultat, puisque l'indécidabilité du problème de l'arrêt prohibe l'existence d'un tel algorithme. Par conséquent, aucun texte formel ne décrirait non plus l'exécution pas à pas de cet algorithme. Cette machine constituerait, dans la terminologie des informaticiens, un « oracle », soit une boite noire associant une sortie à une entrée, sans que rien ne spécifie son mode de fonctionnement au sein de nos langages formels.

Une telle éventualité n'est nullement interdite par l'indécidabilité du problème de l'arrêt, ou par un autre résultat similaire d'indécidabilité. Ceux-ci reposent sur la vérité de la thèse de Church informatique, tout d'abord, et sont fondés sur des raisonnements portant uniquement sur les algorithmes, et non sur des machines physiques, ensuite.

En sus de son intérêt pour la philosophie du langage, la thèse A est essentielle à l'analyse du statut épistémologique des calculs effectués sur des machines. Dans la situation actuelle, nos machines nous permettent d'effectuer des calculs que nous sommes *pratiquement* incapables d'effectuer par un calcul papier-crayon, pour des raisons tenant essentiellement à leurs coûts en temps et en espace. Certaines de nos connaissances mathématiques sont donc justifiées sur la base de considérations empiriques. Cependant, *en droit,* nos machines digitales sont des implémentations de machines de Turing universelles, et sont donc simulables par un calcul papier-crayon. Si nous atteignons *de facto* une partie de notre connaissance mathématique par des moyens empiriques, nous sommes toujours capables *de jure* de l'atteindre par l'exécution symbolique du calcul.

Si la thèse A était fausse, ce statut épistémologique serait bouleversé. Nous atteindrions une partie de notre connaissance mathématique –à savoir, le résultat de certains calculs- par des moyens *purement* empiriques, sans qu'aucune procédure symbolique ne nous permette, même en droit, de justifier cette même connaissance. Il existerait alors une connaissance mathématique, portant sur la valeur de certaines fonctions en certains de leurs arguments, qui serait justifiée uniquement par des considérations empiriques, portant sur le bon fonctionnement d'une machine de calcul.

Ces dernières considérations permettent de souligner la pertinence de la thèse A pour l'étude de la notion d'a priori en philosophie des mathématiques. Si la connaissance obtenue par le calcul est fréquemment tenue pour a priori, c'est parce que le seul moyen nécessaire *en droit* pour justifier un résultat mathématique comme la valeur d'une fonction en un argument est la lecture et compréhension de textes formels : la lecture du programme permettant le calcul de la fonction, et la lecture du texte explicitant l'exécution du calcul pour une entrée donnée. Si la thèse A était fausse, il existerait donc une connaissance mathématique *a posteriori* : nous connaîtrions le résultat de certains calculs, sans qu'aucun texte formel n'explicite la méthode, et l'application de cette méthode à une entrée donnée, par lesquelles nous pourrions justifier a priori à cette connaissance. La thèse A fonde donc le caractère a priori de la connaissance obtenue par le calcul : elle justifie qu'on puisse affirmer qu'en droit, l'intégralité des résultats de calcul est justifiée par des considérations portant uniquement sur des textes.

Nous concluons en montrant pourquoi, malgré son apparente plausibilité, il n'est pas trivial de démontrer la thèse A.