Comment multiplier les très grands NOMBRES ENTIERS ?

RENCONTRE AVEC **JORIS VAN DER HOEVEN (LIX)**

—

PROPOS RECUEILLIS PAR ROBERT RANQUET (72)

JR : Vous venez de vous faire remarquer par la communauté scientifique internationale pour votre publication, avec David Harvey de l'Université de Nouvelle-Galles du Sud, d'une méthode révolutionnaire pour multiplier de très grands nombres entiers. Mais où est la nouveauté ? On ne savait pas faire de multiplications ?

Si, bien sûr ! Mais les algorithmes de multiplication n'ont pas cessé de progresser. Il y a eu d'abord la méthode que nous apprenons tous à l'école primaire, qui remonte aux Babyloniens. Cet algorithme simple a une complexité en N^2 : c'est-à-dire que si vous prenez un très grand nombre, mettons avec 1 milliard de chiffres (vous pouvez tout à fait stocker un nombre aussi grand que ça dans votre smartphone !), et que vous le multipliez par lui-même, cela fera 10^{18} opérations. Même avec un calculateur hyperrapide, à raison d'une nanoseconde par opération, cela fera tout de même 10^9 secondes, soit environ trente ans… On voit tout de suite qu'il faut s'y prendre autrement. C'est dans les années 50 qu'un Russe, Anatoli Karatsouba, a mis au point un algorithme beaucoup plus rapide, qui permet de multiplier des nombres assez grands. Cet algorithme est encore couramment utilisé aujourd'hui pour des nombres entre 100 et 1 000 chiffres, par exemple dans les opérations de cryptologie. Sa complexité est de $N^{\log 3/\log 2}$, soit $N^{1,58\ldots}$.

Puis sont venues les méthodes modernes, basées sur la transformation de Fourier rapide (FFT), inventées indépendamment par Pollard et Schönhage-Strassen en 1971, qui permettent de traiter des nombres au-delà du millier de chiffres. À l'époque, Schönhage et Strassen ont conjecturé qu'on pourrait trouver un algorithme encore plus rapide, avec une complexité en N Log N. C'est de cet algorithme que nous venons de prouver l'existence, avec mon collègue David Harvey. Nous en avons prouvé l'existence, et nous en avons donné une description explicite. Peut-on encore faire mieux et aller plus vite ? À vrai dire, on ne sait pas, mais je pense qu'on ne pourra pas. J'en serais en tout cas très étonné, même si je ne peux pas prouver que c'est possible. Mais il serait tout aussi difficile de prouver que ce n'est pas possible !… C'est une vieille difficulté bien connue des mathématiciens : dans un problème donné, on prouve facilement qu'on a une borne supérieure, mais il est toujours beaucoup plus difficile de prouver qu'il existe une borne inférieure.

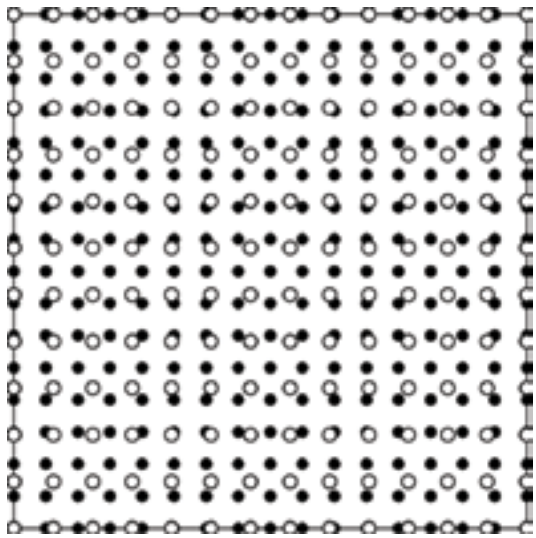

Illustration de la technique de « rééchantillonnage gaussien », qui est un des ingrédients essentiels pour le nouvel algorithme.

On voit bien que la question sous-jacente est celle de la complexité des algorithmes : beaucoup de ces complexités d'algorithmes s'expriment en fonction de la complexité de la multiplication. Par exemple, la complexité d'un algorithme pour calculer la $N^{\text{ème}}$ décimale de π est log N fois la complexité de la multiplication, c'est-à-dire $N (\text{Log } N)^2$, avec notre nouvel algorithme. La multiplication des entiers joue un peu en informatique le rôle de la vitesse de la lumière en physique.

Pourra-t-on appliquer cet algorithme ? Ce ne sera pas chose facile, en tout cas pas avec la description que nous en avons donnée. Mais nous avons utilisé pour notre recherche tout un tas de techniques intéressantes en soi, qui pourront être utilement réutilisées pour améliorer les algorithmes existants, y compris pour des nombres de quelques milliards de chiffres. Au-delà, il existe encore beaucoup d'autres types d'algorithmes sur lesquels il y a des choses à trouver, ceux qui interviennent par exemple dans les calculs de la division, de PGCD, de calcul sur les polynômes, etc.