

Calculs symboliques sur des algorithmes théoriques de produits de matrices

T. Sibut-Pinote,

Inria, École Polytechnique

É. Schost

University of Western Ontario

thomas.sibut-pinote@inria.fr

eschost@uwo.ca

La complexité optimale du produit matriciel, opération élémentaire fondamentale du calcul formel, est une question cruciale pour des raisons tant pratiques que théoriques. Pourtant, elle est à ce jour un problème ouvert.

Le produit de $A \in \mathcal{M}_{m,n}(\mathbb{K})$ et $B \in \mathcal{M}_{n,p}(\mathbb{K})$, noté (m, n, p) , se calcule de manière naïve en mnp multiplications. Strassen [6] fut le premier à introduire un algorithme sous-cubique, qui motiva l'introduction de la constante $\omega = \inf \{ \tau \mid \text{la complexité optimale du calcul de } (m, n, p) \text{ est } \mathcal{O}((mnp)^{\frac{\tau}{3}}) \}$. On sait que $2 \leq \omega < 3$ et on conjecture que $\omega = 2$.

De nombreux travaux donnent des bornes inférieures sur ω , la meilleure étant actuellement celle de Le Gall [1] : $\omega < 2.3728639$. Les algorithmes correspondants ne sont pas considérés comme utilisables en pratique : rarement implémentés, ils ne sont parfois pas complètement explicités, voire proviennent d'un argument d'existence non constructif.

Un algorithme de produit de matrices peut s'écrire comme un tenseur $t := \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^p a_{i,k} \otimes b_{k,j} \otimes c_{i,j}$ et la question de sa complexité correspond à la recherche d'une écriture de t avec un nombre minimal r de tenseurs élémentaires : on sait alors en déduire que $\omega \leq \frac{\log(r)}{\log(mnp)}$. Par exemple, Strassen [6] écrit le produit $(2, 2, 2)$ avec 7 tenseurs élémentaires, ce qui donne par un argument récursif $\omega \leq \log_2(7) \simeq 2.81$.

Le τ -théorème de Schönhage [3] permet de déduire des bornes sur ω à partir d'une telle formule calculant *plusieurs* produits de matrices indépendants. On peut le combiner à une autre technique connue qui consiste à calculer non plus dans $\mathcal{K}$, mais dans $\mathcal{K}[\varepsilon]$. Par exemple, si on appelle t_1 le tenseur

$$\sum_{i=0}^{m-1} \sum_{k=0}^{p-1} (x_{i,0} + \varepsilon u_{0,k,i}) \otimes (y_{0,k} + \varepsilon v_{k,i,0}) \otimes (\varepsilon^2 z_{k,i} + w_{0,0}) - \left(\sum_{i=0}^{m-1} x_{i,0} \right) \otimes \left(\sum_{k=0}^{p-1} y_{0,k} \right) \otimes w_{0,0},$$
il peut s'écrire $t_1 = \varepsilon^2 t_2 + \varepsilon^3 t_3$ où t_2 calcule les produits de matrices indépen-

dants $(m, 1, p)$ et $(1, (m-1)(p-1), 1)$ en seulement $mp+1$ opérations au lieu de $2mp - (m+p-1)$. Cela implique $\omega < 2.55$. Pan [2] présente de nombreux algorithmes de ce type qui produisent des valeurs de plus en plus petites de ω . Cependant, ces résultats reposent sur le fait que les multiplications de polynômes et la prise de modulo qui permettent de transformer ces tenseurs avec ε en algorithmes effectifs sont asymptotiquement négligeables devant le reste des calculs ; or ce n'est pas le cas pour les petites tailles de matrices.

Notre contribution est de deux natures : en examinant des tenseurs obtenus en composant ceux de Pan puis en extrayant certains termes, nous avons découvert un moyen de produire des tenseurs sans ε [4] ; cela nous a mené à développer une bibliothèque [5] qui permet de parser les algorithmes du type de ceux de Pan sous un format agréable pour l'utilisateur, de les transformer en arbres de syntaxe abstraite, de les manipuler (notamment les composer) et de les extraire vers du code Maple ou C++. Elle est écrite en Ocaml.

L'ambition de cette bibliothèque n'est pas de fournir des algorithmes efficaces "clés en mains", mais plutôt de fournir un environnement pour manipuler les algorithmes de produits de matrices (théoriques ou non) de manière formelle, ce qui peut être utile pour guider la compréhension du comportement de ces algorithmes à la fois dans un cadre de recherche et d'enseignement.

Bibliographie

- [1] François Le Gall. Powers of tensors and fast matrix multiplication. *CoRR*, abs/1401.7714, 2014.
- [2] Victor Pan. *How to multiply matrices faster*. Springer-Verlag New York, Inc., New York, NY, USA, 1984.
- [3] Arnold Schönhage. Partial and total matrix multiplication. *SIAM J. Comput.*, 10(3) :434–455, 1981.
- [4] Thomas Sibut-Pinote. Internship report. 2012. Available at http://perso.ens-lyon.fr/thomas.sibutpinote/rapport_matprod.pdf.
- [5] Thomas Sibut-Pinote and Eric Schost. Matrix product experiments, 2012. Available at <https://gitlab.com/matrix-product-experiments/matrix-product-experiments>.
- [6] Volker Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 14(3) :354–356, 1969.