Optimal Matrix Multiplication

by Stefano Gualandi

$\alpha \equiv \sim$
$\beta \equiv \sim$
$\gamma \equiv \sim$

et le sens est inversé : les lignes à colonne

1 Introduction

2 Notation

We are interested in solving by nonlinear programming techniques.

Consider the following $n \times n$ matrix multiplication, with $n = 2$.

$$\begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \times \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix} = \begin{bmatrix} c_1 & c_2 \\ c_3 & c_4 \end{bmatrix} = \begin{bmatrix} a_1b_1 + a_2b_3 & a_1b_2 + a_1b_4 \\ a_3b_1 + a_4b_3 & a_3b_2 + a_4b_4 \end{bmatrix}$$

We want to write the single multiplications block as follows.

$$\begin{aligned} M_k &= (\alpha_{k,1}a_1 + \alpha_{k,2}a_2 + \alpha_{k,3}a_3 + \alpha_{k,4}a_4)(\beta_{k,1}b_1 + \beta_{k,2}b_2 + \beta_{k,3}b_3 + \beta_{k,4}b_4) \\ &= \sum_{i \in [n^2]} \sum_{j \in [n^2]} \alpha_{k,i}\beta_{k,j}(a_ib_j), \quad \forall k \in [K]. \end{aligned} \tag{1}$$

√

Given a number K of multiplication that we allow (we bet on), we can compute each single coefficient of the resulting multiplication matrix as follows.

$$c_h = \sum_{k \in [K]} \gamma_{kh} M_k = \sum_{k \in [K]} \sum_{i \in [n^2]} \sum_{j \in [n^2]} \gamma_{kh}\alpha_{k,i}\beta_{k,j}(a_ib_j), \quad \forall h \in [n^2]. \tag{2}$$

√

Note that we can write the element c_1 in the following vector notation.

$$\begin{array}{ccccccccc} & & a_1b_1 & a_2b_1 & \cdots & a_2b_3 & \cdots & a_4b_4 & \\ c_1 & = [& 1 & 0 & \cdots & 1 & \cdots & 0 &] \end{array}$$

√

Finally, we can fix the following constraining for each single coefficients c_h.

$$c_{h,ij} = \sum_{k \in [K]} \gamma_{kh}\alpha_{k,i}\beta_{k,j}, \quad \forall h \in [n^2], \forall i \in [n^2], \forall j \in [n^2].$$

√ au final on doit retomber sur les mêmes formule pour chaque ci que la formule de base

The decision variables are $\gamma_{kh}, \alpha_{k,i}, \beta_{k,j}$, which to begin with, take values in the domain $\{-1, 0, 1\}$. Since we want to minimize the number of additions, while the number of multiplications is upper bounded by K, we can write the following integer nonlinear programming problem, which minimizes the number of variables which are nonzero.

$$\begin{aligned} \min \quad & \|\alpha\|_1 + \|\beta\|_1 + \|\gamma\|_1 & & (3) \\ \text{s.t.} \quad & \sum_{k \in [K]} \gamma_{kh}\alpha_{k,i}\beta_{k,j} = c_{h,ij}, & \forall h \in [n^2], \forall i \in [n^2], \forall j \in [n^2], & \quad (4) \\ & \gamma_{kh} \in \{-1, 0, 1\}, & k \in [K], \forall h \in [n^2], & \quad (5) \\ & \alpha_{k,i}, \beta_{k,j} \in \{-1, 0, 1\}, & \forall k \in [K], \forall i \in [n^2], \forall j \in [n^2]. & \quad (6) \end{aligned}$$

The only data of the problem are the values $c_{h,ij}$ that can only takes value 0 or 1, dependeding on the formula for multipling two $n \times n$ matrices.

↳ normal n^3 formula

Example 1. *The well-known algorithm of Straussen is obtained by considering for $n = 2$ and $K = 7$ the following rank K matrices, which represent a feasible solution (but is it provable optimal?) for the previous nonlinear programming problem.*

$$\alpha = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & -1 \end{bmatrix},$$

$$\beta = \begin{bmatrix} 1 & 1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 1 & 0 & 1 \end{bmatrix},$$

$$\gamma = \begin{bmatrix} 1 & 0 & 0 & 1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix},$$

which results in the following sequence of operations.

$$\begin{aligned}
M_1 &= (a_1 + a_4)(b_1 + b_4) \\
M_2 &= (a_3 + a_4)(b_1) \\
M_3 &= (a_1)(b_2 - b_4) \\
M_4 &= (a_4)(b_3 - b_1) \\
M_5 &= (a_1 + a_2)(b_4) \\
M_6 &= (a_3 - a_1)(b_1 + b_2) \\
M_7 &= (a_2 - a_4)(b_3 + b_4) \\
c_1 &= M_1 + M_4 - M_5 + M_7 \\
c_2 &= M_3 + M_5 \\
c_3 &= M_2 + M_4 \\
c_4 &= M_1 - M_2 + M_3 + M_6
\end{aligned}$$

Note that the coefficient of a_i, b_j, M_k are given by the corresponding values for α, β, γ.

2.1 Challenges

First, I would try to reproduce the result for $n = 2$. Later, I would move to $n = 3$ and $K = 23$. For $n = 4$, I should check the best known value for K.

- Scaling to large n. For $n = 3$, best known is $K = 23$.
- Handling rectangular matrices (more a matter of notation).
- Exploiting symmetries (how to verify that we find a new real algorithm?). con swap colums
- Math-heuristic to find feasible solutions (with a small value of K, we might find a new multiplication algorithms).
- Proving infeasibility for a given value of K.

Comments by PB:

logique

1. Rather than $||\alpha||_1$, the number of *additions* is $\sum_{k\in[K]}(\sum_{i\in[n^2]} |\alpha_{k,i}| - 1) = ||\alpha||_1 - K$, and the same for β and γ; the objective function is still valid, but we might want to explain it.

$\begin{pmatrix}1\\0\\0\\1\end{pmatrix} \Rightarrow a_1 + a_4 \downarrow$ 1 addition

2. The 1-norm is valid if all variables are bounded (in module) by q with q small enough: if $|\alpha_{k,2}| = 3$, instead of multiplying a_2 by 3 (one multiplication) we'd rather sum a_2 three times (still cheaper than actual CPU multiply by 3).

3 Links

- Nice dissemination post about hybrid methods for discovering new matrix multiplication algorithms:

  https://www.quantamagazine.org/ai-reveals-new-possibilities-in-matrix-multiplication-20221123/

2 options

$x \cdot y \cdot z$

$w = x \cdot y$

$x \cdot w \Rightarrow$ (1) gurobi set parameter "NonConvex" to 2

(2) McCormick envelope

$-1 \leq x \leq 1$
$-1 \leq w \leq 1$ } already the case
$-1 \leq z \leq 1$

+ if one is binary we can do as seen in the course. → not the case