TP 8: les matrices

Guillaume Aubian

Ce document a été grandement inspiré par – et largement copié sur – le document correspondant du responsable précédent de ce cours, Juliusz Chroboczek, avec son accord.

1 Opérations classiques sur les matrices

Faire cet exercice dans un fichier `basic.c`

1. Étant données deux matrices carrées $N \times N$ `A` et `B`, écrire une fonction `int** addmatrices(int** A, int** B, int N)` qui les additionne.
2. Étant données deux matrices carrées $N \times N$ `A` et `B`, écrire une fonction `int** multmatrices(int** A, int** B, int N)` qui les multiplie.
3. Écrire une fonction `int** id(int N)` qui renvoie la matrice identité $N \times N$.
4. Écrire une fonction `int** zero(int N)` qui renvoie la matrice nulle $N \times N$.

2 Exponentiation

Créez un fichier `expo.c` dans lequel vous coderez les fonctions de cet exercice. Importez `basic.c`.

1. Avec une simple boucle for, codez une fonction `int** puissance(int** A, int N, int k)` qui passe une matrice $N \times N$ `A` à la puissance k.
2. En fait, la fonction puissance vérifie la récurrence suivante :

$$A^0 = Id$$
$$A^{2k} = A^k \times A^k$$
$$A^{2k+1} = A^k \times A^k \times A$$

Déduire de cette récurrence un algorithme rapide pour élever une matrice à une certaine puissance.

La suite de Fibonacci est définie par $f_0 = 0$, $f_1 = 1$ et pour tout $k > 1$, $f_k = f_{k-1} + f_{k-2}$. Pour $k \geq 0$, posons $F_k = \begin{bmatrix} f_k \\ f_{k+1} \end{bmatrix}$.

3. Que vaut F_0 ? Que vaut F_{k+1} en fonction de F_k ?
4. À partir des deux questions précédentes, déduire une fonction qui calcule très rapidement le k-ème terme de la suite de Fibonacci.

5. Recodez l'exponentiation rapide (et donc la multiplication de matrices) pour des matrices de flottants.

6. Écrire une fonction `float trace(float** A, int N)` qui renvoie la trace de la matrice $N \times N$ `A`.

Pour calculer les valeurs propres d'un matrice, l'algorithme suivant fonctionne :

1. si la matrice a une trace non-nulle, on divise tous les éléments de la matrice par sa trace,
2. on élève la matrice à une grande puissance
3. si la matrice initiale avait une trace non-nulle, on multiplie tous les éléments du résultat par la trace de la matrice initiale,
4. les valeurs propres sont les éléments sur la diagonale du résultat

7. Implémentez ceci

8. Appliquez ceci à la matrice de la question 4. Qu'observez-vous ?

3 Multiplication de matrices plus rapides

Notre algorithme de multiplication de matrics fait pour l'instant environ N^3 opérations pour multiplier deux matrices de taille $N \times N$: on peut faire mieux !

On va supposer dans cette question que N est une puissance de 2 pour se simplifier la vie. Si on veut multiplier deux matrices A et B pour obtenir une matrice C, avec les 3 matrices de la forme :

$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{A}_{1,2} \\ \mathbf{A}_{2,1} & \mathbf{A}_{2,2} \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \mathbf{B}_{1,1} & \mathbf{B}_{1,2} \\ \mathbf{B}_{2,1} & \mathbf{B}_{2,2} \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{C}_{1,1} & \mathbf{C}_{1,2} \\ \mathbf{C}_{2,1} & \mathbf{C}_{2,2} \end{bmatrix}$

On peut alors simplement faire:

$$C_{1,1} = A_{1,1} \times B_{1,1} + A_{1,2} \times B_{2,1}$$

$$C_{1,2} = A_{1,1} \times B_{1,2} + A_{1,2} \times B_{2,2}$$
$$C_{2,1} = A_{2,1} \times B_{1,1} + A_{2,2} \times B_{2,1}$$
$$C_{2,2} = A_{2,1} \times B_{1,2} + A_{2,2} \times B_{2,2}$$

Mais en fait, si on fait les 8 multiplications récursivement, on retombe sur une complexité en N^3...

Faisons mieux ! On définit les matrices intermédiaires M_i comme dans la figure 1

On peut alors conclure avec les opérations de la figure 2. Implémentez ceci !

$$
\begin{aligned}
\mathbf{M}_1 &:= (\mathbf{A}_{1,1} + \mathbf{A}_{2,2})(\mathbf{B}_{1,1} + \mathbf{B}_{2,2}) \\
\mathbf{M}_2 &:= (\mathbf{A}_{2,1} + \mathbf{A}_{2,2})\mathbf{B}_{1,1} \\
\mathbf{M}_3 &:= \mathbf{A}_{1,1}(\mathbf{B}_{1,2} - \mathbf{B}_{2,2}) \\
\mathbf{M}_4 &:= \mathbf{A}_{2,2}(\mathbf{B}_{2,1} - \mathbf{B}_{1,1}) \\
\mathbf{M}_5 &:= (\mathbf{A}_{1,1} + \mathbf{A}_{1,2})\mathbf{B}_{2,2} \\
\mathbf{M}_6 &:= (\mathbf{A}_{2,1} - \mathbf{A}_{1,1})(\mathbf{B}_{1,1} + \mathbf{B}_{1,2}) \\
\mathbf{M}_7 &:= (\mathbf{A}_{1,2} - \mathbf{A}_{2,2})(\mathbf{B}_{2,1} + \mathbf{B}_{2,2})
\end{aligned}
$$

Figure 1: Les matrices intermédiaires

$$
\begin{aligned}
\mathbf{C}_{1,1} &= \mathbf{M}_1 + \mathbf{M}_4 - \mathbf{M}_5 + \mathbf{M}_7 \\
\mathbf{C}_{1,2} &= \mathbf{M}_3 + \mathbf{M}_5 \\
\mathbf{C}_{2,1} &= \mathbf{M}_2 + \mathbf{M}_4 \\
\mathbf{C}_{2,2} &= \mathbf{M}_1 - \mathbf{M}_2 + \mathbf{M}_3 + \mathbf{M}_6
\end{aligned}
$$

Figure 2: Les opérations finales