Algorithmes et architectures systoliques

Résumé: Dans ce TD, nous étudions le produit de matrice et la décomposition LU sur des architectures systoliques pour les matrices bandes. Le nombre de cellules des réseaux systoliques adaptés au traitement de telles matrices ne dépend que de la largeur de la bande et non de la taille de la matrice.

1 Produit de matrices à structure bande

Une matrice bande possédant p super-diagonales et q sous diagonales a donc une bande de largeur $w = p + q - 1$.

$$A = \begin{bmatrix} \times & \times & \times & & & & \\ \times & \ddots & \ddots & \ddots & & & \\ & \ddots & \ddots & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \times & \\ & & & \ddots & \ddots & \times & \\ & & & & \times & \times & \end{bmatrix}$$

FIG. 1 – Matrice bande avec $p = 3$ et $q = 2$

▷ **Question 1.** Soient A une matrice bande (p_A, q_A) et B une matrice bande (p_B, q_B). Montrer que $C = A \cdot B$ est également une matrice bande.

Réponse. Commençons par remarquer que pour tout matrice bande M, on a $M_{ij} \neq 0 \Leftrightarrow 1 - q_M \leqslant j - i \leqslant p_M - 1$. Étant donné que $c_{ij} = \sum_k a_{ik} b_{kj}$, observons les éléments de cette somme :

$$\begin{aligned} a_{ik} \neq 0 &\Leftrightarrow i + 1 - q_A \leqslant k \leqslant i + p_A - 1 \\ b_{kj} \neq 0 &\Leftrightarrow j + 1 - p_B \leqslant k \leqslant j + q_B - 1 \\ a_{ik} b_{kj} \neq 0 &\Leftrightarrow 2 - (q_A + q_B) \leqslant k \leqslant (p_A + p_B) - 2 \end{aligned}$$

On a donc $c_{ij} \neq 0 \Leftrightarrow 2 - (q_A + q_B) \leqslant k \leqslant (p_A + p_B) - 2$, soit $p_C = p_A + p_B - 1$ et $q_C = q_A + q_B - 1$. □

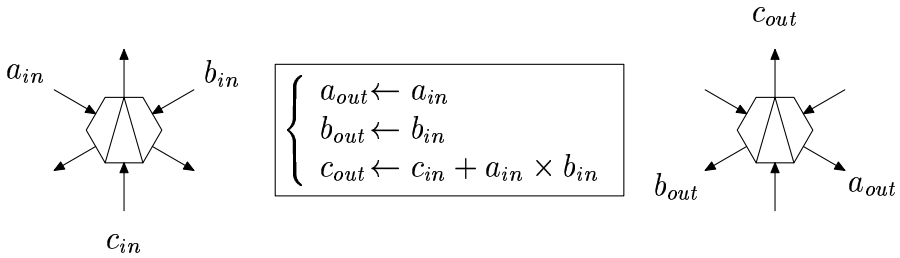
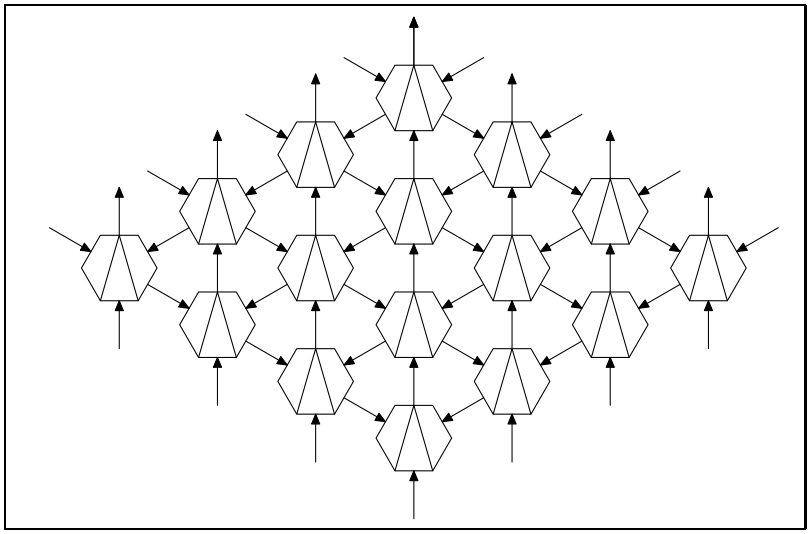


FIG. 2 – Programme des cellules pour le réseau de Kung et Leiserson

▷ **Question 2.** Proposer un réseau permettant de calculer le produit d'une matrice bande (P, Q) par une matrice bande (Q, P) en utilisant les cellules décrites en Figure 2.

Réponse.
Nous allons traiter le cas du produit d'une matrice bande $(3, 2)$ par une matrice bande $(2, 3)$. La matrice résultante étant une matrice bande $(4, 4)$, il est assez naturel d'utiliser le réseau suivant :



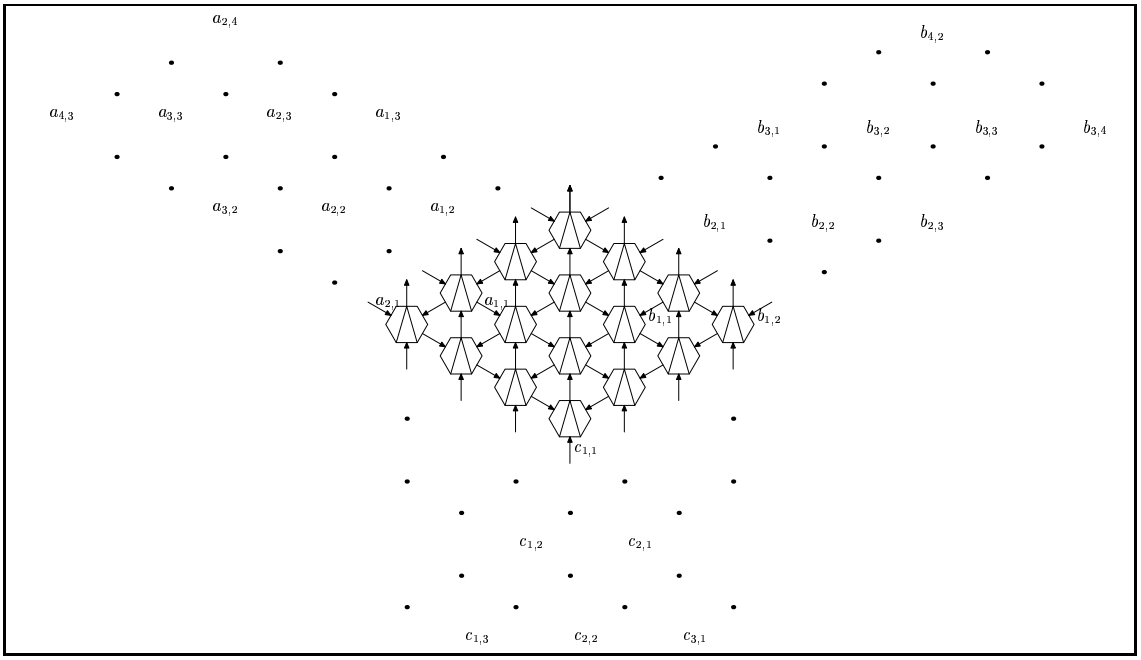
Quelles sont les contraintes auxquelles on fait face ?

1. Tout d'abord, étant donné la structure des matrices, il est impossible de les faire rentrer par ligne si on s'impose un réseau dont la taille est indépendante de la taille des matrices. Il est bien plus naturel de les faire rentrer pas diagonales. Au vu du type de cellule que l'on peut utiliser il est raisonnable de faire entrer A par la gauche, B par la droite et C par le bas. On pourra vérifier que les dimensions du réseau sont bien compatibles avec la structure de A, B et C.

2. La diagonale de C étant celle qui requière le plus de calcul, elle doit circuler vers le haut en passant au milieu du réseau. Considérons par exemple le calcul de c_{33}. On a $c_{33} = a_{32}b_{23} + a_{33}b_{33} + a_{34}b_{43} + a_{35}b_{53}$. Supposons que c_{33} soit en $(0,0)$ au temps t. Alors a_{32} et b_{23} sont aussi en $(0,0)$ au temps t. Au temps $t+1$, c_{33} est en $(1,1)$ et donc a_{33} et b_{33} aussi. a_{33} était donc en $(2,1)$ au temps t et b_{33} en $(1,2)$ au temps t. Au temps $t+2$, c_{33} est en $(2,2)$ et en refaisant le même raisonnement, on en déduit que a_{34} doit être en $(4,2)$ au temps t et b_{43} en $(2,4)$. On sait donc que si le premier élément d'une ligne de A est en $(0,0)$ au temps t, le $d^{\text{ème}}$ est en $(2d,d)$ au temps t ; si le premier élément d'une colonne de B est en $(0,0)$ au temps t, le $d^{\text{ème}}$ est en $(d,2d)$ au temps t.

3. Il nous faut maintenant savoir à quelles vitesse les éléments d'une même diagonale rentrent dans le réseau. Regardons par exemple l'élément de A qui précède a_{32} : a_{43}. a_{43} doit rencontrer b_{33} pour participer au calcul de $c_{43} = a_{43}b_{33} + a_{44}b_{43} + a_{45}b_{53}$. b_{33} étant en $(1,0)$ au temps $t+3$, c_{43} et a_{43} doivent également y être. On en déduit donc que a_{43} est en $(4,0)$ au temps t et c_{43} en $(-1,-2)$. Les éléments d'une même diagonale vont donc rentrer tous les trois tops dans le réseau.

On peut déduire des remarques précédentes le réseau suivant :

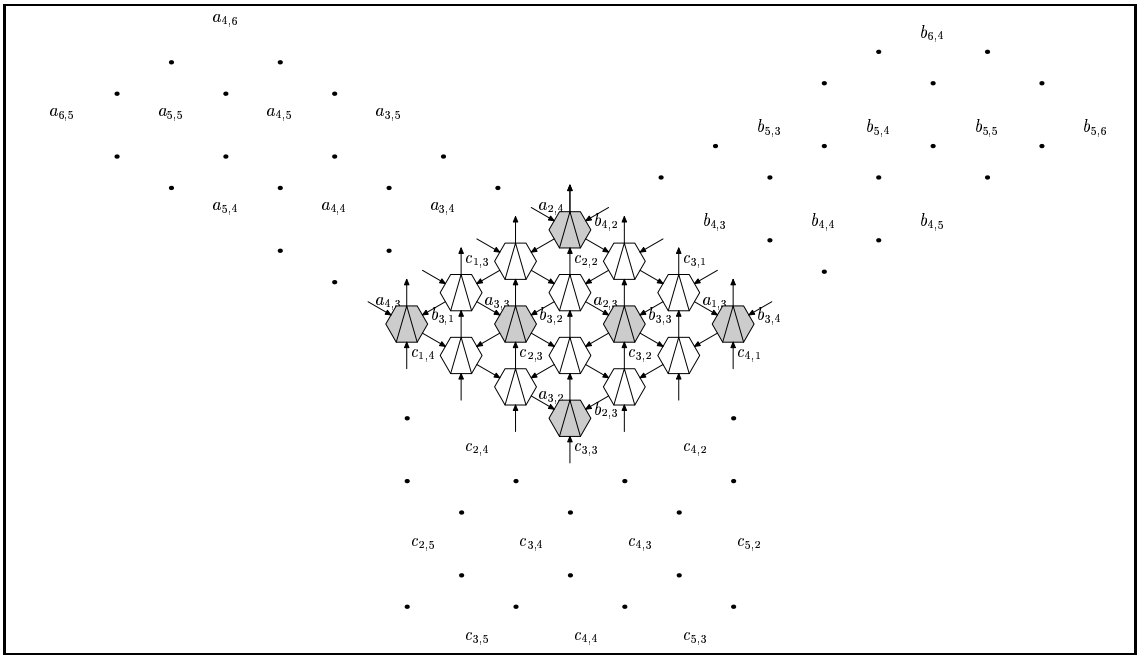


A : anti-diagonale d : $\begin{array}{l} a_{d,d+k} \\ a_{d+l,l} \end{array}$ $\begin{array}{l} (-t+3\cdot d+2\cdot k,\ k+Q\ ) \text{ pour } k\in[1,P] \\ (\ \ -t+3\cdot d+l\ \ ,-l+Q) \text{ pour } l\in[0,Q] \end{array}$

B : anti-diagonale d : $\begin{array}{l} b_{d,d+k} \\ b_{d+l,l} \end{array}$ $\begin{array}{l} (-k+Q,\ -t+3\cdot d+k\ \ ) \text{ pour } k\in[1,P] \\ (\ \ l+Q\ \ ,-t+3\cdot d+2\cdot l) \text{ pour } l\in[0,Q] \end{array}$

C : anti-diagonale d : $\begin{array}{l} c_{d,d+k} \\ c_{d+l,l} \end{array}$ $\begin{array}{l} (t-3\cdot d-2\cdot k+2,\ t-3\cdot d-k+2\ \ ) \text{ pour } k\in[1,P+Q] \\ (\ \ t-3\cdot d-l+2\ \ \ ,t-3\cdot d-2\cdot l+2) \text{ pour } l\in[0,P+Q] \end{array}$

À titre d'exemple, l'état du réseau au temps $t = 7$, est représenté ci-dessous.



Chaque cellule n'est active qu'une fois sur trois et le temps nécessaire pour effectuer le produit de matrice est donc de $3n + \mathcal{O}(P + Q)$. On remarquera qu'il est possible de pipeliner 3 produits de matrices en entrelaçant les matrices pour augmenter le rendement du réseau. □

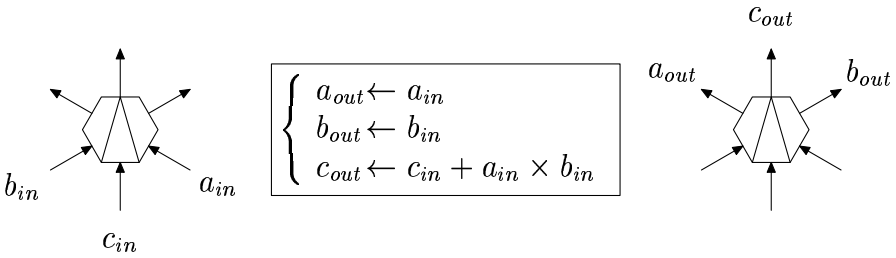
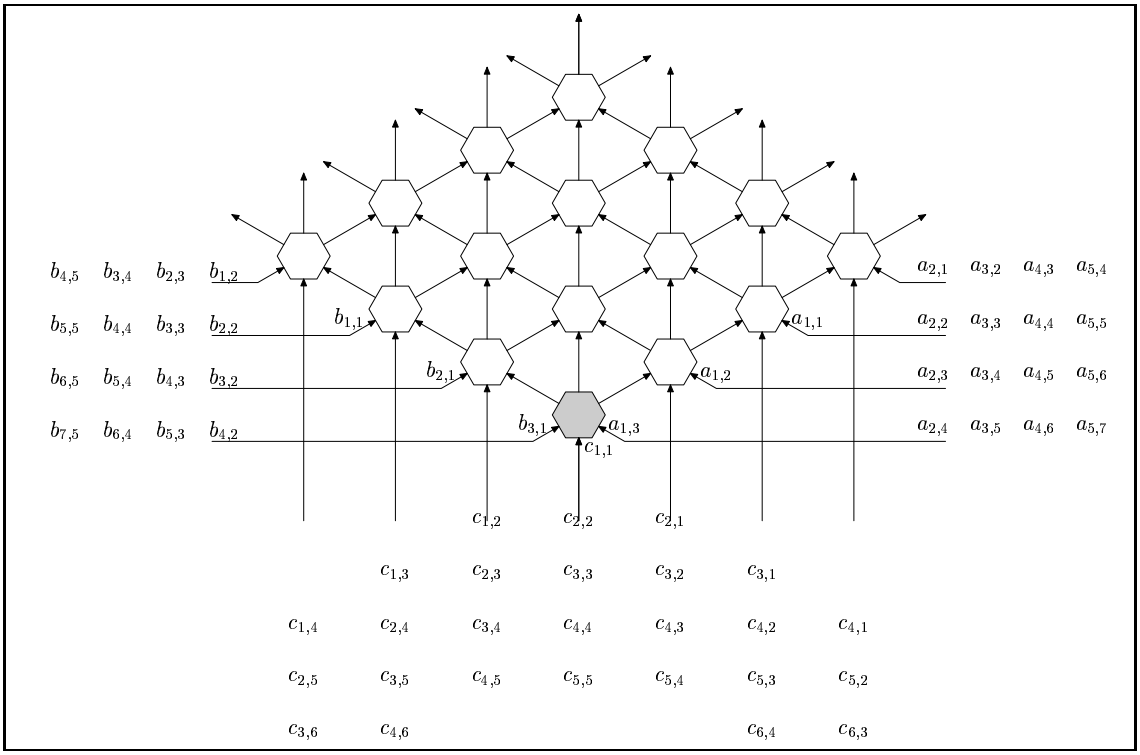
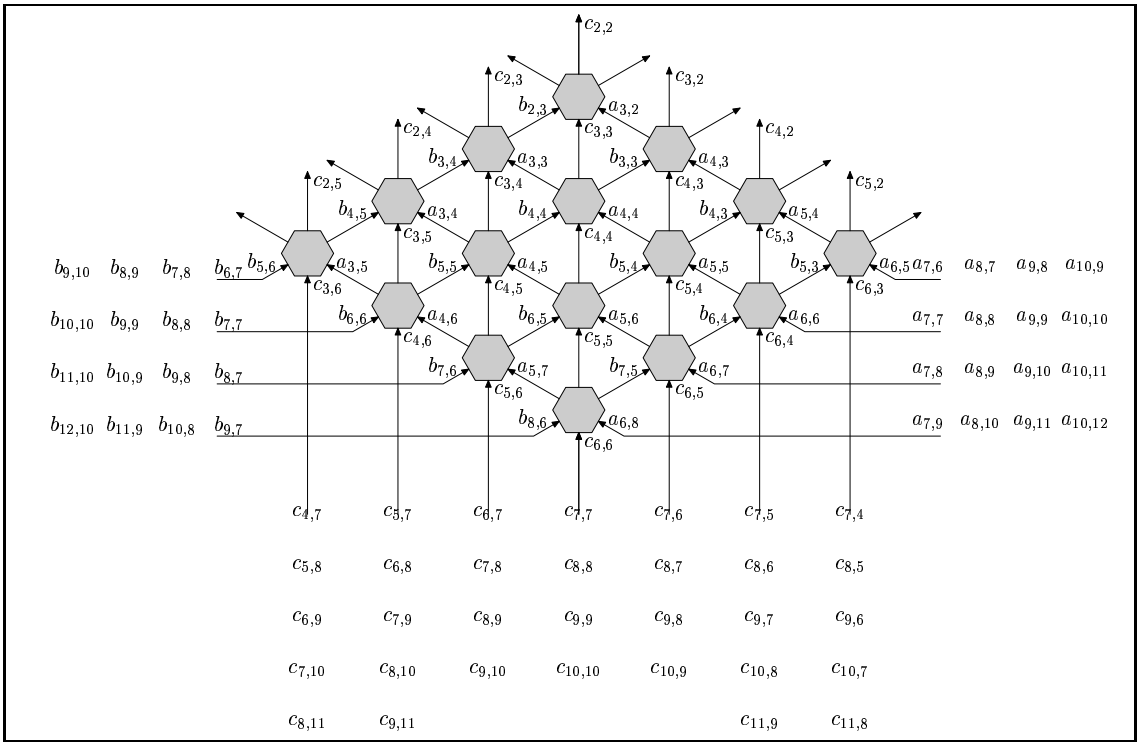


FIG. 3 – Programme des cellules pour le réseau de Weiser et Davis

▷ **Question 3.** Proposer un réseau utilisant les cellules décrite en Figure 3 et permettant de calculer le produit d'une matrice bande (p, q) par une matrice bande (q, p).

Réponse. En effectuant le même type d'analyse que précédemment, on peut construire le réseau suivant :





Son rendement est bien meilleur que le précédent et un produit est effectué en temps $n + \mathcal{O}(P+Q)$. □

2 Factorisation des matrices bandes

Triangularisation d'une matrice par l'algorithme de Gauss et décomposition LU sont intimement liées. En effet, trianguliariser A, c'est la pré-multiplier des matrices élémentaires triangulaires inférieurs à diagonale unité $L_1, L_2, \ldots, L_p$ de telle sorte que la matrice $L_p \ldots L_2 L_1 A = U$ soit triangulaire supérieure. L_i correspond à l'élimination de la $i^{\text{ème}}$ colonne de A et ne diffère avec l'identité que par sa $i^{\text{ème}}$ colonne. En notant L l'inverse de $L_p \ldots L_2 L_1$, on obtient donc la décomposition LU de A. Il se trouve que L s'obtient très facilement à partir des matrices L_i : il suffit de juxtaposer les $i^{\text{ème}}$ colonnes des L_i en changeant le signe des éléments non diagonaux.

On va réutiliser le réseau de la question 2 en introduisant un nouveau type de cellule (voir Figure 4.)

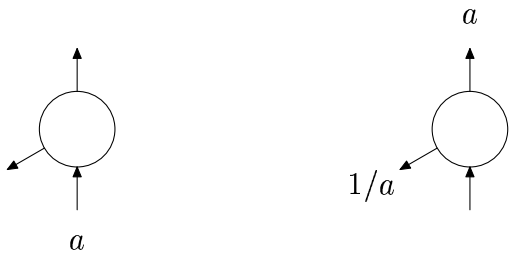


FIG. 4 – Cellule supplémentaire pour faire une décomposition LU

▷ **Question 4.** En observant le graphe de dépendances d'une élimination de Gauss, proposer une parallélisation possible et en déduire une architecture systolique réalisant la décomposition LU d'une matrice bande (p, p).

Réponse. On va faire une élimination ligne par ligne d'une matrice bande possédant 4 sous-diagonales et 4 super-diagonales :

– Instant t : annuler a_{21} à l'aide de a_{11} par combinaison des lignes 1 et 2.
– Instant $t+1$: mettre a_{22} à jour avec a_{12} en raison de la combinaison des lignes 1 et 2 ;
annuler a_{31} à l'aide de a_{11} par combinaison des lignes 1 et 3.
– Instant $t+2$: mettre a_{23} à jour avec a_{13} en raison de la combinaison des lignes 1 et 2 ;
mettre a_{32} à jour avec a_{12} en raison de la combinaison des lignes 1 et 3 ;
annuler a_{41} à l'aide de a_{11} par combinaison des lignes 1 et 4.
– Instant $t+3$: mettre a_{24} à jour avec a_{14} en raison de la combinaison des lignes 1 et 2 ;
mettre a_{33} à jour avec a_{13} en raison de la combinaison des lignes 1 et 3 ;
mettre a_{42} à jour avec a_{12} en raison de la combinaison des lignes 1 et 4 ;
annuler a_{32} à l'aide de a_{12} par combinaison des lignes 2 et 3.

À moins d'accepter de propager le pivot a_{11} à toutes les lignes en un seul top (ce qui n'est pas vraiment dans l'esprit des architectures systoliques), l'élimination des coefficients de la seconde colonne de A ne peut pas commencer avant l'instant $t+3$. Le réseau résultant ne peut délivrer la factorisation qu'en temps $3n+\mathcal{O}(w)$.

On va utiliser le même algorithme que pour les matrices dense et on a donc les récurrences suivantes :

$$
\begin{aligned}
a_{ij}^{(1)} &= a_{ij} \\
a_{ij}^{(k+1)} &= a_{ij}^{(k)} + l_{ik}(-u_{kj}) \\
l_{ik} &= \begin{cases} 0 & \text{si } i < k \\ 1 & \text{si } i = k \\ a_{ik}^{(k)}/u_{kk} & \text{si } i > k \end{cases} \\
u_{kj} &= \begin{cases} 0 & \text{si } k > j \\ a_{kj}^{(k)} & \text{si } k \leqslant j \end{cases}
\end{aligned}
$$

Le réseau qui permet de représenter ce calcul est représenté sur la Figure 5. □

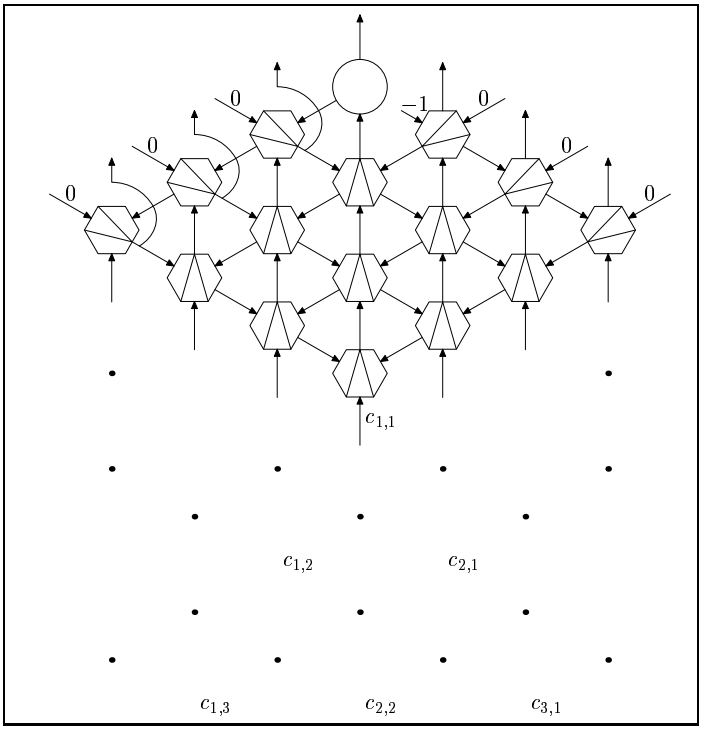


FIG. 5 – Réseau de Kung et Leiserson adapté à la décomposition LU