

ASUPRA PROGRAMĂRII LA MAȘINILE ELECTRONICE DE CALCUL A PRODUSULUI A DOUĂ MATRICE

DE

TEODOR RUS

*Comunicare prezentată de T. POPOVICIU, membru corespondent al Academiei R.P.R.,
în ședința din 10 septembrie 1962*

Să considerăm matricele

$$A = \| a_{ji} \|, B = \| b_{ik} \|, j = 1, \dots, n, i = 1, \dots, m, k = 1, \dots, p, \quad (1)$$

și produsul lor

$$C = A \cdot B = \| c_{jk} \|, c_{jk} = \sum_{i=1}^m a_{ji} b_{ik}. \quad (2)$$

În cele ce urmează vom prezenta un algoritm precum și schema logică operatorială a acestui algoritm prin care se realizează produsul matricelor A și B , înscrind în memoria mașinii electronice la care lucrăm exclusiv elementele diferite de zero ale celor două matrice.

Fără a restrînge generalitatea putem presupune că fiecare linie, respectiv coloană, a matricelor A și B posedă cel puțin un element diferit de zero.

Să presupunem că matricea A are l elemente diferite de zero, iar matricea B , r elemente diferite de zero.

Elementele diferite de zero ale matricei A le înscrîm în memoria mașinii în șirul de celule

$$a + 1, a + 2, \dots, a + l \quad (3)$$

în așa fel încît dacă ultimul element al liniei j a matricei A a fost înscris în celula $a + v$, atunci în celula $a + v + 1$ se va înscrie primul element diferit de zero al liniei $j + 1$ ș.a.m.d. Elementele diferite de zero ale matricei B le înscrîm în celulele

$$b + 1, b + 2, \dots, b + r \quad (4)$$

în mod analog, înscrîmarea făcîndu-se pe coloane.

Un element oarecare al matricei A este $a_v = a_{ji}$, unde j și i sînt indicii săi de linie respectiv coloană. Vom nota cu $i_v = i(a_v)$ indicele de coloană al unui element oarecare a_v . În mod analog pentru matricea B , $b_\mu = b_{ik}$ iar $i_\mu = i(b_\mu)$ este indicele de linie al elementului b_μ .

Se poate constata ușor că numărul maxim de cifre binare ale numerelor $i(a_v)$ și $i(b_\mu)$ scrise în sistem binar de numerație nu depășește numărul $1 + [\log_2 n]$ (zecimal).

Fie acum șirul de numere binare

$$\alpha_1, \alpha_2, \dots, \alpha_v, \dots, \alpha_l, \quad (5)$$

respectiv

$$\gamma_1, \gamma_2, \dots, \gamma_\mu, \dots, \gamma_r, \quad (6)$$

obținut prin transformarea indicilor de coloană $i(a_v)$, $v = 1, \dots, l$, respectiv $i(b_\mu)$, $\mu = 1, \dots, r$, în sistem binar de numerație.

Pentru determinarea locului unui element a_v , respectiv b_μ , în matricele A , respectiv B , vom atașa șirurilor de celule (3) și (4) șirurile de celule

$$d+1, d+2, \dots, d+s, \quad (5')$$

respectiv

$$h+1, h+2, \dots, h+t, \quad (6')$$

unde

$$s = \left\lceil \frac{l}{K} \right\rceil + 1, \quad t = \left\lceil \frac{r}{K} \right\rceil + 1, \quad K = \left\lceil \frac{H}{2 + [\log_2 n]} \right\rceil,$$

H fiind numărul ordenelor unei celule, al căror conținut pe ordine sînt respectiv termenii șirurilor

$$\alpha_1 \beta_\alpha^1, \alpha_2 \beta_\alpha^2, \dots, \alpha_v \beta_\alpha^v, \dots, \alpha_l \beta_\alpha^l \quad (5'')$$

și

$$\gamma_1 \beta_\gamma^1, \gamma_2 \beta_\gamma^2, \dots, \gamma_\mu \beta_\gamma^\mu, \dots, \gamma_r \beta_\gamma^r. \quad (6'')$$

Numerele β_α^v și β_γ^μ , pe care le numim indicatoare, se determină în felul următor:

$$\beta_\alpha^v = \begin{cases} 1, & \text{dacă după } a_v \text{ mai urmează și alte elemente diferite de zero} \\ & \text{în aceeași linie;} \\ 0, & \text{în caz contrar;} \end{cases}$$

$$\beta_\gamma^\mu = \begin{cases} 1, & \text{dacă după } b_\mu \text{ mai urmează și alte elemente diferite de zero} \\ & \text{în aceeași coloană;} \\ 0, & \text{în caz contrar.} \end{cases}$$

În lucrare se pun în evidență proprietăți ale acestor indicatoare, indicîndu-se printre altele modul în care ele determină indicii j și k ai unui element din matricea produs.

Pentru executarea produsului celor două matrice se caută toate perechile de elemente (a_v, b_μ) cu proprietatea $i(a_v) = i(b_\mu)$ și se execută produsul lor înscrindându-se într-un șir de celule $\{c' + p\}$ iar indicii corespunzători j și k , determinați cu ajutorul indicatoarelor β_α^v și β_γ^μ , se înscriu pe ordine în șirul de celule $\{c' + p\}$.

Schema logică operatorială a acestui algoritm se compune din trei blocuri după cum urmează:

Blocul A are schema logică operatorială:

$$\begin{aligned} & \downarrow^{a''} [\langle d+k_1 \rangle \rightarrow (\sigma_1)] \downarrow^{a_2} [\langle \sigma_1 \rangle \rightarrow (\sigma_2) \otimes N_q] D_+^{H-qk'+x+1}(\sigma_2) [\langle \sigma_1 \rangle \rightarrow (\sigma'_2) \otimes \\ & \otimes N_q] D_+^{H-qk'+x}(\sigma'_2) A_1 \downarrow^{a'} [\langle h+k_2 \rangle \rightarrow (\tau_1)] \downarrow^{a_1} [\langle \tau_1 \rangle \rightarrow (\tau_2) \otimes N_q] \times \\ & \times D_+^{H-qk'+x+1}(\tau_2) [\langle \tau_1 \rangle \rightarrow (\tau'_2) \otimes N_q] D_+^{H-qk'+x}(\tau'_2) A_2 \uparrow^b, \\ & x = H - Kk', \quad k' = 2 + [\log_2 n]. \end{aligned}$$

Blocul B are schema logică operatorială:

$$\begin{aligned} & \downarrow^6 p(\langle \sigma_2 \rangle \neq \langle \tau_2 \rangle) \uparrow^{10} \downarrow^{11} p(\langle \sigma'_2 \rangle \neq 0) \uparrow^8 \downarrow^9 p(\langle \tau'_2 \rangle \neq 0) \uparrow^6 \downarrow^7 p(j < l) \uparrow^4 \times \\ & \times p(\langle P_1 \rangle < K) \uparrow^2 p(\langle P_2 \rangle < K) \uparrow^1 F(j) F_2(q) O \uparrow^{a_1} \downarrow^1 F(k_2) F_2^{-K}(q) \times \\ & \times F^{-K}(\langle P_2 \rangle) F(j) O \uparrow^{a'} \downarrow^2 F(k_1) F_1^{-K}(q) F^{-K}(\langle P_1 \rangle) p(\langle P_2 \rangle < K) \uparrow^3 \times \\ & \times F(j) F_2(q) O \uparrow^{a''} \downarrow^3 F^{-K}(\langle P_2 \rangle) F(k_2) F_2^{-K}(q) F(j) O \uparrow^{a''} \downarrow^4 p(i < r) \uparrow^0 \times \\ & \times F^{-(l-1)}(j) F_1(q) F_2^{-K}(q) F^{-K}(\langle P_2 \rangle) p(\langle P_1 \rangle < K) \uparrow^5 O \uparrow^{a_2} \downarrow^5 F(k_1) \times \\ & \times O \uparrow^{a''} \downarrow^6 A_1 O \uparrow^7 \downarrow^8 A_2 O \uparrow^9 \downarrow^{10} A_3 [\langle r_2 \rangle \rightarrow (c' + p) \uparrow^{\frac{H}{2}}] [\langle r_1 \rangle \rightarrow (c' + p) \uparrow^{\frac{H}{2}}] F(p) O \uparrow^{11}. \end{aligned}$$

Blocul C are schema logică operatorială

$$\begin{aligned} & \downarrow^c \downarrow^3 \downarrow^1 p(\langle c' + k \rangle < \langle c' + i + 1 \rangle) \uparrow^4 \downarrow^7 \downarrow^5 p(i + 1 < p) \uparrow^2 F(i) O \uparrow^1 \downarrow^2 \times \\ & \times p(k < p - 1) \uparrow^8 F(k) F^{-i+k}(i) O \uparrow^3 \downarrow^4 p(\langle c' + k \rangle \neq \langle c' + i + 1 \rangle) \downarrow^6 \times \\ & \times [\langle c' + i + 1 \rangle \rightarrow (\delta_1)] \left[\langle c' + v \rangle \rightarrow (c' + v + 1) \right] \left[\langle \delta_1 \rangle \rightarrow (c' + k) \right] \times \\ & \times [\langle c + i + 1 \rangle \rightarrow (\delta_2)] \left[\langle c + v \rangle \rightarrow (c + v + 1) \right] \left[\langle \delta_2 \rangle \rightarrow (c + k) \right] \times \\ & \times O \uparrow^5 \downarrow^6 \Phi(A_1) A_1 \left[\langle c + \mu + 1 \rangle \rightarrow (c + \mu) \right] \left[\langle c' + \mu + 1 \rangle \rightarrow (c + \mu) \right] \times \\ & \left[\mu = i + 1, i + 2, \dots, p - 1 \right] \left[\mu = i + 1, i + 2, \dots, p - 1 \right] \times \\ & \times O \uparrow^7 \downarrow^8 R. \end{aligned}$$

R fiind fie operatorul de tipărire, fie un subbloc al blocului C care realizează aranjarea elementelor matricei produs în așa fel ca să fie posibilă efectuarea unor eventuale operații ulterioare. N_q și N'_q sînt numere ajutătoare precizate în lucrare iar $\otimes$ este operația de înmulțire logică cifră cu cifră.

Algoritmul descris de blocul C se poate aplica și la alte probleme, ca de exemplu la aranjarea nodurilor într-o formulă de interpolare a lui Newton pentru ca restul să fie minim [3], aranjarea cuvintelor într-un dicționar în lingvistica matematică [1] etc.

О ПРОГРАММИРОВАНИИ ПРОИЗВЕДЕНИЯ ДВУХ МАТРИЦ НА ЭЛЕКТРОННЫХ ВЫЧИСЛИТЕЛЬНЫХ МАШИНАХ

РЕЗЮМЕ

В этой статье излагается метод программирования произведения двух матриц, записывая в память машины только элементы, отличные от нуля, двух рассматриваемых матриц. Логическая операторная схема описанного алгоритма состоит из трех блоков A , B , C , предусмотренных для выявления всех пар элементов, произведение которых необходимо вычислить, так как на строках и на столбцах, соответственно, где они находятся, они занимают то же самое место, указанное показателями β_x^y и β_y^x , и их индексы $i(a_x)$ и $i(b_y)$ равны.

Подчеркнем, что роль блока C — упорядочить последовательность индексов произведения, а также и произведения вместе с соответствующими индексами таким образом, чтобы было возможно напечатать их в определенном порядке или перейти к следующим действиям выполняющимся аналогичным образом.

SUR L'ÉTABLISSEMENT DU PROGRAMME D'UN PRODUIT DE DEUX MATRICES DANS UNE CALCULATRICE ÉLECTRONIQUE

RÉSUMÉ

L'auteur présente une méthode permettant d'établir le programme d'un produit de deux matrices, en n'enregistrant, dans la mémoire de la calculatrice que les éléments différents de zéro, des deux matrices. Le schéma logique d'opération de l'algorithme décrit consiste en trois blocs A , B , C , ayant pour but de mettre en évidence toutes les paires d'éléments dont les produits doivent nécessairement être exécutés, puisque, dans les lignes respectives, les colonnes où ils se trouvent occupent la même place, marquée par les indicateurs β_x^y , et β_y^x , et leurs indices $i(a_x)$ et $i(b_y)$ sont égaux.

L'auteur fait remarquer que le bloc C a le rôle d'ordonner la suite d'indices des éléments-produit, ainsi que les éléments-produit avec leurs indices respectifs, de manière qu'il soit possible soit de les imprimer dans un ordre bien déterminé, soit de passer à des opérations ultérieures qui seront exécutées de la même façon.

BIBLIOGRAFIE

1. A. D. BOOTH, L. BRANDWOOD, I. P. CLEAVE. *Mechanical Resolution of Linguistic Problems*, London, 1958.
2. E. MUNTEANU, T. RUS, *O metodă de programare a produsului a două matrice*. St. cerc. mat. Cluj, 1961, **XII**, anexă.
3. T. POPOVICIU, *Considerații teoretice asupra utilizării practice a unor formule de interpolare*. Bul. științ., Secția mat. fiz., 1951, **III**, 4.