Matrix Multiplication Algorithms and Complexity – Ex1

Question 1:

(a) Write a U V W representation of the following ⟨2, 1, 2; 4⟩-algorithm.

We know that $W^T\left(U\vec{A}\odot V\vec{B}\right)=\vec{C}$

From the M_1, M_2, M_3, M_4 we can build the U, V matrices and we get:

$$U=\begin{pmatrix}1 & 0\\ 1 & -1\\ 0 & 1\\ 0 & 1\end{pmatrix} \qquad V=\begin{pmatrix}1 & 1\\ 0 & 1\\ 1 & -1\\ 0 & 1\end{pmatrix}$$

And for W from $c_{11}, c_{12}, c_{21}, c_{22}$ we get:

$$W=\begin{pmatrix}1 & 0 & 0 & 0\\ -1 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ -1 & 1 & 1 & 1\end{pmatrix}$$

(b) What are the dimensions, exponent, and leading coefficient of this matrix multiplication algorithm?

Dimensions:

For all matrices we have 11 rows meaning there are 11 multiplications in this algorithm, and both U, V have 6 columns meaning A, B are either 2×3 or 3×2 each. But W has only 4 columns so that means that A is 2×3 and B is 3×2.

So we get the dimensions $<2{,}3{,}2;11>$.

Exponent:

We have $<2{,}3{,}2;11>$ algorithm so we'll use the dimensions duality to calculate the exponent as follows:

$$<2{,}3{,}2;11>\otimes<2{,}2{,}3;11>\otimes<3{,}2{,}2;11>\to<12{,}12{,}12;11^3>$$

So we get $\log_{12}11^3\cong 2.8949$.

Leading coefficient:

Given the formula:

$$T(n,m,k)=\left(\frac{q_u}{n_0^{\omega_0}-n_0m_0}+\frac{q_v}{n_0^{\omega_0}-m_0k_0}+\frac{q_w}{n_0^{\omega_0}-n_0k_0}+1\right)n^{\log_{n_0}t}-\left(\frac{q_u}{n_0^{\omega_0}-n_0m_0}n\cdot m+\frac{q_v}{n_0^{\omega_0}-m_0k_0}m\cdot k+\frac{q_w}{n_0^{\omega_0}-n_0k_0}n\cdot k\right)$$

We know that $n_0=2\,, M_0=3\,, k_0=2\ \ , \omega_0=2.8949$

So, we only need to find q_v, q_u, q_w as follows:

$$q_u = nn(U) - t_0 \quad , \quad q_v = nn(V) - t_0 \quad , \quad q_w = nn(W) - \#columns\ of\ W$$

We get:

$$q_u = 5, q_v = 5, q_w = 12$$

So overall we get:

$$leading\ coefficient = \left(\frac{5}{11-6} + \frac{5}{11-6} + \frac{12}{11-4} + 1\right) \cong 4.71$$

Question 2:

(a) Prove that there is a ⟨c, b, a; t⟩-algorithm.

Suppose we want to multiply two matrices $A \in R^{c\times b}$ and $B \in R^{b\times a}$ we will use the hint:

$$AB = ((AB)^T)^T = (B^T A^T)^T$$

Now, we will use the given $< a, b, c; t > -algorithm$ we will calculate the multiplication $B^T\ A^T$ using t multiplications. After using the algorithm all we need is to transpose the answer to get AB.

And we got a $< c, b, a; t > -algotihm$.

(b) Prove that there is a ⟨c, a, b; t⟩-algorithm. Hint: Triple product.

Given $< a, b, c; t > -algorithm$ with the corresponding U, V, W that maintain the triple product condition $\sum_{s=1}^{t} U_{s,(i_1,k_1)} V_{s,(k_2,j_1)} W_{s,(i_2,j_2)} = \delta_{i_1,i_2}\delta_{j_1,j_2}\delta_{k_1,k_2}$, we will name $U' = W, V' = U, W' = V$.

Now, we got 3 matrices that correspond to the needed dimensions c, a, b and hold the triple product condition – that means that according to the triple product claim U', V', W' are encoding/decoding matrices of an $< c, a, b; t >$ $-algorithm$.

(c) Conclude that for every permutation (d, e, f) of (a, b, c), there is a ⟨d, e, f; t⟩-algorithm.

We will got over every premutation of (a,b,c):

I. $< a, b, c; t >$ is given.
II. $< c, b, a; t >$ proved in (a).
III. $< c, a, b; t >$ proved in (b).
IV. $< b, c, a; t >$ apply the (b)permutation twice.
V. $< a, c, b; t >$ apply the (a) permutation on (IV).
VI. $< b, a, c; t >$ apply the (a) permutation on (III).

Question 3:

(a) Prove that de-Groote equivalence (slides 8, 9 in the second presentation) is an equivalence relation. That is, show that it is reflexive, symmetric, and transitive.

Reflexive:

Given a triple (U, V, W), we can apply the *identity* version of each transformation:

- For the permutation step we choose $P = I$.
- For the diagonal step we choose $D_1 = D_2 = D_3 = I$.
- For the invertible transformation we choose $X = Y = Z = I$.

All these choices leave (U, V, W) unchanged, so (U, V, W) is de-Groote equivalent to itself. Hence the relation is reflexive.

Symmetric:

The relation is symmetric if whenever $(U, V, W) \sim (U', V', W')$, then also $(U', V', W') \sim (U, V, W)$.

- Each one of the three types of transformations above is invertible:
  1. A permutation matrix P has inverse $P^{-1} = P^T$, which undoes the transformation $(U, V, W) \to (PU, PV, PW)$.
  2. A diagonal transformation with $D_1 D_2 D_3 = I$ can be inverted by the obvious inverse diagonal matrices (again respecting the product equals I).
  3. For invertible X, Y, Z, the map
     $(A, B) \to (XAY, Y^{-1}BZ^{-1}), \quad C' \to X^{-1}C'Z^{-1}$
     Is inverted by using X^{-1}, Y^{-1}, Z^{-1} in the same scheme.

  Thus, if (U', V', W') is obtained from (U, V, W) by one of these transformations, we can always reverse it by the inverse transformation.

  Consequently, whenever $(U, V, W) \sim (U', V', W')$, we can also have $(U', V', W') \sim (U, V, W)$. This proves symmetry.

transitive:

The relation is transitive if whenever $(U, V, W) \sim (U', V', W')$ and $(U', V', W') \sim (U'', V'', W'')$, then $(U, V, W) \sim (U'', V'', W'')$.

The key is that *composing* any of the allowed transformations yields another transformation of the same form (or a finite sequence of them).

1. Composition of permutation matrices is another permutation matrix.

2. Composition of diagonal transformations (subject to the product constraint $D_1 D_2 D_3 = I$) can be merged into a single diagonal transformation.

3. Composition of invertible transformations X, Y, Z with another triple X', Y', Z' simply results in the invertible triple $X'X, Y'Y, Z'Z$.

Hence if (U', V', W') is obtained from (U, V, W) by some finite sequence of these transformations, and (U'', V'', W'') is obtained from (U', V', W') by another finite sequence of transformations, then (U'', V'', W'') is obtained from (U, V, W) by the composition of those two sequences. Therefore the relation is transitive.

Since de-Groote equivalence is **reflexive**, **symmetric**, and **transitive**, it is an **equivalence relation**.

(b) Let (U, V, W) , (U', V', W') be two algorithms in the same De-Groote equivalence class. Prove that if U contains duplicate rows then U' contains duplicate rows up to a multiplication by a constant.

Going from (U, V, W) to (U', V', W') is done by a sequence of 3 moves. So if we check how each individual move affects duplicate rows, we can chain them all together.

1. Multiplying U on the left by a permutation matrix:
   If P is a permutation matrix, then
   $$U \rightarrow PU$$
   Just reorders the rows of U. Concretely:
   - If row i of U was the same as row j of U, then in PU, the rows get permuted accordingly. The new row $p(i)$ will coincide with the new row $p(j)$.

   Hence two identical rows remain two identical rows just possibly at different indices.
2. Multiplying U on the left by a diagonal matrix
   If D_1 is diagonal with all diagonal entries non-zeros, then
   $$U \rightarrow D_1 U$$
   Suppose row i equals to row j in U, after multiplying on the left by D_1, we get:
   - New row $i = (D_1(i,i))(old\ row\ i)$.
   - New row $j = (D_1(j,j))(old\ row\ j)$.

   But since $old\ row\ i = old\ row\ j$, the new row i is just a constant multiple of the new row j, namely
   $$new\ row\ i = \frac{D_1(i,i)}{D_1(j,j)}(new\ row\ j)$$

So, the old two identical rows in U become two rows that coincide *up to* a scalar.

3. Multiplying U on the right by an invertible matrix (X, Y, Z) can be viewed from U's perspective as multiplying U on the right by some invertible matrix. Explicitly:
$$U \to UM \qquad \qquad for\ some\ invertible\ M$$
But if row i and row j of U are the same then:
$$(row\ i)M = (row\ j)M$$
Right multiplication by the same matrix M does not separate them, it preserves the equivalence of row i and j.

Because every step in the sequence of transformations either:

- Reorders the rows (permutation on the left).
- Scales the row by a diagonal factor (diagonal from the left).
- Right multiplies all rows by an invertible matrix.

Hence, if U contains duplicate rows then U' contains duplicate rows up to a multiplication by a constant.

(c) Show that the following matrix multiplication algorithms are not de-Groote equivalent.

We proved in (b) that if U contains duplicate rows then U' contains duplicate rows up to a multiplication by a constant. That means that if U contains duplicate rows but U' doesn't contain duplicate rows up to a multiplication by a constant then (U', V', W') is not De-Groote equivalent to (U, V, W).

As we can see in ALG_1 the first and second rows are identical and there no identical row up to a multiplication by a constant in ALG_2.

Hence, ALG_1 and ALG_2 are not De-Groote equivalent.

(d) Describe the following de-groote transformation as an alternative basis.

נתונים:

$$X = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \quad Y = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad Z = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

נראה כי $Y^{-1} = Y$ כי $Y^2 = I$

וגם $Z^{-1} = Z$

$$X^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$$

נחשב את ϕ:

מכיוון ש- $Y^T = Y$ (Y אלכסונית)

נקבל:

$$\phi = Y^T \otimes X = Y \otimes \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -2 \end{pmatrix}$$

נחשב את ψ:

נזכיר כי $Y^{-1} = Y$, $(Z^{-1})^T = Z^T = Z$

נקבל:

$$\psi = (Z^{-1})^T \otimes Y^{-1} = Z \otimes Y$$

כש- $Z = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $Y = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$

נקבל:

$$Z \otimes Y = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}$$

נחשב את ν:

$$X^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{pmatrix}$$

$$\nu = (Z^{-1})^T \otimes X^{-1} = Z \otimes X^{-1}$$

$$= \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \frac{1}{2} \\ 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \end{pmatrix}$$