

Time Complexity of Matrix Transpose Algorithm using Identity Matrix as Reference Matrix

Sanil Shanker KP¹, Mohammed Shameer MC²

^{1,2}Dept. of Computer Science
Farook College, Kozhikode, India.

Abstract- This paper presents the time complexity of matrix transpose algorithm using identity matrix as reference matrix. We computed the time complexity of the algorithm as $O(mn)$.

Keywords: Identity matrix, Reference matrix, Sanil's Matrix Transpose.

I. INTRODUCTION

Transpose of the matrix can be obtained by combining the characteristics of logical AND (\wedge) with logical OR (\vee) operations [1, 2]. In Sanil's matrix transpose algorithm, the identity matrix acts as the kernel of the transformation [3]. For example, let the matrix $A_{(3 \times 4)}$ be

17	2	13	7
41	11	29	19
19	3	23	11

The transformation can be computed as:

Input: $A_{(3 \times 4)}$ logical AND I_3

17	2	13	7	\wedge	1	0	0
41	11	29	19	\wedge	0	1	0
19	3	23	11	\wedge	0	0	1

↓ ↓ ↓

17	41	19
2	11	3
13	29	23
7	19	11

Output: $A^T_{(4 \times 3)}$

Here, identity matrix acts as the kernel to find the transpose.

II. TIME COMPLEXITY

Let $A_{(m \times n)}$ and $B_{(m \times m)}$ be the input matrix of order $(m \times n)$ and the reference matrix of order $(m \times m)$ respectively. The value of c_{11} can be computed from the Figure- 1, as $c_{11} := (a_{11} * b_{11}) + (a_{21} * b_{21}) + (a_{31} * b_{31})$.

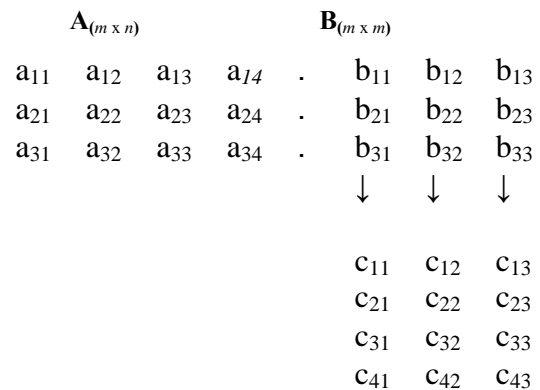


Fig. 1 $A_{(n \times m)}^T := C_{(n \times m)}$

To compute one cell value, there exists 'm' multiplications and 'm-1' additions. For the transformation, $c_{nm} \leftarrow a_{nm}$, the computational time is $O(m)$. If there exists 'm' rows, time will be $O(m) + O(m) + \dots + m$ times = $O(m^2)$. For 'n' columns, the computational time is $O(nm^2)$.

In the case of identity matrix as reference matrix, ($a_i = m, j = n * I_{i=m, j=m}$) exists and other will be zero (Figure- 2) [2]. This implies the time for one multiplication operation will be $O(1)$.

If there exists 'm' rows, time will be $O(1) + \dots + m$ times = $O(m)$. In general, for 'n' columns, time = $O(mn)$.

$$\begin{array}{ccccccc}
 & \mathbf{A}_{(m \times n)} & & & & \mathbf{I}_{(i=m, j=m)} & \\
 \mathbf{a}_{11} & \mathbf{a}_{12} & \mathbf{a}_{13} & \mathbf{a}_{14} & \cdot & \mathbf{I}_{11} & \mathbf{0} & \mathbf{0} \\
 \mathbf{a}_{21} & \mathbf{a}_{22} & \mathbf{a}_{23} & \mathbf{a}_{24} & \cdot & \mathbf{0} & \mathbf{I}_{22} & \mathbf{0} \\
 \mathbf{a}_{31} & \mathbf{a}_{32} & \mathbf{a}_{33} & \mathbf{a}_{34} & \cdot & \mathbf{0} & \mathbf{0} & \mathbf{I}_{33} \\
 & & & & & \downarrow & \downarrow & \downarrow \\
 & & & & & \mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13} \\
 & & & & & \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23} \\
 & & & & & \mathbf{C}_{31} & \mathbf{C}_{32} & \mathbf{C}_{33} \\
 & & & & & \mathbf{C}_{41} & \mathbf{C}_{42} & \mathbf{C}_{43}
 \end{array}$$

Fig. 2 $\mathbf{A}_{(n \times m)}^T := \mathbf{C}_{(n \times m)}$

III. SUMMARY

The computational time of matrix transpose algorithm using identity matrix as reference matrix is $O(mn)$. Suppose, if the given matrix is a square matrix, the running time will be $O(n^2)$.

REFERENCES

- [1] Sanil Shanker KP, *An Algorithm to Transpose Zero- One Matrix*. Int. Journal of Com. Sci. and Inf. Tech, Vol. 7 (4), 2016 , 1960- 1961.
- [2] Sanil Shanker KP, *Will the reference Matrix act as the Kernel of Matrix Transformation?*, Int. Journal of Math. Tre. and Tech, Vol. 36 (1), 2016, 80- 81.
- [3] Mohammed Shameer. MC, Sanil Shanker. K. P, *Java Implementation of Sanil's Matrix Transpose*, Int. Journal of Com. Sci. and Inf. Tech. Vol. 7(5), 2016, 2145- 2146.