DAA Assignment

Nykaj AK 2022bcs36
Harshit 2022bcs0096
Anushila Biswas 2022bcs69
Prathamesh Vasaikar 2022bcs72
Arjun Deshmukh 2022bcs84
Arsh Abel Kumar 2022bcs90
Swayam Poddar 2022bcs105
Y. Nissi Veronika 2022bcs138
Sambhram R. Vijapur 2022bcs165

November 5, 2023

1 Introduction

Our goal is to show that we cannot solve the Maximum Subarray problem on a matrix of size $n \times n$ in time $O(n^{3-\epsilon})$. It is known that an $O(n^{3-\epsilon})$ time algorithm for the Maximum Subarray problem implies an $O(n^{3-\epsilon/10})$ time algorithm for the All-Pairs Shortest Paths problem [**?**]. Combining our reduction with the latter one, we obtain Theorem 2 from the introduction, which we restate here:

Effort: We made attempts, but regrettably, we couldn't formulate a superior solution.

Theorem 2. For any constant $\epsilon > 0$, an $O(n^{3-\epsilon})$ time algorithm for the Maximum Subarray problem on $n \times n$ matrices implies an $O(n^{3-\epsilon/10})$ time algorithm for the All-Pairs Shortest Paths problem.

Clearly, the Negative Triangle problem is equivalent to the Positive Triangle problem. In the remainder of this section, we, therefore, reduce the problem of detecting whether a graph has a positive triangle to the Maximum Subarray problem.

We need the following intermediate problem:

Definition 13 (Maximum 4-Combination). Given a matrix $B \in R^{m \times m}$, output

$$\max_{i,i_0,j,j_0 \in [m]: i \le i_0 \text{ and } j \le j_0} \left(B[i,j] + B[i_0,j_0] - B[i,j_0] - B[i_0,j]\right).$$

Our reduction consists of two steps:

1. Reduce the Positive Triangle problem on an n-vertex graph to the Maximum 4-Combination problem on a $2n \times 2n$ matrix.

2. Reduce the Maximum 4-Combination problem on an $n \times n$ matrix to the Maximum Subarray matrix of size $n \times n$.

1.1 Positive Triangle $\Rightarrow$ Maximum 4-Combination

Let A be the weighted adjacency matrix of size $n \times n$ of the graph, and let M be the largest absolute value of an entry in A. Let $M_0 := 10M$ and $M_{00} := 100M$. We define matrix $D \in \mathrm{R}^{n \times n}$:

$$D_{i,j} = \begin{cases} M_0 + M_{00} & \text{if } i = j, \\ M_{00} & \text{otherwise.} \end{cases}$$

We define matrix $B \in \mathrm{R}^{2n \times 2n}$ as:

$$B = \begin{pmatrix} A - A^T & -A^T D \end{pmatrix}$$

The reduction follows from the following lemma:

Lemma 14. Let X be the weight of the max-weight triangle in the graph corresponding to the adjacency matrix A. Let Y be the output of the Maximum 4-Combination algorithm when run on matrix B. The following equality holds:

$$Y = X + M_0 + M_{00}.$$

Proof. Consider integers i, j, i_0, j_0 that achieve a maximum in the Maximum 4-Combination instance as per Definition 13. Our first claim is that $i, j \leq n$ and $i_0, j_0 \geq n + 1$. If this is not true, we do not collect the weight M_{00}, and the largest output that we can get is $\leq 4M_0 \leq 9M_{00}/10$. Note that we can easily achieve a larger output with $i = j = 1$ and $i_0 = j_0 = n + 1$.

Our second claim is that $i_0 = j_0$. If this is not so, we do not collect the weight M_0, and the largest output that we can get is $M_{00} + 4M \leq M_{00} + M_0/2$. Note that we can easily achieve a larger output with $i = j = 1$ and $i_0 = j_0 = n + 1$. Thus, we can denote $i_0 = j_0 = k + n$.

Now, by the construction of B, we have:

$$B[i, j] + B[i_0, j_0] - B[i, j_0] - B[i_0, j] = A[i, j] + A[j, k] + A[k, i] + M_0 + M_{00}.$$

We get the equality we need.

2 Conclusion

In this assignment, we have explored the reduction of the Maximum Subarray problem in a 2D matrix to the Negative Triangle problem in graph theory. Our goal was to demonstrate that we cannot solve the Maximum Subarray problem in a matrix of size $n \times n$ in time $O(n^{3-\epsilon})$.

We presented a reduction involving two key steps: first, the reduction of the Positive Triangle problem to the Maximum 4-Combination problem, and then the reduction of the Maximum 4-Combination problem to the Maximum Subarray problem. Our reduction was based on careful construction of matrices and the analysis of maximum weights in various contexts.

3 References

Arturs Backurs, Nishanth Dikkala, and Christos Tzamos. Tight hardness results for maximum weight rectangles. arXiv preprint arXiv:1602.05837, 2016. URL: http://arxiv.org/abs/1602.05837