COLLEGE OF ARTS AND SCIENCES
UNIVERSITY OF THE PHILIPPINES LOS BAÑOS
1/F Wing C Physical Sciences Bldg., Harold Cuzner Royal Palm Ave.
College 4031, Laguna, Philippines
Phone +63 49 536 2313
ics.uplb@up.edu.ph www.ics.uplb.edu.ph
INSTITUTE OF COMPUTER SCIENCE

CMSC 180: Introduction to Parallel Computing
Second Semester 2022-2023

Runtime-Efficient Serial Implementation of Interpolating the elevations into a higher resolution digital elevation matrix M given a lower resolution digital elevation matrix N

Ariel Doria

Introduction

After discussing the iterative real-world computational problem, we will discuss how we can efficiently run the program.

Contents

Learning Objectives

At the end of this discussion, the student must be able to:

- Improve their written program to run it efficiently; and
- Understand how can we compute the complexity of the written program.

Discussion

Improvements in Elevation Interpolation Matrix

Federal Communications Commission (FCC) Interpolation

In improving the runtime of this method, we can use the idea of getting first the interpolated elevation that will be used iteratively such as those interpolated elevation along the bounding box that will be used over and over again in interpolating elevation of those points inside the bounding box.

Improvement FCC Method

```
//Interpolate Elevation of points within the Bounding Box
    for(i = 0; i<size; i++){
        for (j = 0; j<size; j++){
            if(!(i%10 ==0 && j%10==0)){
```

```
                if(i%10 ==0 || j%10 == 0){
                    M[i][j] = y1 + ((i-x1)/(x2-x1))* (y2-y1);
                }
            }
        }
    }

//Interpolate Elevation of points inside the Bounding Box
    for(i = 0; i<size; i++){
        for (j = 0; j<size; j++){
            if(!(i%10 ==0 && j%10==0)){
                if(!(i%10 ==0 || j%10 == 0)){
                    M[i][j] = y1 + ((i-x1)/(x2-x1))*(y2-y1);
                }
            }
        }
    }
```

Area Weighted Interpolation

Since the elevation in area weighted interpolation is interpolated independently from other points, memory caching can be an option for improvement. For memory caching, you can store the bounding points that are used iteratively within the bounding box

Area Weighted Method

```
//Interpolate Elevation of points within the Bounding Box
    for(i = 0; i<size; i++){
        for (j = 0; j<size; j++){
            if(!(i%10 ==0 && j%10==0)){
                M[i][j] = ((a*A) + (b*B) + (c*C) + (d*D)) / 100
            }
        }
    }
```

Complexity Analysis

The analysis of written programs can be analyzed using theoretical complexity. Let's try to use this pseudocode of interpolating the elevations into a higher resolution digital elevation matrix M given a lower resolution digital elevation matrix N using the FCC Method.

```
//Interpolate Elevation of points within the Bounding Box
    for(i = 0; i<size; i++){
        for (j = 0; j<size; j++){
            if(!(i%10 ==0 && j%10==0)){
                if(i%10 ==0 || j%10 == 0){
                    M[i][j] = y1 + ((i-x1)/(x2-x1))* (y2-y1);
                }
```

1/F Wing C Physical Sciences Bldg., Harold Cuzner Royal Palm Ave.
College 4031, Laguna, Philippines
Phone +63 49 536 2313
ics.uplb@up.edu.ph www.ics.uplb.edu.ph


```
            }
        }
    }

//Interpolate Elevation of points inside the Bounding Box
    for(i = 0; i<size; i++){
        for (j = 0; j<size; j++){
            if(!(i%10 ==0 && j%10==0)){
                if(!(i%10 ==0 || j%10 == 0)){
                    M[i][j] = y1 + ((i-x1)/(x2-x1))*(y2-y1);
                }
            }
        }
    }
```

It can easily be inferred from this piece of code that it takes $O(N^2)$ steps to compute a matrix product because of the two nested for-loops. Since there are two separated two nested loops, it will lead to a time complexity of $2*O(N^2)$. But note that your code complexity depends on how you code the program.

If we just used Area Weighted, the time complexity is about $O(N^2)$ because of two nested loops.

```
//Interpolate Elevation of points within the Bounding Box
    for(i = 0; i<size; i++){
        for (j = 0; j<size; j++){
            if(!(i%10 ==0 && j%10==0)){
                M[i][j] = ((a*A) + (b*B) + (c*C) + (d*D)) / 100
            }
        }
    }
```

There are also other operations executed such as reading and writing to memory. In line with memory, the required space needed to store both input and output matrices for both method take also $O(N^2)$ space.

References

- Poserio, C. (2020). Week 1: Review on Serial Programming. CMSC 180 Laboratory Handouts.