Università
Ca'Foscari
Venezia

LEARNING WITH MASSIVE DATA
CM 0622-1

Assignment 1

Student:

Giosuè Zannini 873810

Academic year 2022/2023

Introduction

In this first assignment I had to count the number of triangles in an indirect graphs using threads. As dataset I used the graphs available at the following link.

The hardware used to perform this task is as follows (compiled with option -O3 -fopenmp):

- Processor: AMD Ryzen 5 3500U with Radeon Vega Gfx (8CPUs) ~2.1GHz.
- RAM: 8GB.

I have made the following assumptions to perform this task:

- The edges are ordered.
- My sequential algorithm is the best.

The data sets that I used are the following:

Graph name	Nodes	Edges
Orkut social network	3072441	117185083
Facebook network	4039	88234
DBLP collaboration network	317080	1049866
Amazon product co-purchasing network	334863	925872
Autonomous systems by Skitter network	1696415	11095298
Youtube social network	1134890	2987624

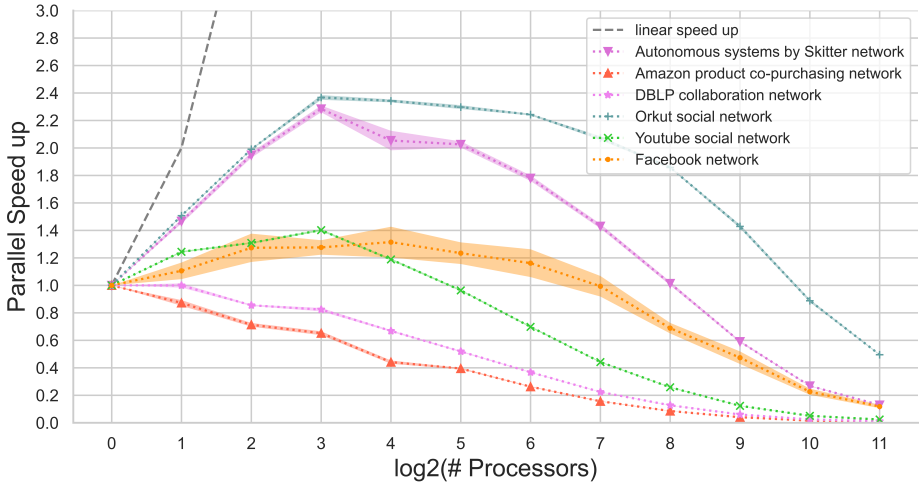
My idea for the implementation is to use CSR as matrix compression and also for each row of the same I don't consider the last column because the set of the first node will be empty and with the same idea I don't consider the last row. As for the parallelization I decided to implement the strategy V view in pdf 6 of the course using a dynamic schedule since not all rows have the same size and for this reason a static schedule would not be a good idea.

Performance discussion

In this section I want to check the change in performance by changing the number of threads ($2^i,\ i = 0, \ldots, 11$) used for each individual data set shown in the table above. Considering Amdahl's law the maximum speed up attainable by the various data sets is as follows:

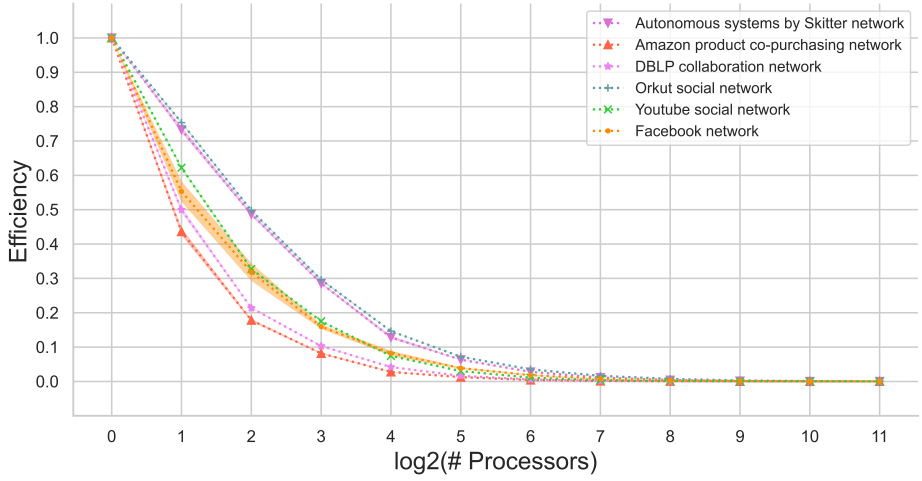
Graph name	Max S
Orkut social network	4.686
Facebook network	2.247
DBLP collaboration network	1.408
Amazon product co-purchasing network	1.333
Autonomous systems by Skitter network	6.194
Youtube social network	2.844

The first graph I want to show is the one related to speed up.



From this graph I can see that all S are less than p with $p > 1$ due to overhead and also for most of the known graphs that there is an uptrend up to 8 threads that is then followed by a general downtrend due to the fact that the machine on which I performed these tests presents a total of 8 CPU. The worst speed up is given by Amazon network and the second worst is given by DBLP network being that the maximum speed up using the Amdahl's law indicates that the time spent to non parallel task is greater than 70%, for this reason there is no sense to parallelize. The two graph that have the highest speed up are Orkut and Skitter network since that the time spent to non parallel task is less than 25%.

The second graph that I want to show is the one related to efficiency.



From this graph I can see that as the number of threads increases the percentage of resources used decreases as the overhead continues to rise.