Homework 4 - Mining Massive Data

Gabriele Giacobazzi

1 Code README

- The .py file and the graphs file need to be located in the same folder
- It is possible to change graph file and algorithm type using the two variables *graph_name* and *algo_type*:
  - **graph_name:** "ants" / "erdos"
  - **algo_type:** "True" / "False" (Asynchronous - Synchronous respectively)
- It is possible to run only one of the assignments commenting the lines using "#"

  `e.g: #hierarchical_clus(graph_name, nodes_list, graph)`

```
if __name__ ==  '__main__':
    graph_name = "ants"
    algo_type = False
    nodes_list, graph = graph_setup(graph_name)
    plt.show()

    # First assignment
    community_update(nodes_list, algo_type)

    # Second assignment
    hierarchical_clus(graph_name, nodes_list, graph)
```

Figure 1: Main method

2 First assignment: Community update

- **Async-Ants:** using the asynchronous method the community algorithm always terminates at the end of the maximum selected iteration number

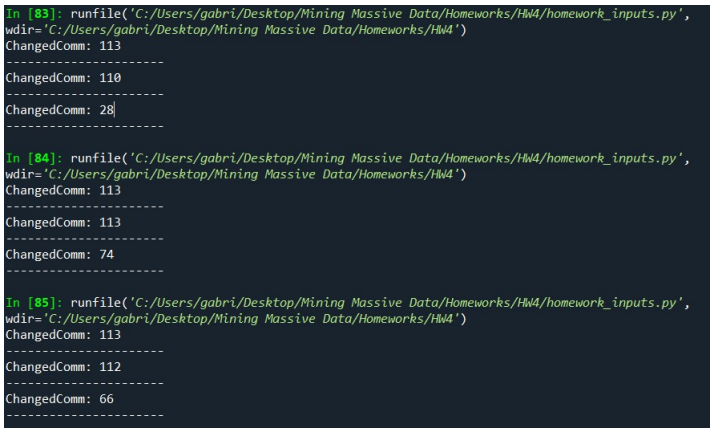

Figure 2: Async - Ants

- **Sync-Ants:** using the synchronous method the community algorithm terminates before the selected iteration number

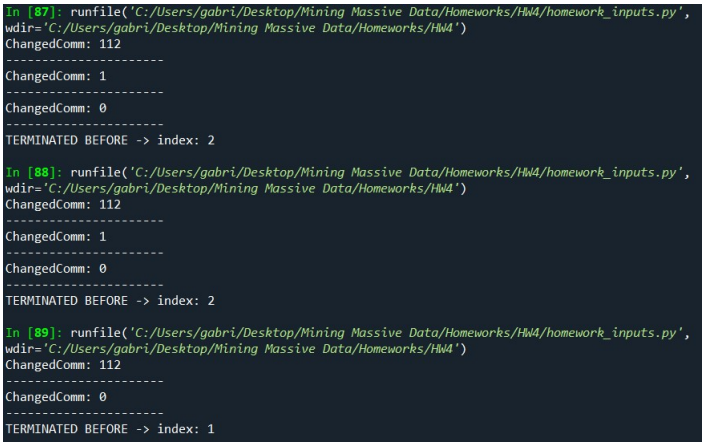

Figure 3: Sync - Ants

- **Async-Erdos:** using the asynchronous method the community algorithm always terminates at the end of the maximum selected iteration number

```
In [90]: runfile('C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4/homework_inputs.py',
wdir='C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4')
ChangedComm: 609
----------------------
ChangedComm: 609
----------------------
ChangedComm: 609
----------------------

In [91]: runfile('C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4/homework_inputs.py',
wdir='C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4')
ChangedComm: 609
----------------------
ChangedComm: 609
----------------------
ChangedComm: 609
----------------------

In [92]: runfile('C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4/homework_inputs.py',
wdir='C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4')
ChangedComm: 609
----------------------
ChangedComm: 609
----------------------
ChangedComm: 609
----------------------
```

Figure 4: Async - Erdos

- **Sync-Erdos:** using the asynchronous method the community algorithm always terminates at the end of the maximum selected iteration number, but the number of changed communities decreases quickly

```
In [93]: runfile('C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4/homework_inputs.py',
wdir='C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4')
ChangedComm: 421
----------------------
ChangedComm: 95
----------------------
ChangedComm: 38
----------------------

In [94]: runfile('C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4/homework_inputs.py',
wdir='C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4')
ChangedComm: 418
----------------------
ChangedComm: 90
----------------------
ChangedComm: 46
----------------------

In [95]: runfile('C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4/homework_inputs.py',
wdir='C:/Users/gabri/Desktop/Mining Massive Data/Homeworks/HW4')
ChangedComm: 421
----------------------
ChangedComm: 98
----------------------
ChangedComm: 47
----------------------
```

Figure 5: Sync - Erdos

3 Second assignment: Hierarchical Clustering

- The shortest path algorithm used is Dijkstra's path length
- The centroids of the initial clusters are picked in a uniformly random way
- If there is no path between two nodes a high number is assigned to the shortest path distance to continue the execution of the algorithm
- For the Erdos graph I tested only using chunks of clusters, otherwise, the computational time would be really large due to the calculation of the distance matrixes, but the results are similar to the Ants ones (a single cluster is obtained in the end)
- It is possible to modify the size of the chunks for the Erdos graph, changing the attribute *processes*. The processes number is used to divide the initial clusters into x clusters

```
if big_graph:
    processes = 100
    chunk_size = int(len(clusters) / processes)
    cluster_chunks = chunks(clusters, chunk_size)
    for clus in cluster_chunks:
        clusters = clus
        break
```

Figure 6: Chunk size

- **Async-Ants:** if an asynchronous community update method is used before, the hierarchical clustering phase may change some clusters in the Ants graph

```
Clusters num before hierarchical clustering: 4
Picking centroids
Populating clusters
Starting merging phase
Computing matr dist of Ant156 1/4
Finished computing
Computing matr dist of Ant43 2/4
Finished computing
Computing matr dist of Ant215 3/4
Finished computing
Computing matr dist of Ant159 4/4
Finished computing
Merging Ant159 Ant156
Source nodes: 7
Target nodes: 84
Merged cluster len: 92
Computing matr dist of Ant43 1/3
Finished computing
Computing matr dist of Ant215 2/3
Finished computing
Computing matr dist of Ant159 3/3
Finished computing
Merging Ant159 Ant215
Source nodes: 92
Target nodes: 7
Merged cluster len: 100
Computing matr dist of Ant43 1/2
Finished computing
Computing matr dist of Ant159 2/2
Finished computing
Computing matr dist of Ant43 1/2
```

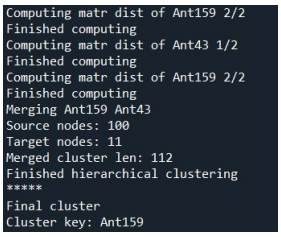

Figure 7: Async - Hierarchical - Ants

- **Sync-Ants:** if a synchronous community update method is used before, the hierarchical clustering phase doesn't change any clusters in the Ants graph

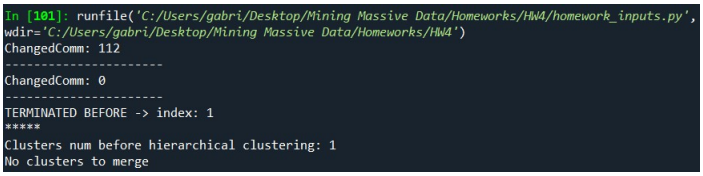

Figure 8: Sync - Hierarchical - Ants

- **Async-Erdos:**

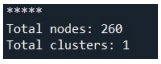

Figure 9: Async - Hierarchical - Erdos

- **Sync-Erdos:**

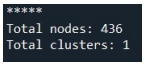

Figure 10: Sync - Hierarchical - Erdos