Method for Overlapping Community Detection in Networks

Alexander Ponomarenko

National Research University Higher School of Economics
Laboratory of Algorithm and Technologies for Network Analysis

Problem statement

Let $G(V,E)$ is a graph with the set of n nodes $V = 1, 2, ..., n$ and set of m edges $E \subset V \times V$. Needs to build a *cover* $C = \{C_1, C_2, ..., C_k\}$, and matrix of *belonging factor* $A = (a_{ic})_{i=1,c=1}^{n,k}$, where is k is the number of clusters,

$$0 \leq a_{ic} \leq 1 \; \forall i \in V, \forall c \in C \tag{1}$$

and,

$$\sum_{c=1}^{k} a_{ic} = 1 \tag{2}$$

The proposed method has the following steps:

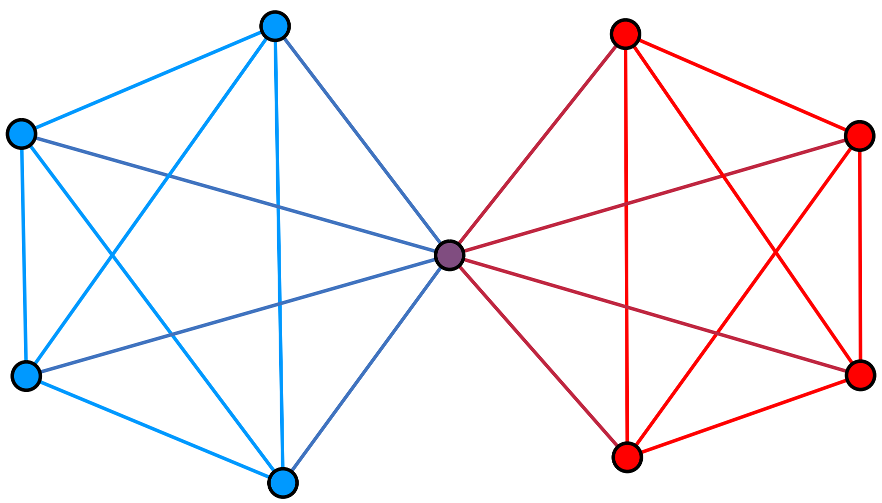
1. Building liner graph $L(G)$
2. Find k disjoint communities.
   (a) Compute the distance matrix between each pair of nodes based on the structure of $L(G)$
   (b) Solve the P-median problem with the P equals to the given number of clusters.
3. Build covering for the original graph G

Link partitioning approach

Let $D = (d_{ij})_{i=1,j=1}^{m,m}$ is a distance matrix defined on the set of edges

We calculate the belonging factor of node i to cluster c as

$$a_{ic} = \frac{\sum_{(i,j)\in E} x_{jc}}{|N_G(i)|}$$



Partitioning around medoids

Let $D = (d_{ij})_{i=1,j=1}^{m,m}$ is a distance matrix defined on the set of edges

Centers of the clusters is a set of k vertices of line graph $L(G)$

$$S = \{s_1, s_2, ..., s_k\}$$

$$\sum_{c=1}^{k} d_{jc} x_{jc}, j \in E \to \min, \tag{3}$$

$$x_{jc} = \begin{cases} 1, & \text{if } d_{jc} \leq d_{js},\ s \in S, \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

[Kaufman, L., & Rousseeuw, P. (1987). *Clustering by means of medoids*. North-Holland.]

Partitioning around medoids

Let $D = (d_{ij})_{i=1,j=1}^{m,m}$ is a distance matrix defined on the set of edges

Centers of the clusters is a set of k vertices of line graph $L(G)$

$$S = \{s_1, s_2, ..., s_k\}$$

$$\sum_{c=1}^{k} d_{jc}x_{jc}, j \in E \rightarrow \min, \tag{3}$$

$$x_{jc} = \begin{cases} 1, & \text{if } d_{jc} \leq d_{js},\ s \in S, \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

p-median problem also known as *facility location problem*

We solve p-median problem exactly with LP_solve by using efficient model of Goldengorin

[AlBdaiwi, B. F., Ghosh, D., & Goldengorin, B. (2011). Data aggregation for p-median problems. *Journal of Combinatorial Optimization*, *21*(3), 348-363.]

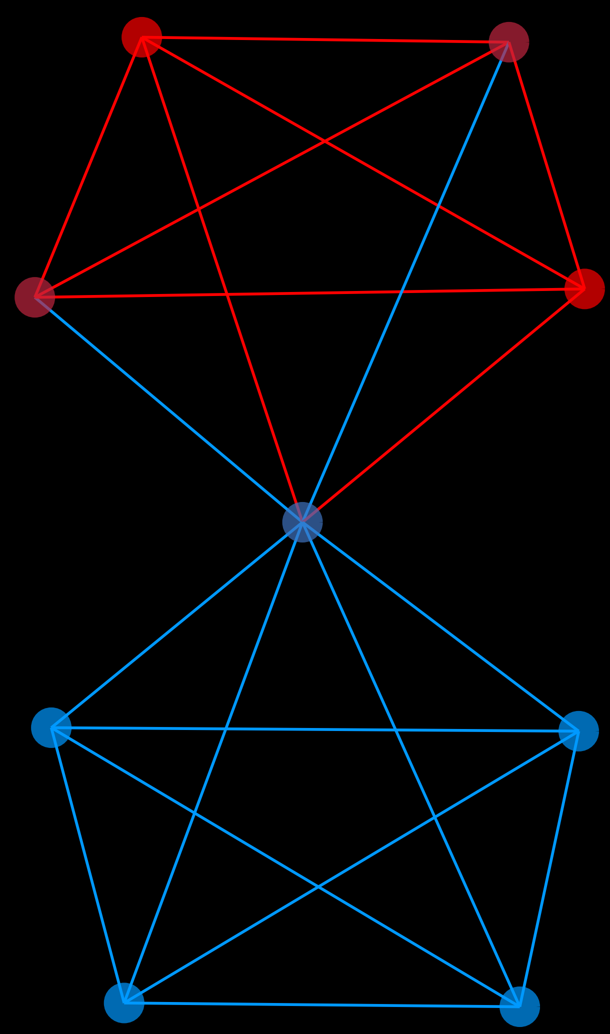
Distance functions

- Shortest path distance
  [Floyd, R. W. (1962). Algorithm 97: shortest path. *Communications of the ACM*, *5*(6), 345]

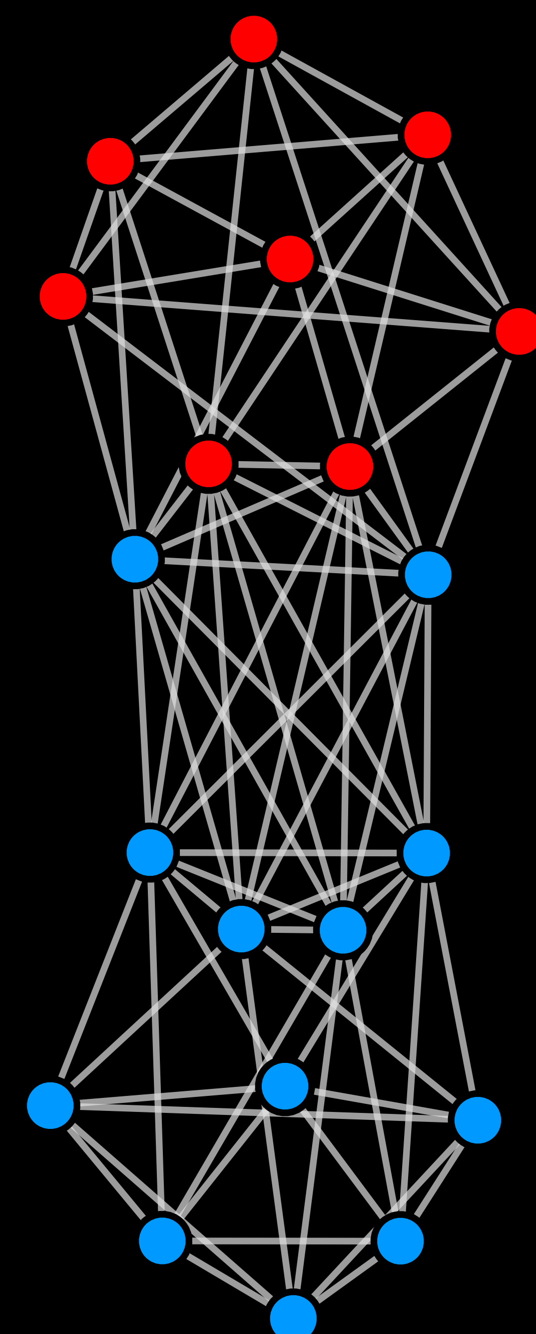
- Commute distance
  [Yen, L., Vanvyve, D., Wouters, F., Fouss, F., Verleysen, M., & Saerens, M. (2005). clustering using a random walk based distance measure. In *ESANN* (pp. 317-324)]

- Amplified commute distance
  [Luxburg, U. V., Radl, A., & Hein, M. (2010). Getting lost in space: Large sample analysis of the resistance distance. In *Advances in Neural Information Processing Systems* (pp. 2622-2630)]

Distance: Shortest path
Number of Clusters: 2

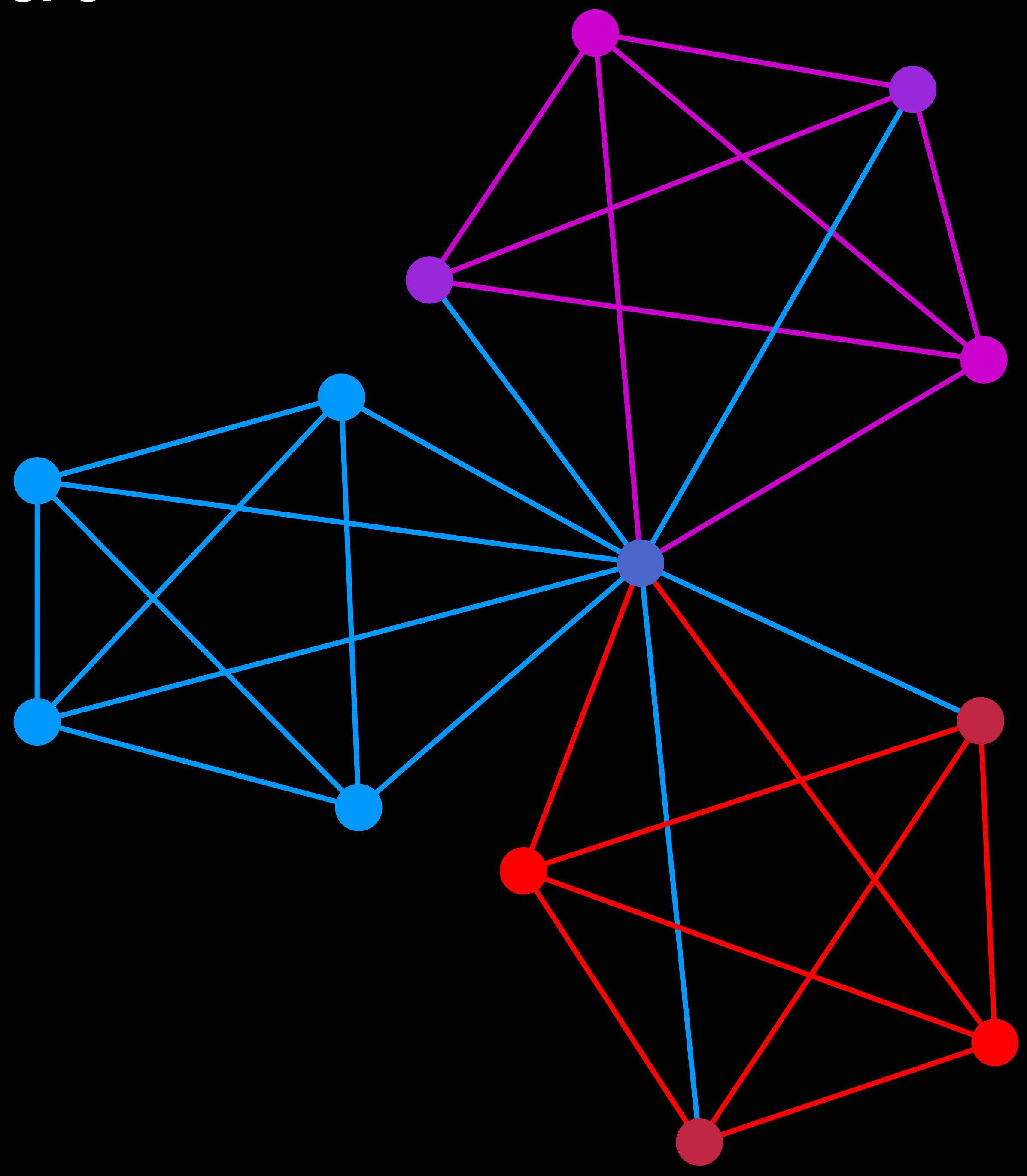


Original Graph



Line Graph

Distance: Shortest path
Number of Clusters: 3



Commute distance

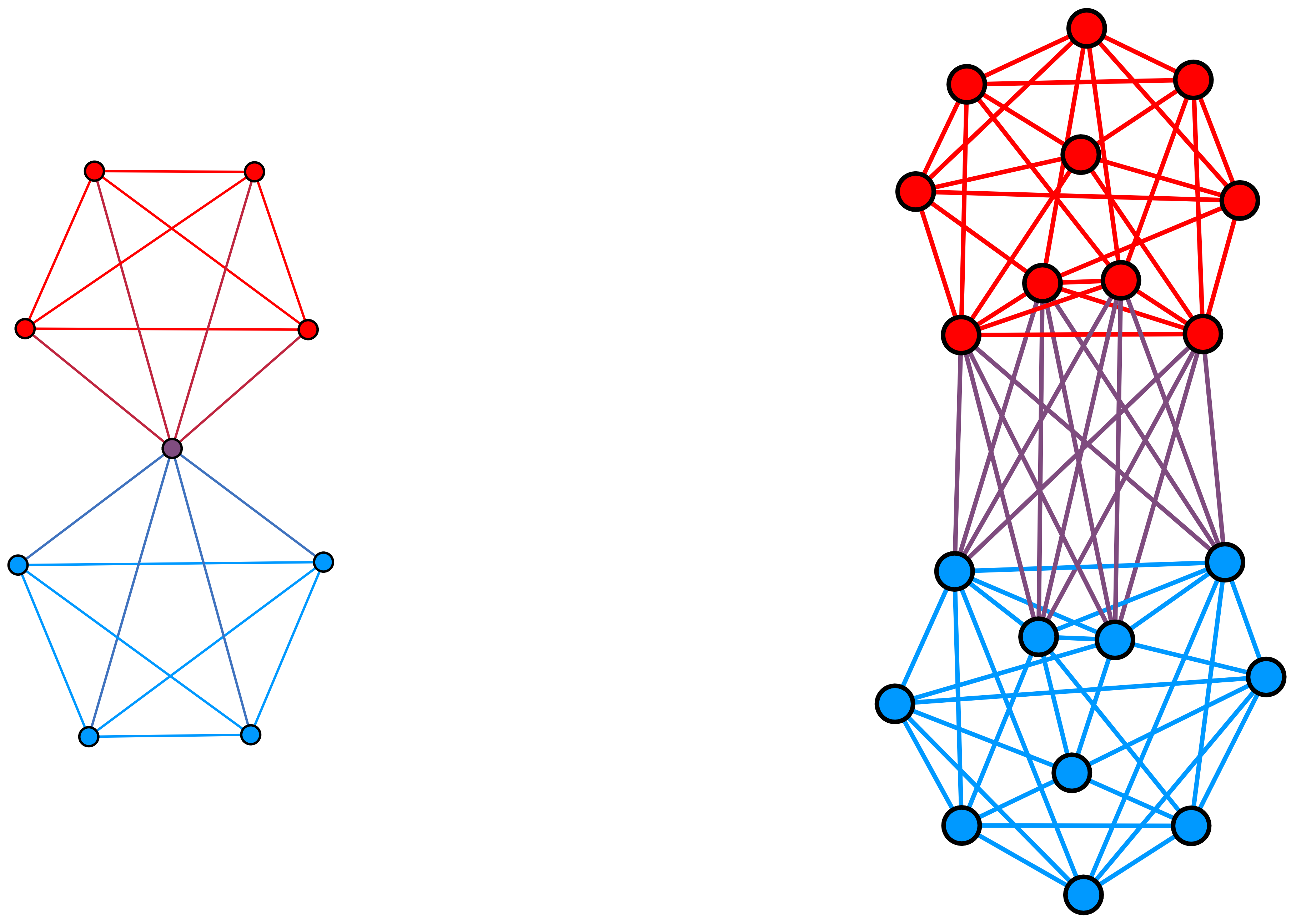
$$\text{Commute distance is } C_{ij} := H_{ij} + H_{ji}$$

where H_{ij} is a hitting time, defined as the expected time for a random walk starting in vertex v_i to travel to vertex to v_j

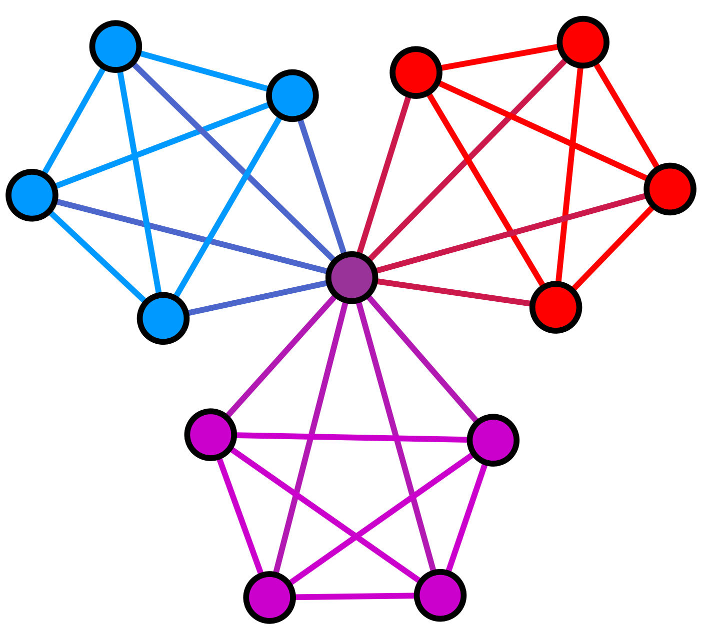
A nice property: it becomes smaller when the number of path are increasing

[Yen, L., Vanvyve, D., Wouters, F., Fouss, F., Verleysen, M., & Saerens, M. (2005). clustering using a random walk based distance measure. In *ESANN* (pp. 317-324)]

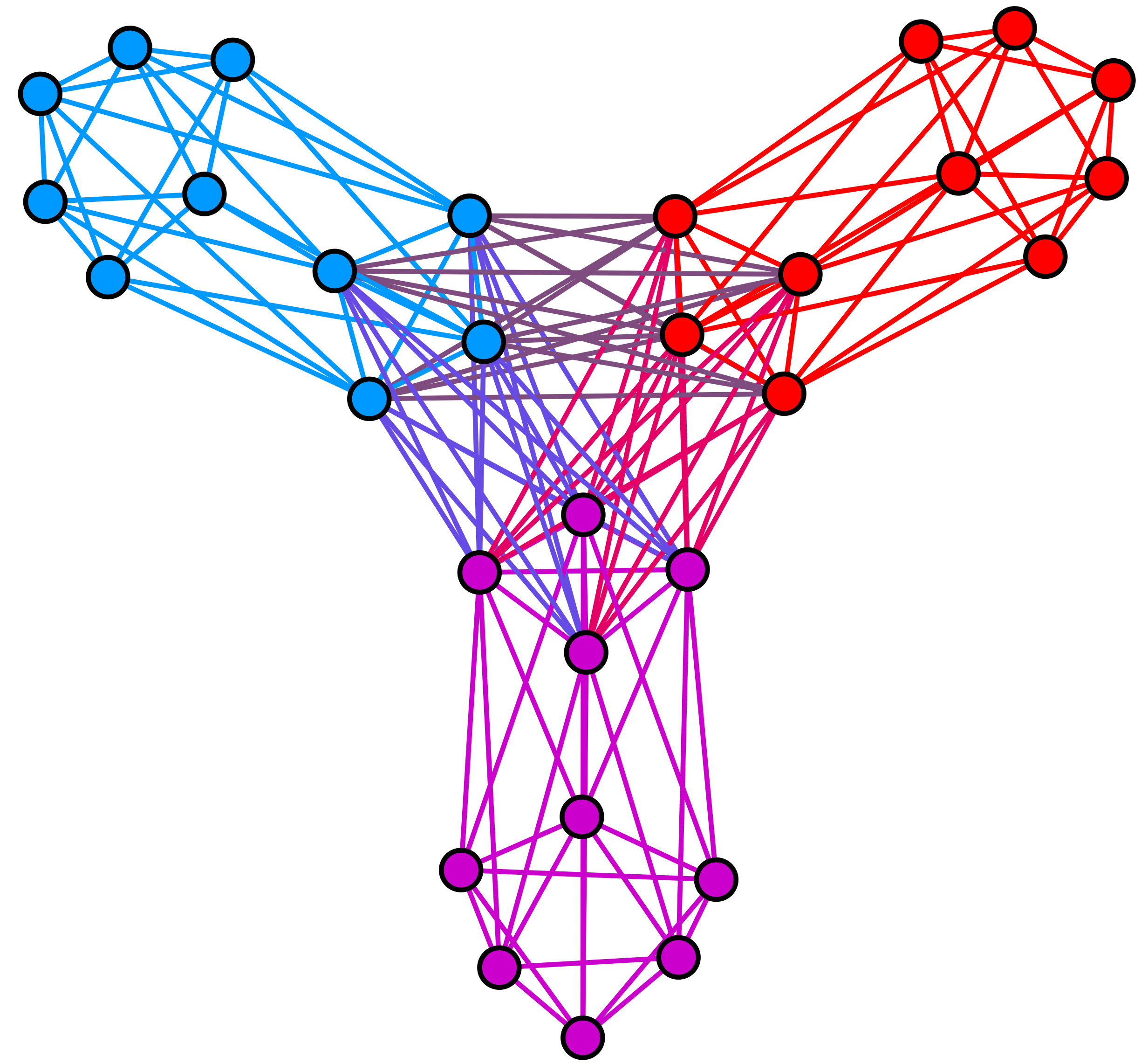
Distance: Commute Distance
Number of clusters: 2 Clusters



Distance: Commute Distance
Number of clusters: 3 Clusters



Original graph



Line graph

Compared methods

- Label Propagation Method
  [Raghavan, U. N., Albert, R., & Kumara, S. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, *76*(3), 036106]
- Modularity optimisation with simulated annealing
  [Sales-Pardo, M., Guimera, R., Moreira, A. A., & Amaral, L. A. N. (2007). Extracting the hierarchical organization of complex systems. *Proceedings of the National Academy of Sciences*, *104*(39), 15224-15229]
- Clique percolation method
  [Palla, G., Derényi, I., Farkas, I., & Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks in nature and society. *nature*, *435*(7043), 814.]
- OSLOM
  [Lancichinetti, A., Radicchi, F., Ramasco, J. J., & Fortunato, S. (2011). Finding statistically significant communities in networks. *PloS one*, *6*(4), e18961.]
- Louvain
  [Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment, 2008(10), P10008]

[Collins, L. M., & Dent, C. W. (1988). Omega: A general formulation of the rand index of cluster recovery suitable for non-disjoint solutions. *Multivariate Behavioral Research*, *23*(2), 231-242.]

Comparing with the ground truth

	LPM	SA	Louvain	CPM	HI	OSLOM	Comm ute dist. PMP	Shortest path PMP
Overlapping	-	-	-	+	-	+	+	+
Zachary (34, 78)	0.70	0.54	0.49	0.09	0.7	0.93	0.5	0.29
Word adj (112, 425)	-0.006	-0.01	-0.01	-0.001	-0.003	0	0.001	0.001
Pol. Books (105, 441)	0.60	0.59	0.31	0.52	0.58	0.60	0.51	0.30
Football (115, 613)	0.85	0.81	0.89	0.06	0.9	0.076	0.12	-
(20, 33)	0.057	0.007	0.04	0.04	0.007	0	-0.005	0.06
(30, 42)	0.63	0.65	0.55	0.59	0.63	0	0.62	0.66
(40, 124)	0.66	0.66	0.7	0.78	0.66	0.58	0.41	0.30
(50, 125)	0.73	0.74	0.78	0.82	0.76	0.7	0.36	0.07
(80, 265)	0.61	0.58	0.64	0.56	0.63	0.3	0.20	0.20
(100, 221)	0.34	0.45	0.41	0.28	0.51	0.17	0.19	0.36
(120, 293)	0.53	0.60	0.50	0.38	0.56	0.45	0.32	0.36
(150, 593)	0.60	0.56	0.54	0.57	0.53	0.15	0.15	0.2
(200, 534)	0.69	0.60	0.77	0.63	0.72	0.41	0.41	0.4

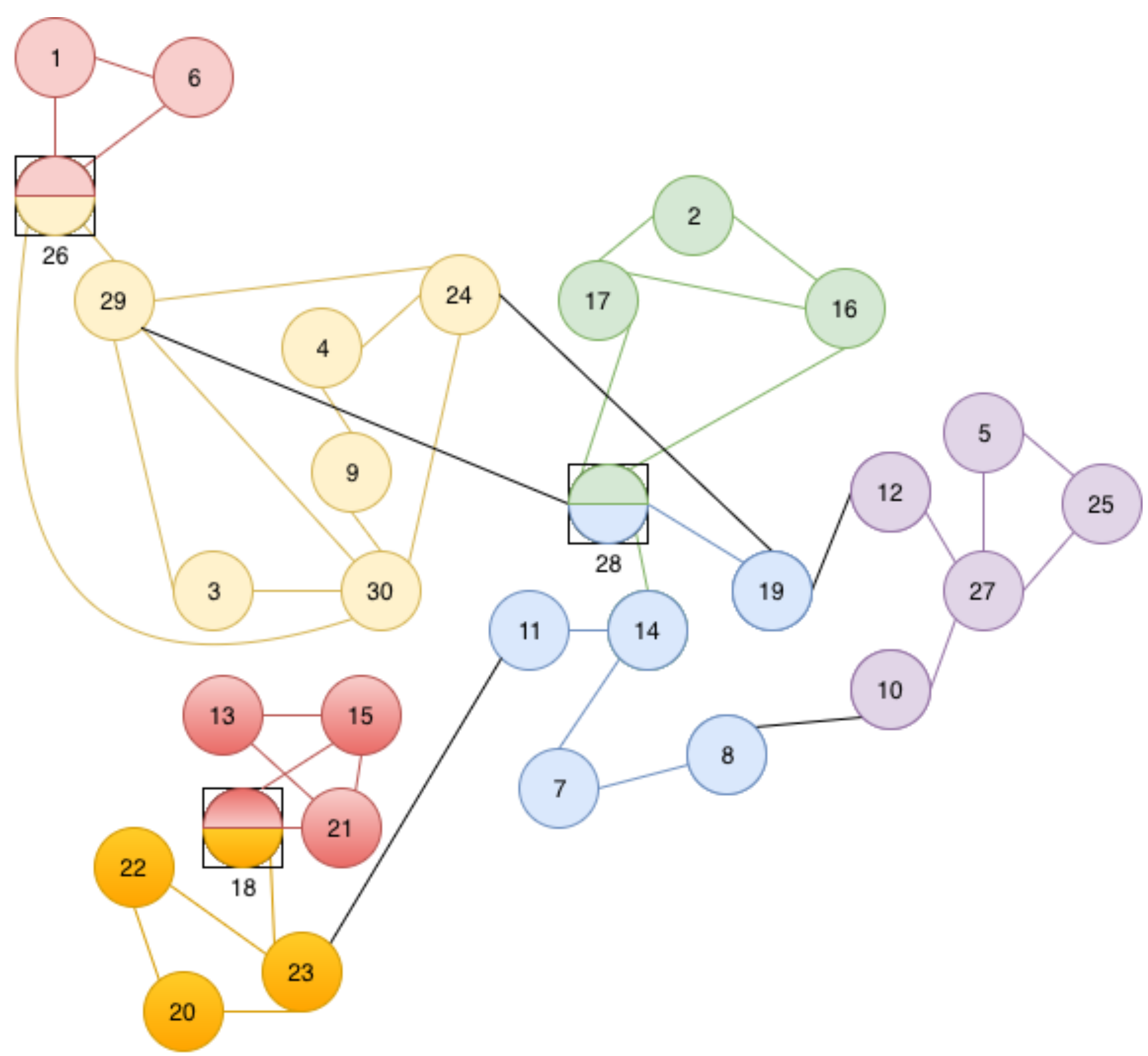
Omega index results

(. , .) graphs was generated by benchmark graphs generating tool

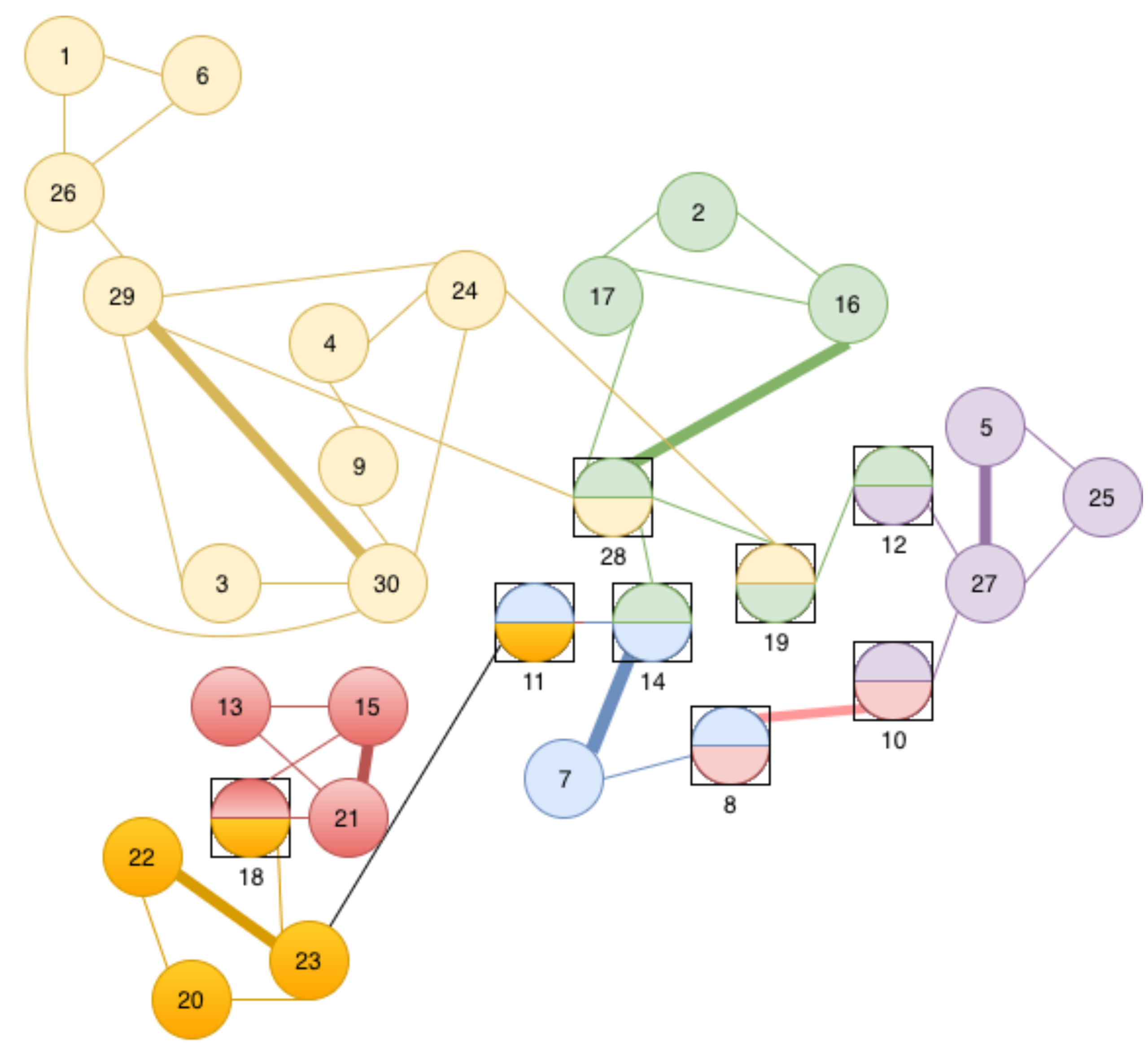
[Lancichinetti, A., Fortunato, S., & Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. *Physical review E*, *78*(4), 046110]

[Collins, L. M., & Dent, C. W. (1988). Omega: A general formulation of the rand index of cluster recovery suitable for non-disjoint solutions. *Multivariate Behavioral Research*, *23*(2), 231-242.]

Distance: Commute Distance
Number of clusters: 6 Clusters



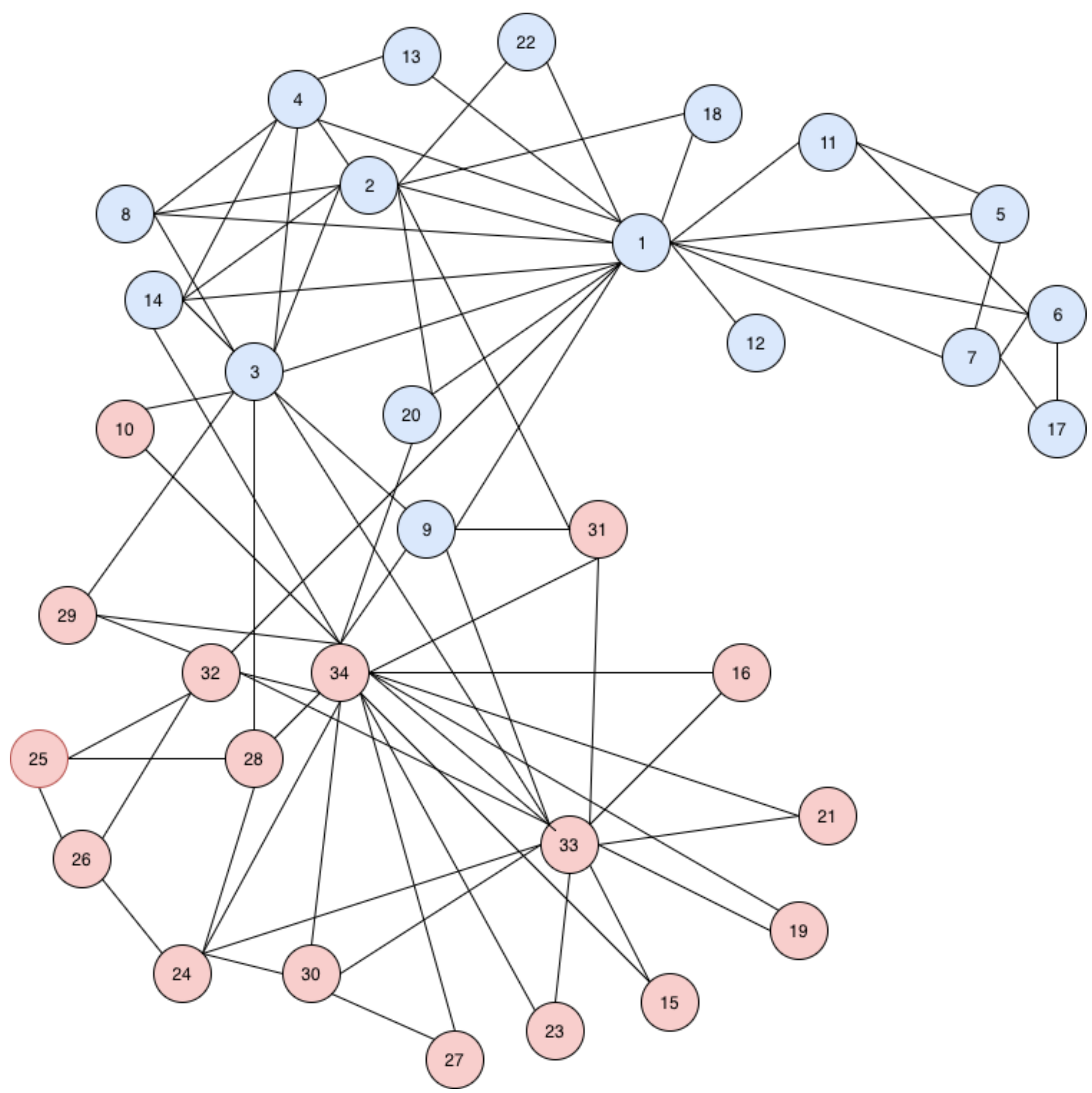
ground truth



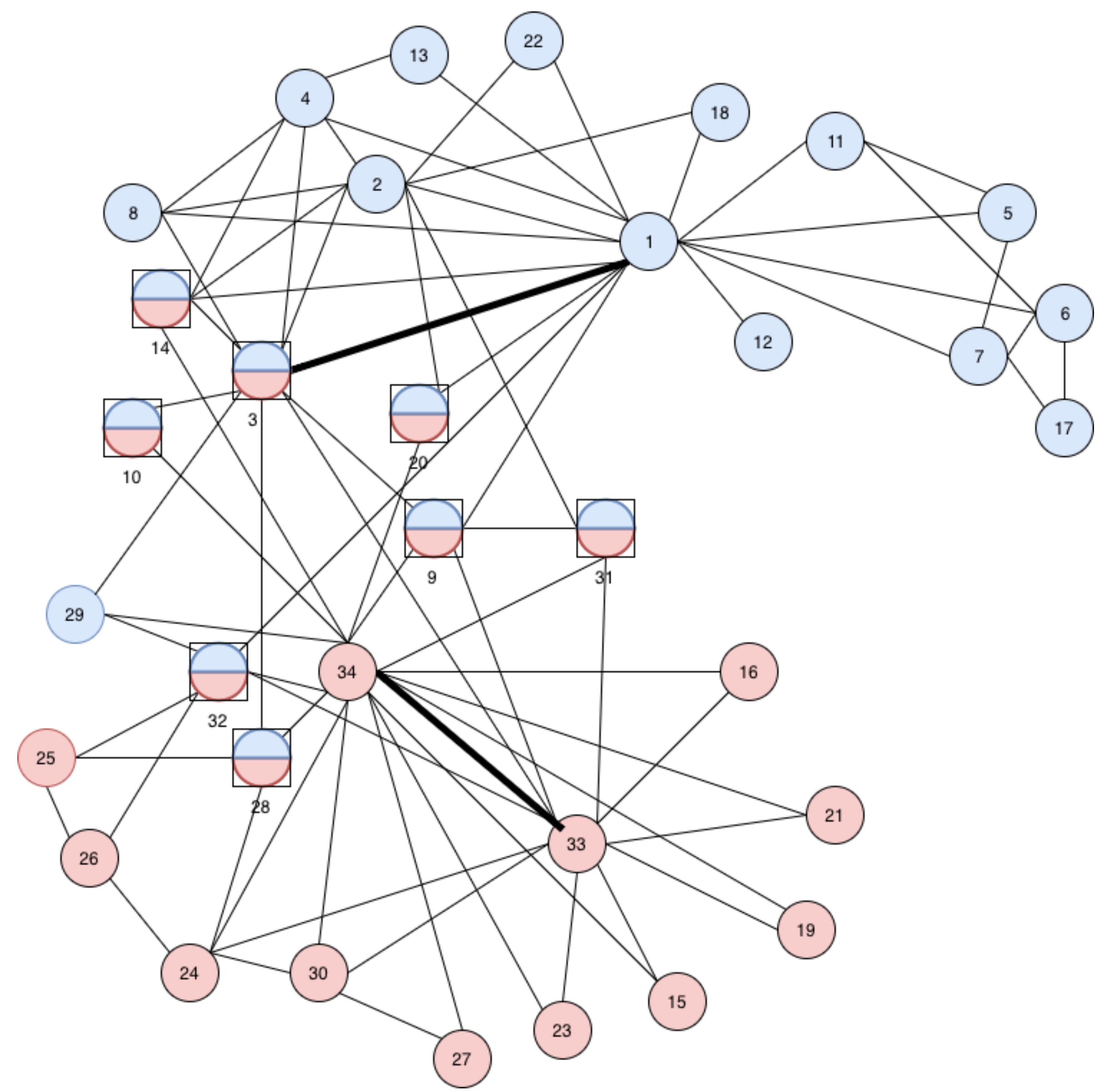
method output

Distance: Commute Distance
Number of clusters: 6 Clusters

Zachary Karate Club



ground truth

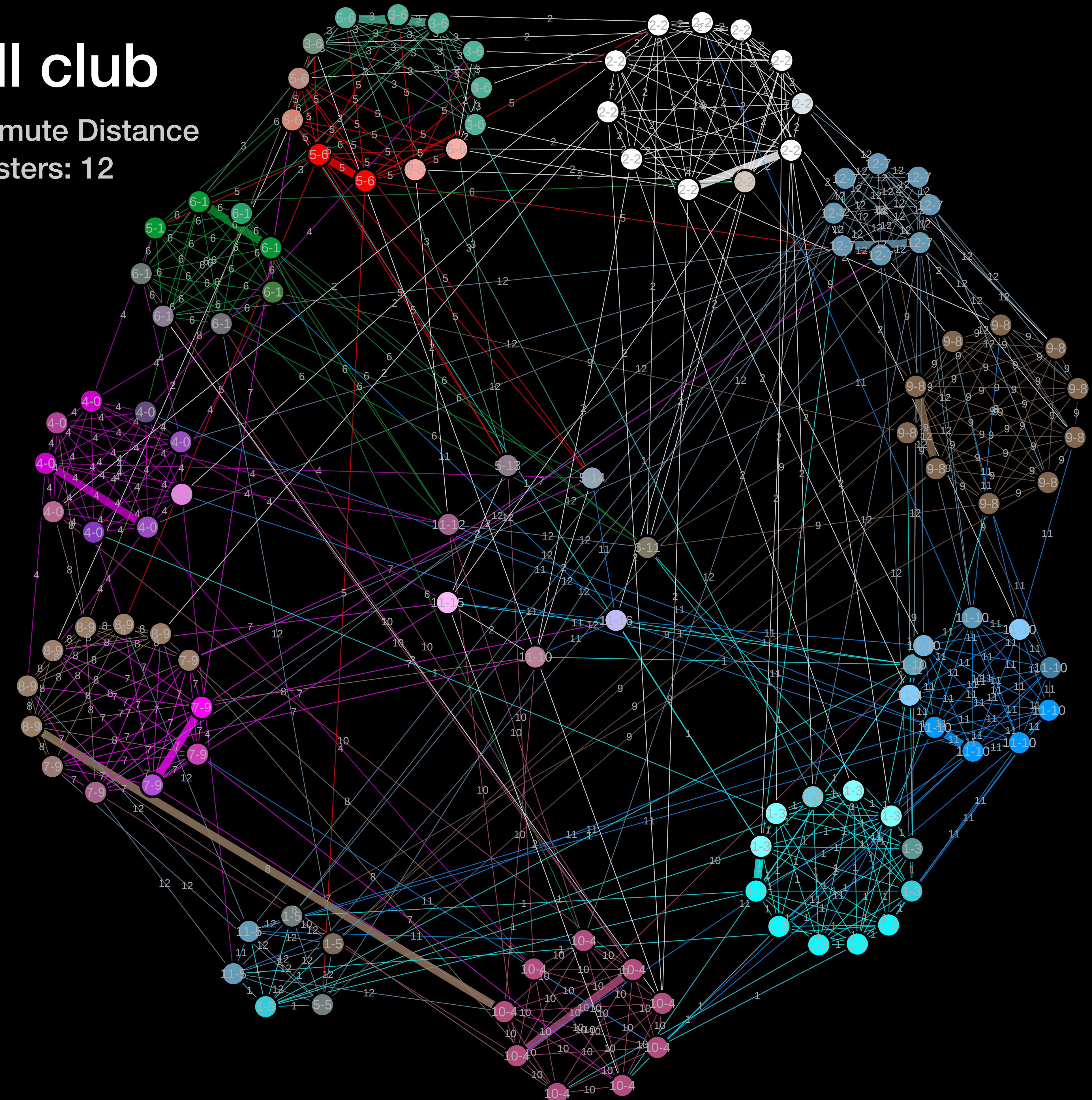


method output

Football club

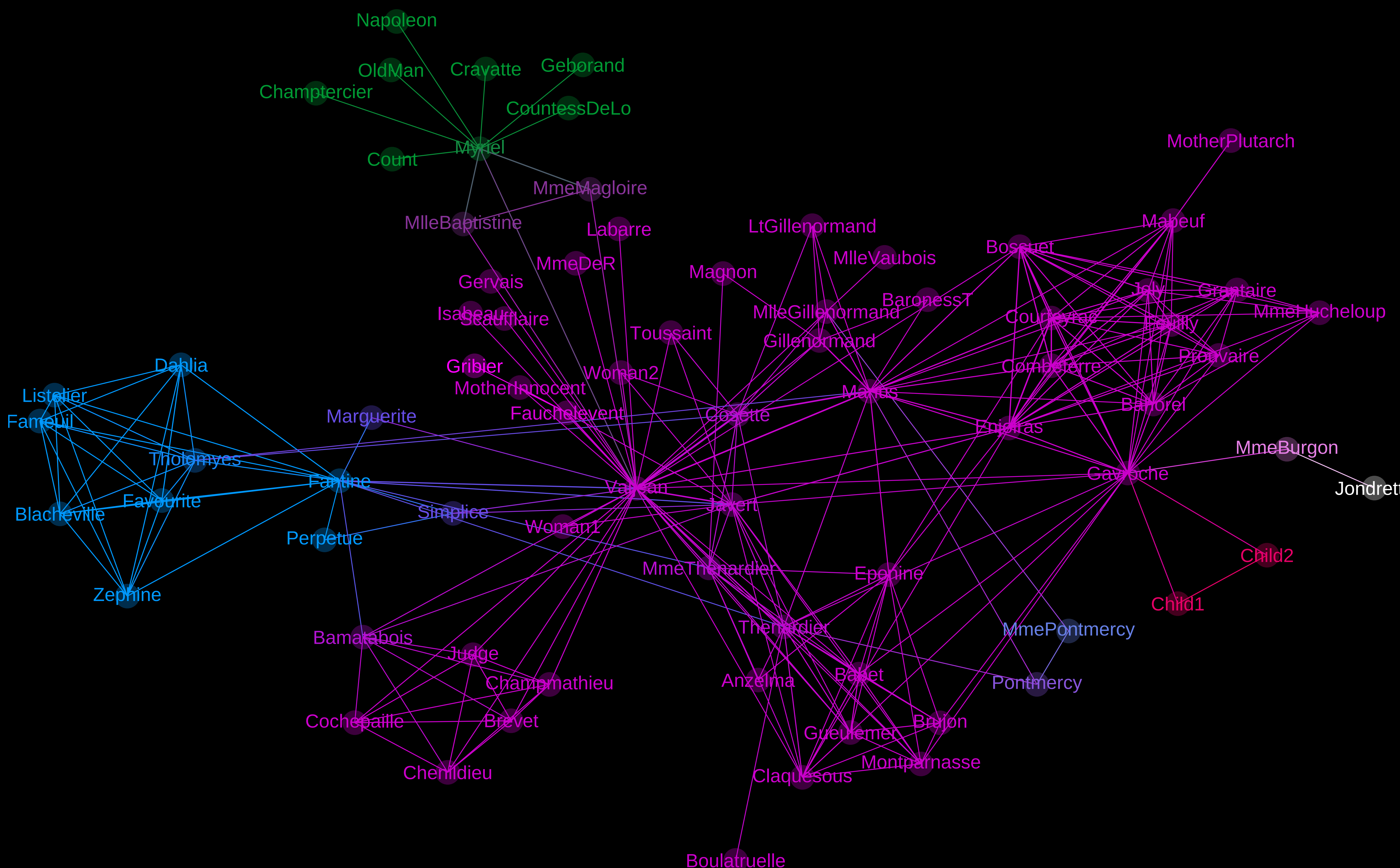
Distance: Commute Distance

Number of Clusters: 12



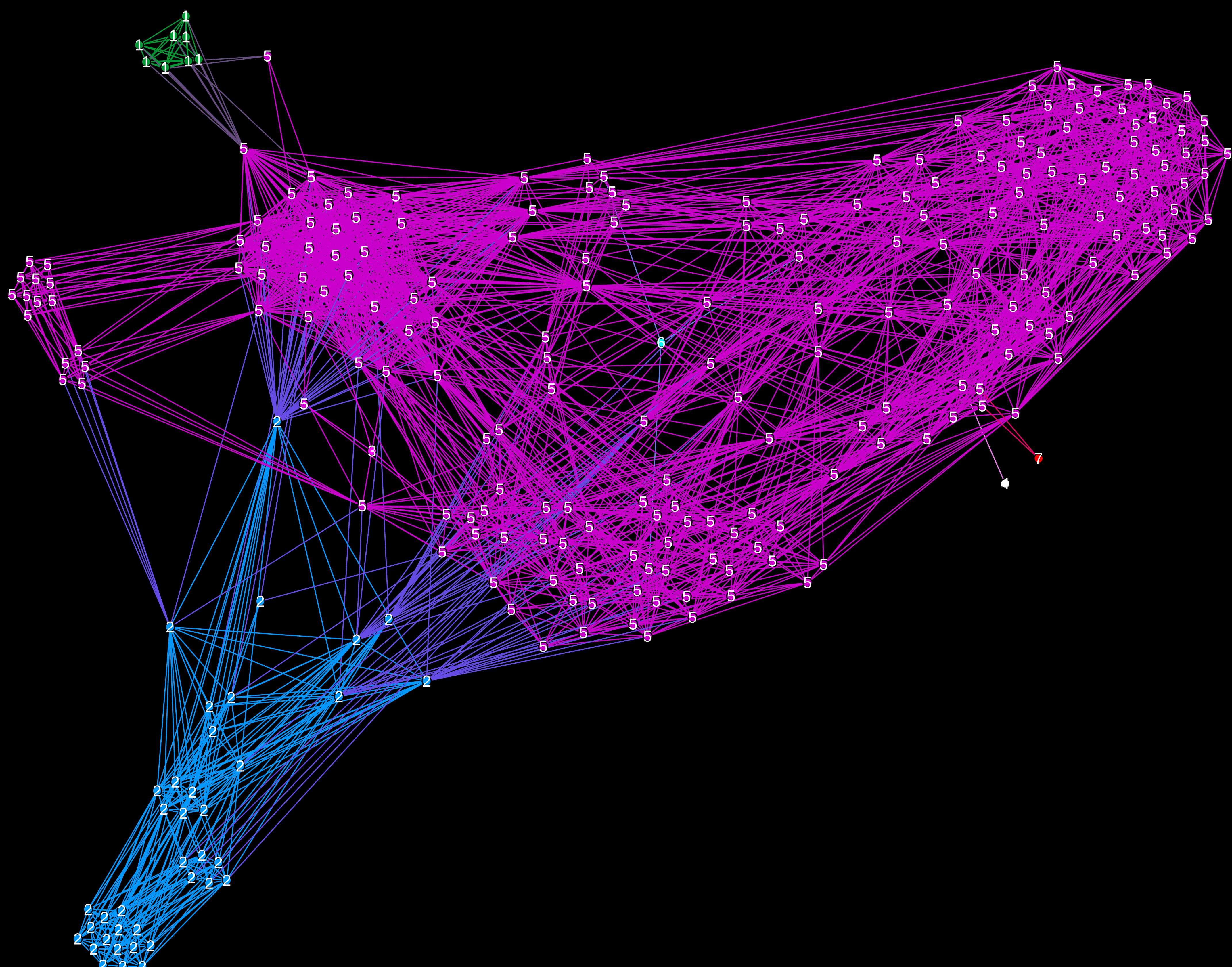
Les Miserables

Distance: Commute Distance
Number of Clusters: 7



Les Miserables – line graph

Distance: Commute Distance
Number of Clusters: 7



Community distance lost in space

Property (★): Vertices in the same cluster of the graph have a small commute distance, whereas two vertices in different clusters of the graph have a "large" commute distance.

$$\frac{1}{vol(g)} C_{ij} \approx \frac{1}{d_i} + \frac{1}{d_j}$$

The commute distance is not a useful distance function on large graphs

[Luxburg, U. V., Radl, A., & Hein, M. (2010). Getting lost in space: Large sample analysis of the resistance distance. In *Advances in Neural Information Processing Systems* (pp. 2622-2630)]

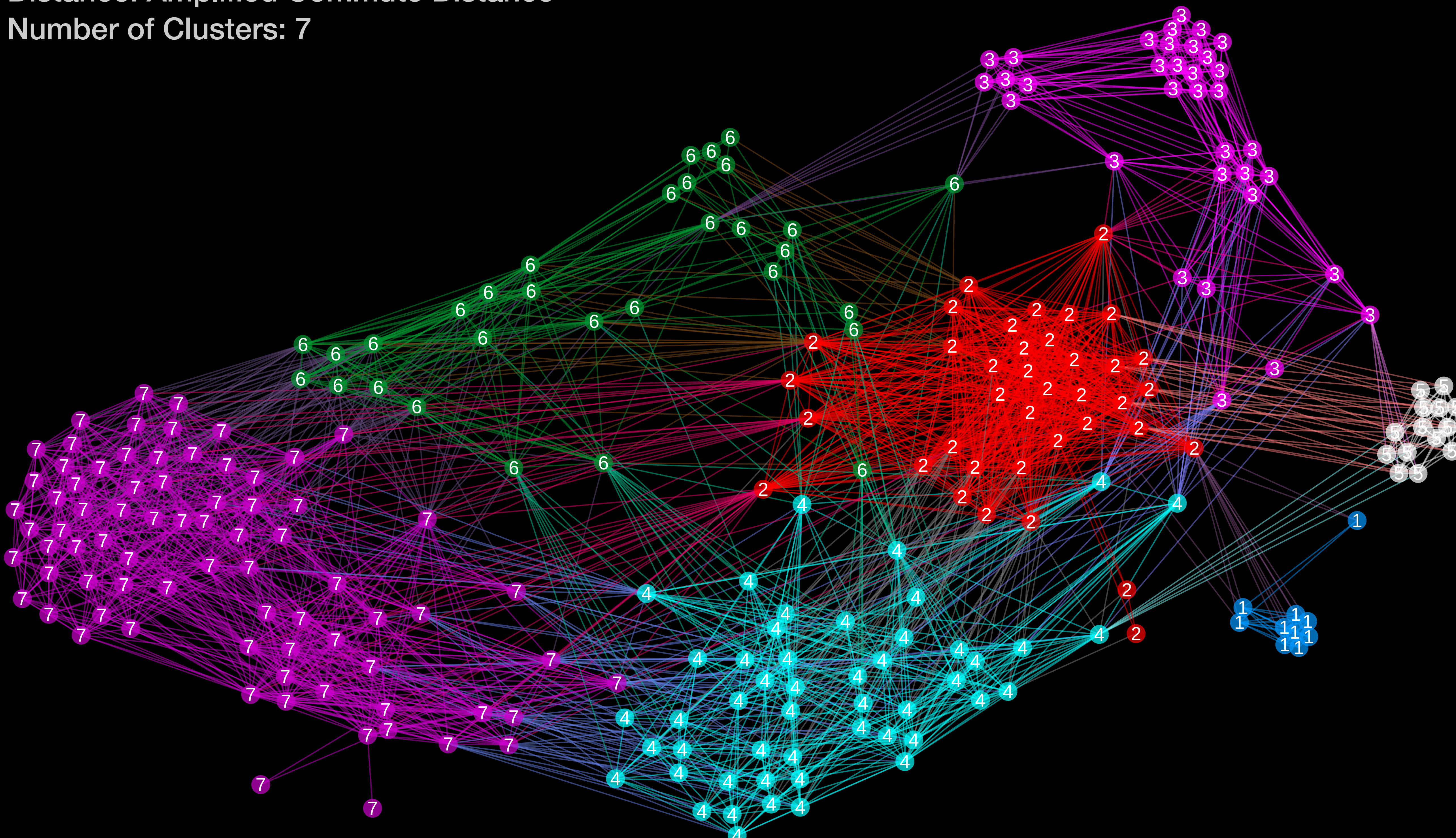
Amplified Commute distance

$$C_{amp}(i,j) = \frac{C_{i,j}}{vol(G)} - \frac{1}{d_i} - \frac{1}{d_j} + \frac{2w_{ij}}{d_i d_j} - \frac{w_{ii}}{d_i^2} - \frac{w_{jj}}{d_j^2}$$

[Luxburg, U. V., Radl, A., & Hein, M. (2010). Getting lost in space: Large sample analysis of the resistance distance. In *Advances in Neural Information Processing Systems* (pp. 2622-2630)]

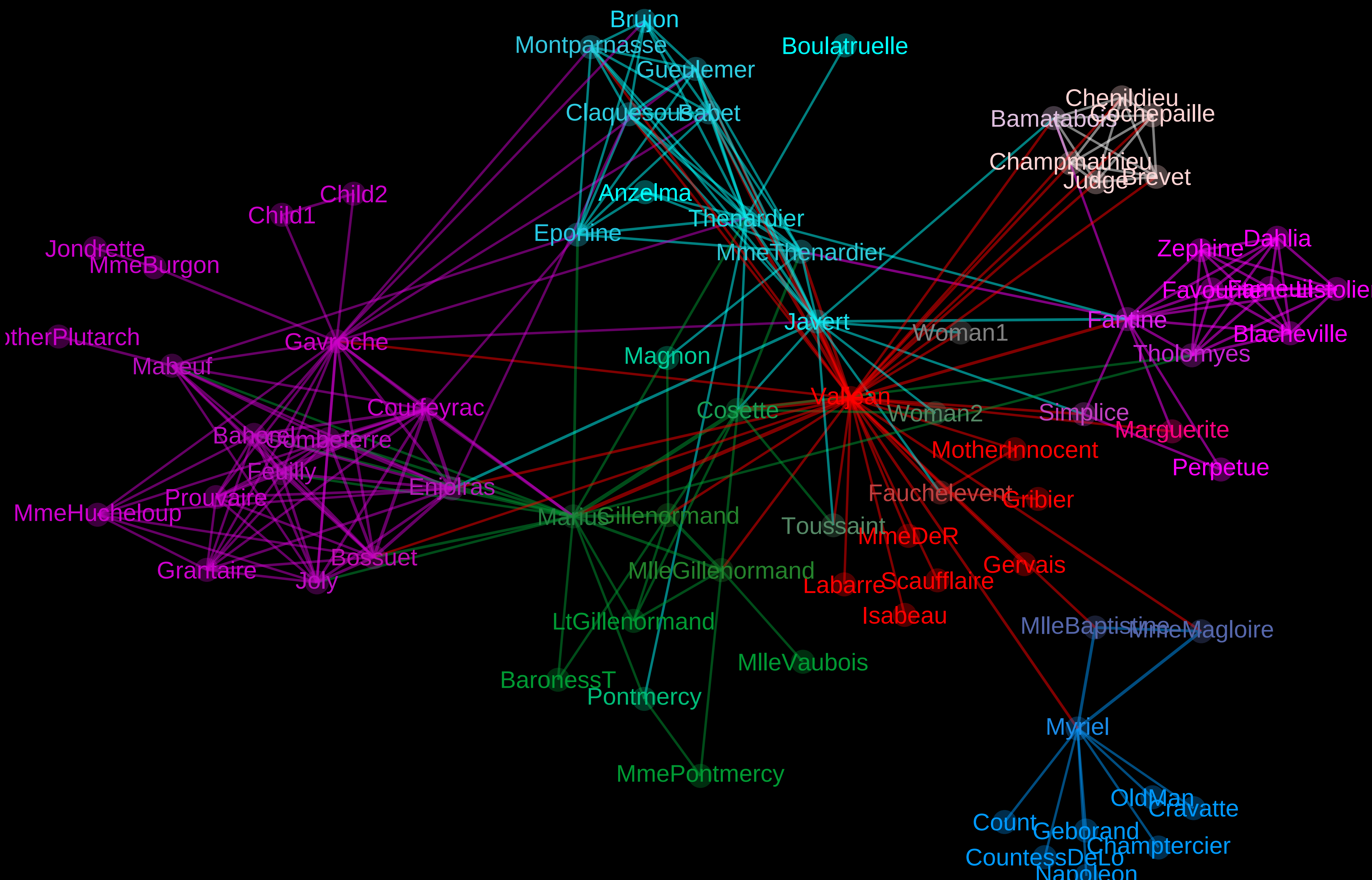
Les Miserables – line graph

Distance: Amplified Commute Distance
Number of Clusters: 7



Les Miserables

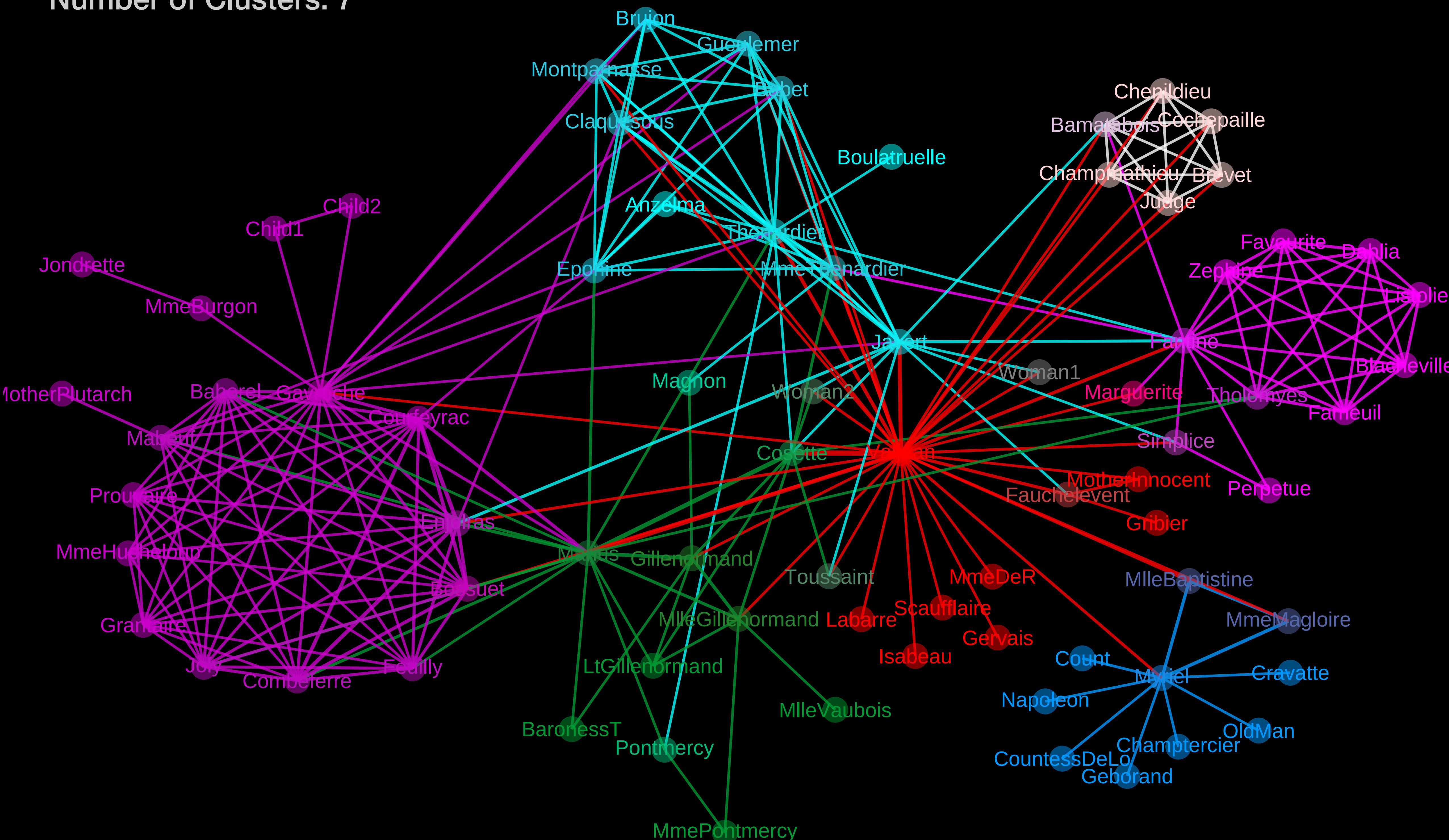
Distance: Amplified Commute Distance
Number of Clusters: 7



Les Miserables

Distance: Amplified Commute Distance

Number of Clusters: 7



Thank you for your attention

Questions?

“And what is the next step?”

–Panos M. Pardalos © 2018