PERFORMANCE OF A SYNCHRONIZED COMMUNITY DETECTION ALGORITHM*

A. BROWET†, P.-A. ABSIL† , AND P. VAN DOOREN†

Key words. community detection, modularity

AMS subject classifications. 05C70 , 92-08, 91C20

Extended Abstract. Networks are everywhere. In many applications, they are used to represent interactions between the different elements of interest. But at the same time as theoretical knowledge and practical tools of graph theory were developed, network sizes grew tremendously. Social networks, which were studied for a few dozens of individuals, are now composed of millions of people sharing online friendships; mobile phone networks have a penetration rate that grew from around 20% in the late nineties to almost 100% nowadays in developed countries; biological networks such as protein-to-protein interactions, may have billions of nodes; etc.

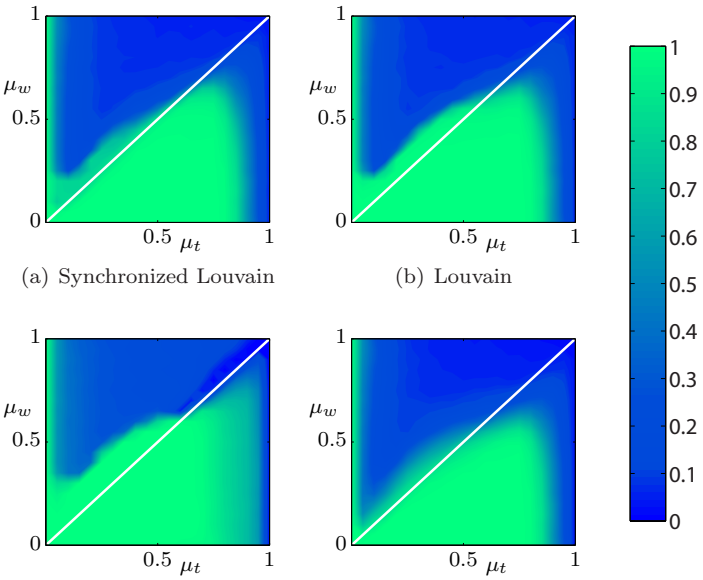
Many network properties (degree distribution, average distance or flow propagation for example) have been studied extensively. A key feature of real networks lies in the non-homogeneous local distribution of edges. Each node tends to be more densely connected with a specific subset of the vertices of the graph than with the rest of the network. These structural organizations of nodes are called communities.

Extracting communities from large networks have been a challenging problem for years, and a popular class of methods consists of optimizing the so-called modularity. This quality function introduced by Newman and Girvan [1] defines good communities as group of nodes with an internal edge density larger than the associated density expected from the degree distribution of the graph. Optimizing the modularity has been shown to be a NP-complete problem [2] and while many heuristics have been developed for specific contexts, a lot of them are not suitable for a more general framework due to their computational requirements or the structure of the network. The Louvain method [3] provides a good tradeoff between the complexity of the algorithm and the quality of the clustering. However the method still struggles to extract community structures for very large networks and is not suitable for parallel implementation.

We propose a synchronized version of the Louvain method where the synchronization also allows an efficient parallelization of the algorithm. Even if our implementation is built upon modularity maximization, we point out that our algorithm is also suitable for different quality functions like the Potts model of Reichardt and Bornholdt [4] for example. First, each node chooses an assignment to one of its neighbors, independently of the choice of the other nodes, based on the local gain of the quality function to merge the two nodes. Communities are then defined as the connected components in this directed assignment graph. Then, knowing the local decision of each node, we build a recursive synchronized correction step designed to improve the global behavior of the method. Three levels of correction are proposed: first we impose that the global gain for each community must be positive; we also impose that

*This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its author(s).

†Institute of Information and Communication Technologies, Electronics and Applied Mathematics, Catholic University of Louvain, 4 Av. G. Lemaitre, 1348 Louvain-la-Neuve, Belgium. email: arnaud.browet@uclouvain.be



(a) Synchronized Louvain (b) Louvain

(c) Infomap (d) Clauset

FIG. 1. *Performance of four community detection algorithms on the weighted LFR benchmark [5] with* 1000 *nodes. The quality of the clustering is computed as the normalized mutual information. The mixing parameter of the edge weight* μ_w *and the mixing parameter on the topology* μ_t *range from* 0 *to* 1.

the gain of assigning each node to its community must be positive; the last correction level imposes that the gain to assign a node to its community is larger than the gain to assign it to any other community. Those correction levels may produce switches of nodes between communities but also merges and splittings of communities. Finally, when a steady state has been found, communities are aggregated to form a new graph where each node represents a community and each vertex is the sum of the edges between two communities. Notice that the internal edges of communities are represented as self loops in the aggregated graph.

We compare the results of our algorithm on the popular benchmark of Lancichinetti and Fortunato [5] with the classical Louvain method [3], the fast modularity algorithm by Clauset and Newman [6] and the Infomap algorithm proposed by Rosvall and Bergstrom [7] (not based on modularity optimization). Some results are presented in Fig. 1 as heat maps of the normalized mutual information [8](NMI) for networks with 1000 nodes. The NMI is the normalized mutual entropy of 2 distribution of nodes in communities and ranges from 0 to 1. The extracted community structures are compared with the a-priori defined communities whose structures are controlled by two parameters. The topological mixing parameter μ_t controls the average proportion of edges pointing outside the communities, and the weight mixing parameter μ_w controls the proportion of the node strength (i.e. the sum of the edge weights) pointing outside the communities. The larger those two parameters are, the harder it becomes to extract good communities since there are either more edges pointing outside the communities or those edges are carrying more weight. It can be shown that the average weights of an edge from node i inside its community, $\left\langle w_i^{(int)} \right\rangle$, and

outside its community $\left\langle w_i^{(ext)} \right\rangle$ are given by

$$\left\langle w_i^{(in)} \right\rangle = \frac{1-\mu_w}{1-\mu_t} s_i \qquad \left\langle w_i^{(ext)} \right\rangle = \frac{\mu_w}{\mu_t} s_i,$$

where s_i is the strength of node i. The boundary $\mu_t = \mu_w$, depicted as a straight line in Fig. 1, represents the set of parameters where edges are expected to have the same weight on average inside and outside the communities. The results show that our algorithm is able to produce community structures whose quality is comparable to those obtained by the other algorithms. At the same time, we demonstrate that the computational time required by our method is considerably lower than for the other methods. In particular, our algorithm might be very appropriate as a preprocessing step for any other method.

REFERENCES

[1] M. E. J. Newman, and M. Girvan, *Finding and evaluating community structure in networks*, Physical Review E - Statistical, Nonlinear and Soft Matter Physics, 69 (2004).

[2] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner, *On Modularity Clustering*, IEEE Transactions on Knowledge and Data Engineering, 20 (2008), pp. 172–188.

[3] V. D. Blondel, J.-L. Guillaume, R. Lambiotte and E. Lefebvre, *Fast unfolding of communities in large networks*, Journal of Statistical Mechanics: Theory and Experiment (2008).

[4] J. Reichardt, and S. Bornholdt *Statistical mechanics of community detection*, Physical Review E, 74 (2006).

[5] A. Lancichinetti, and S. Fortunato, *Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities*, Physical Review E, 80 (2009).

[6] A. Clauset, M. E. J. Newman, and C. Moore *Finding community structure in very large networks*, Physical Review E, 70 (2004), pp. 1–6.

[7] M. Rosvall and C. T. Bergstrom, *Maps of random walks on complex networks reveal community structure*, Proceedings of the National Academy of Sciences of the United States of America, 105 (2008), pp. 1118–1123.

[8] L. Danon, J. Duch, A. Diaz-Guilera, and A. Arenas, *Comparing community structure identification*, Journal of Statistical Mechanics: Theory and Experiment, 9 (2005), p. 10.