A Fast Algorithm for the Iterative Calculation of Betweenness Centrality

J. Griebsch, T. Binder and A. Buchner
Institut für Informatik, Lehrstuhl für Effiziente Algorithmen
Technische Universität München, D-80290 München
{griebsch, buchner}@in.tum.de

Introduction

Various centrality measures capture different notions of a vertex's importance in a network [KS04]. The computationally rather involved betweenness centrality index has also been employed to identify functionally related sub-units/-networks in biochemical networks.

Methods

We use a dynamic variant for the iterative application of the Brandes Algorithm. In each iteration only one node is removed and, on average, large parts of the shortest paths graphs do not need to be recomputed. Our algorithm uses this observation to restrict the necessary re-computations to the affected nodes - an approach that is employed by a whole class of algorithms, the so called "Dynamic Graph" algorithms [DPZ95, EGI99].

In detail our algorithm works as follows:

The first step is one complete iteration of the Brandes algorithm, but in order to reuse the intermediate results, we store for each start vertex the distances, the number of shortest paths, as well as the list of predecessors for the other nodes. After the identification of the most central vertex we perform a breadth-first-traversal, starting at that node, in order to mark all vertices in which the distances become empty. This step is followed by an update of the respective distance values, number of shortest paths, predecessor lists and the fraction of shortest paths that contain the vertex.

After the identification of the most central node in a network and recomputing the centrality of the remaining nodes [HHJ03], in our experiments such a decomposition of the metabolic network of E. coli (1993 nodes and 3997 edges) took 1.5 hours on a 1 GHz machine using the fastest known algorithm for the computation of betweenness centrality by Brandes [Br01].

Results

Decomposition example: Citric Cycle and Glycolysis / Glyconeogenesis from E. coli

Automated separation of two metabolic pathways using betweenness centrality

Conclusions

The worst-case time complexity of the dynamic algorithm is equal compared to the iterated Brandes Algorithm while the observed running time for the complete decomposition of metabolic networks of different species is approximately 3-5 times faster. However, to achieve this speedup more memory usage is required.

For more details of the algorithm and the decomposition process of metabolic networks please contact one of the mentioned email addresses.

References