Quick Detection of Nodes with Large Degrees*

Konstantin Avrachenkov\textsuperscript{1}, Nelly Litvak\textsuperscript{2}, Marina Sokol\textsuperscript{1}, and Don Towsley\textsuperscript{3}

\textsuperscript{1} INRIA, 2004 Route des Lucioles, Sophia-Antipolis, France
\textsuperscript{2} University of Twente, The Netherlands
\textsuperscript{3} University of Massachusetts Amherst, USA

Abstract. Our goal is to quickly find top $k$ lists of nodes with the largest degrees in large complex networks. If the adjacency list of the network is known (not often the case in complex networks), a deterministic algorithm to find the top $k$ list of nodes with the largest degrees requires an average complexity of $O(n)$, where $n$ is the number of nodes in the network. Even this modest complexity can be very high for large complex networks. We propose to use the random walk based method. We show theoretically and by numerical experiments that for large networks the random walk method finds good quality top lists of nodes with high probability and with computational savings of orders of magnitude. We also propose stopping criteria for the random walk method which requires very little knowledge about the structure of the network.

1 Introduction

We are interested in quickly detecting nodes with large degrees in very large networks. Firstly, node degree is one of centrality measures used for the analysis of complex networks. Secondly, large degree nodes can serve as proxies for central nodes corresponding to the other centrality measures as betweenness centrality or closeness centrality \cite{8,9}. In the present work we restrict ourself to undirected networks or symmetrized versions of directed networks. In particular, this assumption is well justified in social networks. Typically, friendship or acquaintance is a symmetric relation. If the adjacency list of the network is known (not often the case in complex networks), a deterministic algorithm to find the top $k$ list of nodes with the largest degrees requires an average complexity of $O(n)$, where $n$ is the number of nodes in the network. We assume that the degree is available when accessing a node (if this is not the case, the complexity should be counted in terms of links). However, even linear complexity can be very high for very large, possibly varying, complex networks. In the present work we suggest using random walk based methods for detecting a small number of nodes with the largest degree. The main idea is that the random walk very quickly comes across large degree nodes. In our numerical experiments random walks outperform the standard deterministic algorithms by orders of magnitude in terms of

* This research was sponsored by INRIA Alcatel-Lucent Joint Lab, by the NSF under CNS-1065133, and the U.S. Army Research Laboratory under Cooperative Agreement W911NF-09-2-0053.
computational complexity. For instance, in our experiments with the web graph of the UK domain (about 18,500,000 nodes) the random walk method spends on average only about 5400 steps to detect the largest degree node. Potential memory savings are also significant since the method does not require knowledge of the entire network. In many practical applications we do not need a complete ordering of the nodes and even can tolerate some errors in the top list of nodes. We observe that the random walk method obtains many nodes in the top list correctly and even those nodes that are erroneously placed in the top list have large degrees. Therefore, as typically happens in randomized algorithms \cite{12,13}, we trade off exact results for very good approximate results or for exact results with high probability and gain significantly in computational efficiency.

The paper is organized as follows: in the next section we introduce our basic random walk with uniform jumps and demonstrate that it is able to quickly find large degree nodes. Then, in Section 3 using configuration model we provide an estimate for the necessary number of steps for the random walk. In Section 4 we propose stopping criteria that use very little information about the network. In Section 5 we show the benefits of allowing few erroneous elements in the top \( k \) list. Finally, we conclude the paper in Section 6.

2 Random Walk with Uniform Jumps

Let us consider a random walk with uniform jumps which serves as a basic algorithm for quick detection of large degree nodes. The random walk with uniform jumps is described by the following transition probabilities \cite{1}

\[
p_{ij} = \begin{cases} \frac{\alpha/n + 1}{d_i + \alpha}, & \text{if } i \text{ has a link to } j, \\ \frac{\alpha}{d_i + \alpha}, & \text{if } i \text{ does not have a link to } j, \end{cases}
\]

where \( d_i \) is the degree of node \( i \). The random walk with uniform jumps can be regarded as a random walk on a modified graph where all the nodes in the graph are connected by artificial edges with a weight \( \alpha/n \). The parameter \( \alpha \) controls the rate of jumps. Introduction of jumps helps in a number of ways. As was shown in \cite{1}, it reduces the mixing time to stationarity. It also solves a problem encountered by a random walk on a graph consisting of two or more components, namely the inability to visit all nodes. The random walk with jumps also reduces the variance of the network function estimator \cite{1}. This random walk resembles the PageRank random walk. However, unlike the PageRank random walk, the introduced random walk is reversible. One important consequence of the reversibility of the random walk is that its stationary distribution is given by a simple formula

\[
\pi_i(\alpha) = \frac{d_i + \alpha}{2|E| + n\alpha} \quad \forall i \in V,
\]

from which the stationary distribution of the unperturbed random walk can easily be retrieved. We observe that the modification preserves the monotonicity of