REPORT KSHELL DECOMPOSITION

Pankaj Verma, 2015CSB1022
Soumyadeep Roy, 2015CSB1035

November 27, 2018

1 INTRODUCTION

K-Shell decomposition helps in finding out the shell numbers of various nodes in a particular graph. This is particularly helpful in finding out the core of any graph. With the help of this knowledge we are working on ways to reach the core of the graph from any random point with only local knowledge to rely upon.

2 Literature Review

2.1 The H-index of a network node and its relation to degree and coreness

This paper discusses how degree and coreness are related with each other wih the help of a property called H-index. H-index is the property of a node that is difined by the relationship given below :

Let $G(V, E)$ denote a graph G with V vertices and E edges making it up. Now choosing any particular node let the degree of the node i be denoted by k_i. Then the zero-order H-index of node i is $h_i^{(0)} = k_i$, i.e is the degree of the node. Let the $(n-1)$-order H-index of neighbours of i be denoted as $h_{j_1}^{(n-1)}, h_{j_2}^{(n-1)}, h_{j_3}^{(n-1)}, ..., h_{j_{k_i}}^{(n-1)}$. Now the n-order H-index can be calculated as :

$$h_i^{(n)} = \mathcal{H}(h_{j_1}^{(n-1)}, h_{j_2}^{(n-1)}, h_{j_3}^{(n-1)}, ..., h_{j_{k_i}}^{(n-1)}).$$

NOTE : Here $y = \mathcal{H}(x_1, x_2, x_3, ..., x_n)$ signifies that y is the maximum integer such that there exist atleast y elements in $(x_1, x_2, x_3, ..., x_n)$ each of which is no less than y.

This paper helps in establishing the fact that as the order of H-index value increases to infinity, the value of the H-ndex converges to the coreness value of the node. This has been proved by a theorem.

The paper also discusses about Asynchronous updating procedure..

2.2 Estimating Shell-Index in a Graph with Local Information

This main idea of this paper is to find the shell index of a node using local information. This will help to avoid the major drawback of K-Shell decomposition i.e. requirement of the the entire graph, which is not feasible for large scale dynamic networks.

2.2.1 Theorem

The Shell index of a node u can be computed as $k_s(u) = \mathcal{H}(k_s(v)| \in ngh(u))$, where $ngh(u)$ is the set of the neighbours of node u.

Proof : In the i^{th} iteration, nodes which have i or less connections with other nodes are removed. As it is the i^{th} iteration, therefore these removed nodes are connected with nodes which have shell index i or greater than i.

2.2.2 Hill Climbing Based approach to identify top rank nodes

In this algorithm inputs are Graph G, Initial node u, Repeat count number k (maximum no of times a crawler is allowed to reach a local maxima) and $maxindex$ (maximum $H_2 - Index$ in the graph G). Starting from the node u the crawler traverses to one of its non visited neighbours which also has the highest $H_2 - Index$. Upon continuing this process the crawler at one point reaches a maximum. If the maximum is a local maxima then the counter (initially 0) is increased by 1 and the flow of travel is passed on to one of its non visited neighbours randomly. This processes continues untill the crawler reaches the $maxindex$ or the counter reaches the value of k, at which point traversal is terminated.

Some modifications in the given algorithm includes traversal to highest degree neighbour instead of randomly choosen node in case of local maxima.

3 Work Done

Various methods were applied to reach from periphry to core based on only a limited number of node traversal. The following methods are discussed below :

3.1 Random walk

Here starting from any node in the graph, the core of the graph was to be reached. Movement from one node to the next neighbouring node was done randomly.

3.2 Hill climbing

Here starting from any node in the graph, the core of the graph was to be reached based on the value of $H_2 - Index$ of the neighbouring node. Two variations of the hill climbing method was seen.

3.2.1 Travelling highest degree node which is least travelled

Traversal to subsequent nodes is decided based on the highest $H_2 - Index$ neighbours of a node and among those the least travelled node. This allows traversal to be towards higher $H_2 - Index$ nodes mostly.

3.2.2 Travelling least travelled node with highest degree

One of the basic issue of the previous approach is hinderance of traversal due to reaching of local maxima. In order to avoid that, a new method was applied where, the next neighbouring node to be travelled is decided on certain factors. The first factor is that the node should be least travelled. Second factor is that among all the least travelled nodes it should have the highest degree. This method of progression allows us to cover more number of nodes. Also if a node is already traversed, then it is better to try a different node as it may open up the oppurtunity to reach the core.