McNair Scholars Journal

Volume 20 | Issue 1 Article 16

2016

Improving African Internet Traffic through Maximization of Node Centrality

Gloire Rubambiza
Grand Valley State University

Improving African Internet Traffic through Maximization of Node Centrality

Gloire Rubambiza
McNair Scholar

Jerry Scripps
Faculty Mentor

ABSTRACT

For most African nations, the cost of an Internet subscription is more than the average yearly per capita income – which inherently makes Internet access a quixotic amenity for the majority of Africans. This disproportionate connectivity creates a disadvantage for Africa's academic potential because most of its Internet traffic is routed through international fiber optic links – which is costlier than direct connections within Africa [1]. One of the causes for slower Internet traffic is the lack of cooperation between the Internet Service Providers (ISPs) across Africa. This study will explore whether we can increase the average closeness of a sample network representing African Internet traffic by designing and testing strategic link-prediction algorithms versus a random link-prediction algorithm.

1. INTRODUCTION

Networks are used extensively to represent relationships among social connections, biological processes, collaboration between faculty of a university, road maps in a country, computer connections, etc. The entities in the network such as computers in an organization are known as *nodes* or *vertices* of the network and the links between them are called the *edges* [2] . The size of networks generally depends on the number of nodes and edges as shown in Figure 1. They can vary from a small organization of ten computers to a social network like Facebook with billions of nodes and edges.

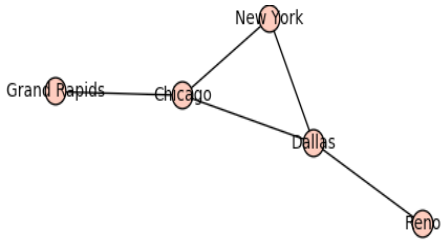

Figure 1: Sample network of cities with direct flights to every other city

The complexity of a network can be defined in terms of the number of nodes and edges present in the network, and it can increase based on the relationships among the vertices. Communities or cliques in a network can be discovered through application of various computer algorithms on the network. Properties of nodes, edges, and networks can be expressed using metrics. A few examples of metrics expressing the centrality [3] of a network include diameter, closeness, and farness among others.

The network used in our computations represents the average round trip time (RTT) of packets sent via pinging of vantage points located across Africa and North America. The nodes are web servers with edges between them characterizing the average RTT in milliseconds.

Over the past decade, a plethora of Internet Service Providers (ISPs) have sprung up on the African continent. However, their services' effectiveness is limited by geography, competition, and most importantly governmental over-regulation, thereby diminishing collaboration that could potentially improve the intra-connection on the continent. As a result, Internet traffic from large organizations such as universities is forced to travel through intercontinental fiber optic links in Europe and North America to access information already on the continent at other universities [1], inadvertently increasing the RTT of pings between the universities. The objective of our experiments below is to observe whether a shortest path-based, closeness-based, or degree-based algorithm increases the average closeness of the network by adding k number of links between the nodes – in this case a ground link between universities or other vantage points – more efficiently than a random addition of k links.

2. NOTATION AND METRICS

A network $G = \{V, E\} G = \{V, E\}$ is a set of vertices $= \{v_1, v_2 \ldots v_n\} = \{v_1, v_2 \ldots v_n\}$, and a set of edges $E \in V \times V E \in V \times V$ where each edge $e_{ij} e_{ij}$ connects two vertices – in this case $v_i\, v_i$ and $v_j v_j$. For a better understanding of the algorithms, we shall represent the sets of edges as an adjacency matrix $Adj = [a_{ij}]_{n \times n} Adj = [a_{ij}]_{n \times n}$ where $a_{ij} = 1 a_{ij} = 1$ if there is a link between nodes $v_i v_i$ and $v_j v_j$ and nn is the number of vertices in the network.

The following metrics characterize the relationships between vertices of VV vis-à-

vis each other and the network in general:

Metric 1: The shortest path between two vertices $v_i\ v_i$ and v_jv_j is the least sum of traversed edges to travel from v_iv_i to v_jv_j

$$p_{ij} = n \ means \exists \ a \ path \ , e_{i1}, e_{i2}, e_{i3}, \ldots\ldots$$

$$, e_{nj} \ and \ no \ shorter \ path \ can \ be \ found$$

Metric 2: The degree of a vertex v_iv_i is the sum of all edges connected to v_iv_i, as expressed in terms of the neighborhood N N of v_iv_i

$$N(v_i) = \{\, v_j \ \in V \mid e_{ij} \ \in E\}$$

$$deg_{vi} = |N(v_i)|$$

Metric 3: The closeness of a vertex v_iv_i the reciprocal of the sum of distances from vertex v_iv_i to every other vertex in the network

$$c_i = \frac{1}{\sum_{j=1}^{n} p_{ij}} c_i = \frac{1}{\sum_{j=1}^{n} p_{ij}}$$

Metric 4: The average closeness of a network is the sum of the closeness of individual vertices divided by the total number of vertices

$$\bar{c} = \sum_{i=1}^{n} c_i \bar{c} = \sum_{i=1}^{n} c_i \ /(n)n)$$

These metrics constitute the basis for the strategic algorithms described in the following method section.

3. METHOD

The method used comprises of tests on an adjacency matrix and algorithms described below. The algorithms operate on a matrix representing a sample network of pinging time between vantage points in Africa and North America (See Figure 2).

The adjacency matrix, as shown in Figure 3, was built as follows. The Stanford Linear Accelerator Center (SLAC) vertex, a U.S Department of Energy National Laboratory operated by Stanford University, represents a hopping point of a ping traveling through an intercontinental route. Since SLAC – represented by the Stanford node in Figure 2 – can successfully ping all vantage points, all its matrix entries are set to 1. For two vantage points located in Africa whose pinging time is less than the intercontinental route via Stanford, the corresponding entry is set to 1 in the matrix. Otherwise, the matrix entry is set to 0. Additionally, if the ping times out on all four packets sent, the matrix entry is set to 0. All the algorithms are given the

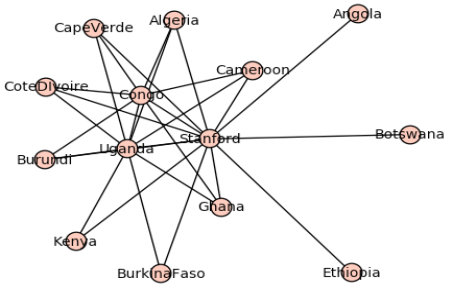

Figure 2: Network representation of pinging across vantage points in Africa and North America

0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	0	1	0	0	0	0	1	1
1	0	0	0	1	1	1	0	1	0	1	0	1	1
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	0	1	1	1	0

Figure 3: Adjacency Matrix of pinging possibility across vantage points in Africa and North America

3.1 Diameter-based Algorithm

Using Dijkstra's shortest path algorithm, all the shortest paths between nodes in the matrix are computed and stored in a two-dimensional array. Next, the highest value, $a_{ik}a_{ik}$, is retrieved in the array, where $i\ i$ and kk are the nodes linked by the path. If the nodes are not directly linked, then entry $Adj_{ik}Adj_{ik}$ is set to 1.

node1=0, node2= 0, highestpath= 0, shortestpaths [ ][ ];

foreach $i \ \in shortestpaths$ $i \ \in shortestpaths$ **do**

| **foreach** $k \ \in shorestpaths[i]$ $k \ \in shorestpaths[i]$ **do**

| | **if** shortestpaths [ii][$k\ k$] > highestpath && shortestpaths [ii][$k\ k$] > 1 **then**

| | | highestpath = shortestpaths [ii][$k\ k$];

| | | node1 = ii;

| | | node2 = kk;

| | **end**

| **end**

end

$Adj_{ik}Adj_{ik}$ = 1;

3.2 Degree-based Algorithm

A call to an algorithm that computes the degrees of each node in the matrix returns the degrees in an array. Next, a copy of the array is made, and the original array is sorted. If there is no direct link between the two nodes ii and kk with the lowest degrees, then $Adj_{ik}Adj_{ik}$ and $Adj_{ki}Adj_{ki}$ are set to 1.

degrees [ ], copy [ ];

sort(degrees);

foreach $i \ \in degreesi \ \in degrees$ **do**

| **foreach** $k \ \in degrees$ $k \ \in degrees$ **do**

| | **if** copy [ii] <= degrees [1] && copy [kk] <= degrees [1] && $Adj_{ik}Adj_{ik}$<1 **then**

| | | $Adj_{ik}Adj_{ik}$ = $Adj_{ki}Adj_{ki}$ = 1;

| | **end**

| **end**

end

3.3 Closeness-based Algorithm

A call to an algorithm that computes the closeness of individual nodes in the matrix returns the closeness values in an array. Next, a copy of the array is made, and the original array is sorted. If there is no direct link between the lowest nodes ii and k k with the lowest closeness values, then $Adj_{ik}Adj_{ik}$ and $Adj_{ki}Adj_{ki}$ are set to 1.

closeness [ ], copy [ ];

sort(closeness);

foreach $i \ \in closenessi \ \in closeness$ **do**

| **foreach** $k \ \in closeness$ $k \ \in closeness$ **do**

| | **if** copy [ii] <= closeness [1] && copy [kk] <= closeness [1] && $Adj_{ik}Adj_{ik}$ <1 **then**

| | | $Adj_{ik}Adj_{ik}$ = $Adj_{ki}Adj_{ki}$ = 1;

| | **end**

| **end**

end

3.4 Random Algorithm

While traversing the adjacency matrix, we save the ii and kk values of every Adj_{ik} Adj_{ik} with a 0 entry in a two-dimensional array. Next, a random row is selected in the two-dimensional array. The Adj_{ik} Adj_{ik} entry with ii and kk values from the random choice is set to 1.

counter = 0, currentRow = 0, node1 = 0, node2 = 0, randomRow = 0;

foreach $i \in Adji \in Adj$ **do**

| **foreach** $k \in Adjk \in Adj$ **do**

| | **if** copy $ii < kk$ && $Adj_{ik}Adj_{ik} = 0$ **then**

| | | counter++;

| | **end**

| **end**

end

nonEdges [counter][2];

foreach $i \in nonEdgesi \in nonEdges$ **do**

| **foreach** $k \in nonEdges[i]$ $k \in nonEdges[i]$ **do**

| | **if** copy $ii < kk$ && $Adj_{ik}Adj_{ik} = 0$ **then**

| | | nonEdges[currentRow][0] = v;

| | | nonEdges[currentRow][1] = w;

| | | currentRow++;

| | **end**

| **end**

end

rdm = new Random ();

randomRow = rmd.nextInt;

node1 = nonEdges[randomRow][0];

node2 = nonEdges[randomRow][1];

$Adj_{node1node2}Adj_{node1node2} = 1$;

3.5 Complexity

Shortest path-based: Since the algorithm iterates through all the shortest paths in the network and performs the three assignments if the condition is met, the worst case scenario is approximately $4n^2 + 54n^2 + 5$, which simplifies to $O(n^2)O(n^2)$.

Degree-based: With only one operation being performed in the double iteration, the worst case is $2n^2 + 32n^2 + 3$. Thus, the complexity is $O(n^2)O(n^2)$.

Closeness-based: The speed of the double iteration depends on the number of elements in the network itself, hence the complexity is $O(n^2)O(n^2)$.

Random: While the worst case scenario for complete algorithm is approximately $6n^2$ $6n^2$, the algorithm can speed up incredibly if there are not a lot of non-edges; therefore, the complexity is only $O(n^2)$ $O(n^2)$.

4. EXPERIMENTS

The experiments were designed to compare the performance of the different algorithms discussed in the previous section. We worked with slightly different data sets to ensure the performance of the algorithms was consistent – at least in the realm of the pinging network of Africa. Although the complexities of the algorithms are the same, it is shown that at least a majority of the strategic algorithm perform better than the random link addition algorithm.

4.1 Data Collection and Set up

Most of our data was provided by an ongoing collection of pinging data found at the SLAC National Accelerator Laboratory's website (www-wanmon.slac.stanford.edu/cgi-wrap/pingtable.pl) that represents a table of average monthly RTTs from the lab to hundreds of web servers across the world. We picked pings of strategic points in Africa in order to cover pings from most of the regions on the continent. For emphasis purposes, most of the web servers chosen are located on university campuses in Africa. The monthly pings from July 2015 to June 2016 for each point were averaged and these constituted the criteria for a node and links to said node as explained in section 3.

Due to time and financial restraints, the remainder of the pinging data was provided by two former colleagues: Elie Nsesi, who is located in Goma, D.R. Congo, and Peter Bampeire who is located in Kampala, Uganda. Both colleagues pinged the other strategic points including Stanford and other African pinging points as shown in Table1. For any other points in Africa where physical pinging was infeasible, we assumed a pinging time of 0 and the shortest-path to other countries to be going through SLAC.

4.2 Algorithm Comparison

Table 1 was used to create the adjacency matrix shown in Figure 3 earlier. All four algorithms are given the adjacency matrix as input, and the link addition specified by the algorithm is repeated kk times for each algorithm, then average closeness is computed after each addition for observation purposes.

The experiments were performed with the intention to compare the results for average closeness of the four algorithms after kk additions of links in the adjacency matrix. The results, shown in Figure 4 and Figure 5, represent the increase in average closeness of the algorithms on the same scale after 5 and 10 additions respectively. Figure 6 represents the increase in average closeness of the algorithms on the same scale after 25 link additions.

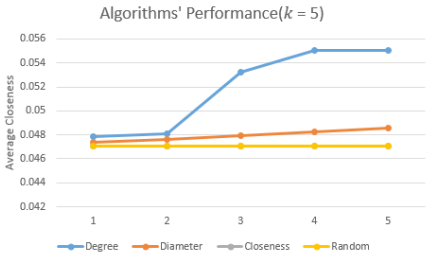

Figure 4: Individual algorithm performance after 5 link additions

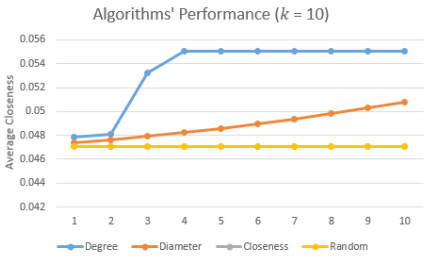

Figure 5: Individual algorithm performance after 10 link additions

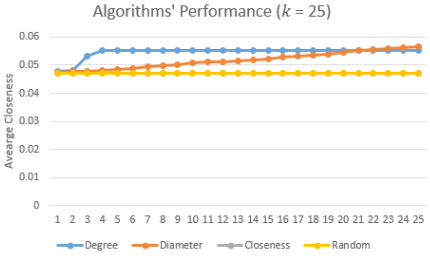

Figure 6: Individual algorithm performance after 25 link additions

	AG	AN	BF	BTW	BU	CA	CV	CNG	CI	ET	GH	KE	UG	ST
Algeria	0							258					313	216
Angola		0						0					0	244
Burkina Faso			0					397					488	321
Botswana				0				0					0	355
Burundi					0			112					293	321
Cameroon						0		370					332	267
Cape Verde							0	353					375	235
Congo D.R	258	0	397	0	112	370	353	0	299	0	307	648	423	392
Cote D'Ivoire								299	0				309	232
Ethiopia								0		0			558	378
Ghana								307			0		66	252
Kenya								648				0	524	285
Uganda	313	0	488	0	293	332	375	423	309	558	446	524	0	560
Stanford	216	244	321	355	321	267	235	392	232	378	252	285	560	0

Table 1: RTT of pings between web servers across Africa and Stanford in milliseconds

4.3 Results

As shown in the algorithm performance data in Figures 4, 5, and 6 the algorithms performed differently on the same scale. All the algorithms had the same initial average closeness, but they achieved different values of average closeness as links were added.

The degree-based algorithm, represented in Figure 5, spikes the average closeness by approximately 17% in the first few link additions from 0.047 to 0.055 but stabilizes after 4 link additions at 0.055. This implies that after 4 link additions, every other link that can be added would not change the average closeness of the network.

Unlike the degree-based algorithm, the diameter algorithm increases the average closeness steadily with every link addition. The average closeness climbs steadily from .047 and stabilizes at approximately 0.075 after 64 link additions – a 60% increase. Additionally, the diameter-based algorithm, as shown in Figure 6 above, surpasses the degree-based algorithm in average closeness after 21 link additions. A larger network may allow the diameter-based algorithm to overtake the degree-based algorithm earlier than it did in the African Internet traffic network.

In contrast, the closeness-based and random algorithms stay stagnant in terms of increasing the average closeness. This implies that the probability of a link addition using the random algorithm improving the average closeness is nearly 0. Although the closeness-based algorithm does not show an improvement in comparison to the random algorithm, the majority of the strategic link addition algorithms perform significantly better than the random algorithm – which concurs with our original hypothesis. Despite having the same complexity, the diameter-based and degree-based algorithm show more promise than the closeness-based and random algorithms.

RELATED WORK

Improvement in Internet connectivity in Africa can have significant impacts on the continent's development on many fronts including health, research [5], and education [3]. A recent study of networking of intra-African Internet traffic by Chavula, Feamster et al [1] showed that 75% of Internet traffic originating in African universities and destined for Africa takes a link outside the continent, with a latency that is double that of intra-Africa traffic. The study provides concrete evidence for a need in improvement of African Internet connectivity. The traffic can be represented as a network whose topology can be studied and enhanced by computations. A question of interest is whether there exists a technique to predict what links, if added, could change the topology and evolution of a network significantly – a question explored by Liben-Nowell and Kleinberg [4]. This study fuses the topics of African Internet traffic and link prediction to explore whether there is an efficient technique to add links in a sample network and improve the average centrality of the network and thus the African Internet traffic flow.

CONCLUSION

Three strategic link prediction algorithms and a random algorithm were presented in this study to explore whether there exists an efficient way to add fiber optic or other networking links in Africa to improve the average closeness in a sample network representing Internet traffic in Africa. It was shown that the diameter-based and degree-based algorithms performed significantly better than the closeness-based and random algorithm by 60% and 17% respectively. Though the study had time and financial limitations in terms of data collection of pinging RTTs in Africa, future work will include a trip to Africa for extensive data collection and a proposal to African governments for a consideration of the algorithms when planning future additions of inter-continental fiber-optic links. Other future directions may involve optimizing the strategic algorithms to work with larger data sets such as world Internet traffic. Ultimately, this study targeted solving the African Internet traffic problem, but the algorithms may have a myriad of other applications.

ACKNOWLEDGEMENTS

We thank the GVSU McNair Scholars Program for funding this study. Additionally, we thank Eli Nsesi and Peter Bampeire for their generous time and effort to obtain pinging data on location from Goma, D.R. Congo, and Kampala, Uganda respectively.

REFERENCES

[1] J. Chavula, N. Feamster, A. Bagula and H. Suleman. Quantifying the effects of circuitous routes on the latency of intra-Africa Internet traffic: A study of research and education networks. *6th International Conference on e-Infrastructure and e-Services for Developing Countries*, Kampala, 2014.

[2] M. Newman. Networks: An Introduction, Ann Arbor: Oxford University Press, 2010.

[3] O. Coeur De Roy. The African challenge: Internet, networking and connectivity activities in a developing environment. *Third World Quarterly,* 18(5): 883-898, 1997.

[4] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. *Twelth Annual ACM International Conference on Information and Knowledge Management(CIKM'03)*, New York, 2003.

[5] R. Echezona and U. C.F. African university libraries and Internet connectivity: challenges and the way forward. *Library Philosophy and Practice(e-Journal)*, Nsukka, 2010.