Prediction Of Arrival Of Nodes In A Scale Free Network

Vijay Mahantesh SM
Student, PESIT, Bangalore, India
Intern, ISI, Chennai, India
vijaym123@gmail.com

Sudarshan Iyengar
ISI
Chennai, India
sudarshaniisc@gmail.com

Vijesh M
Student, PESIT, Bangalore, India
Intern, ISI, Chennai, India
mv.vijesh@gmail.com

Shruthi R Nayak
Student, PESIT, Bangalore, India
Intern, ISI, Chennai, India
rn.shruthi@gmail.com

Nikitha Shenoy
Student, PESIT, Bangalore, India
Intern, ISI, Chennai, India
nikithashenoyk@gmail.com

Ravi Sundaram
ISI
Chennai, India

***Abstract*—Most of the networks observed in real life obey power-law degree distribution. It is hypothesized that the emergence of such a degree distribution is due to preferential attachment of the nodes. Barabasi-Albert model is a generative procedure that uses preferential attachment based on degree and one can use this model to generate networks with power-law degree distribution. In this model, the network is assumed to grow one node every time step. After the evolution of such a network, it is impossible for one to predict the exact order of node arrivals. We present in this article, a novel strategy to partially predict the order of node arrivals in such an evolved network. We show that our proposed method outperforms other centrality measure based approaches. We bin the nodes and predict the order of node arrivals between the bins with an accuracy of above 80%.**

***Keywords*-preferential attachment, scale-free networks, node-arrival ordering, node aging**

I. Introduction

Real world networks such as biological, social and technological networks are the products of an evolutionary process. These networks are generally classified as Scale Free Networks (SFN) by nature. SFNs are a class of networks in which degree distribution follows Power Law. Generative models such as Duplicate-Mutation [8], Forest Fire [2] and Preferential Attachment [1] have been proposed to synthesize SFNs. The synthesis of dynamic SFNs involves a continuous addition of new nodes to the existing network. The behavior of each new node depends on the generative model being used. It is intersting to study how nodes get assembled in complex network. Given the snapshot of a dynamic network, is it possible to probabilistically predict the evolutionary sequence of the nodes in the network?

II. Preliminaries and Notations

A. Scale Free Networks

A Scale-Free Network (SFN) is a network whose degree distribution follows a *power law*. Many real world networks are known to exhibit a decaying degree distribution. This kind of distribution is called a power law.

B. Centrality Measures

A centrality measure [5] is a function that associates a real value with each vertex in a network. The value indicates how central or important the vertex is, in the network. Here, "important" is a subjective term. This gives rise to many centrality measures, each of which rates the nodes according to some property. Some of the prominent ones include Degree Centrality [10], Eigenvector Centrality [4], [7], Betweenness Centrality [9], [6] etc.

C. Reference Network

In our experiments, we study the SFNs generated using the Barabasi-Albert Model [3]. Let $G_m(V_m, C_m)$ represent a Barabasi-Albert Network whose vertex arrival order is to be deduced. Here, V_m is the Vertex set and C_m is the number of nodes that each new node gets attached to. For evaluative purposes, we record the order of arrival of vertices in G_m during its inception. Let $list_{true}$ be a sequence of vertices that represent the actual order of arrival of vertices in G_m. We will be referring to $G_m(V_m, C_m)$ in all the further sections as the input network to the proposed algorithm that predicts order of arrival of nodes.

III. Centrality Measure based Methods

A naive approach towards the solution to the vertex arrival order prediction problem is to explore the contribution of centrality indices of the nodes. Does centrality index of the nodes help in predicting their order? If so, which type of centrality gives the most accurate result? To answer this, we start with the most intuitive of centrality measures, the Degree Centrality. From the preferential model of SFN construction, it is evident that the last few nodes that get connected to the network will have a relatively low degree, as compared to the nodes that had arrived in the initial stages. Consider the network G_m from section II-C. Intuitively, we hypothesize that higher the degree of a node earlier it might have arrived during the network evolution. Hence, we rank the nodes in the decreasing order of their degree centrality. There exists

many nodes with the same degree centrality. To predict the order amongst these nodes, we place the nodes with the same ranking into a hypothetical container, referred to as a bin. The main drawback of binning based on degree is that the degree centrality indices associated with the nodes are not distinct in G_m. Hence, binning based on degree centrality results in a small number of bins, with a large number of nodes per bin. For other centrality measures, we follow a slightly different approach that doesn't give up-to-mark results.

A. Binning Quality Measure (BQM)

Binning Quality Measure (BQM) is used to compute the accuracy of the prediction of order of arrival of nodes *across* the bins. BQM quantifies the prediction accuracy on a scale of 0 to 1. Let δ be the number of bins. Let $B = [B_0, B_1, B_2, ..., B_\delta]$ be the predicted chronological bin ordering. We associate a score β between every pair of bins. The final prediction measure η is computed as a ratio of sum of β for all bin-pairs and the total number of bin-pairs.

To calculate β for a pair of bins B_i and B_j, with $i < j$: Here, we claim that the nodes in B_i has arrived before the nodes in B_j Hence, we impose the condition $i < j$, with reference to the predicted chronological bin ordering B. For a pair of vertices $u \in B_i$ and $v \in B_j$, we define

if $index_{list_{true}}(u) < index_{list_{true}}(v)$ **then**
  $vertexOrder(u, v) = 1$
else
  $vertexOrder(u, v) = 0$
end if

$$\beta(i,j) = \frac{\sum_{u \in B_i, v \in B_j} vertexOrder(u,v)}{|B_i||B_j|}$$

The final prediction measure η is given by

$$\eta = \frac{\sum_{0<i<j\leq\delta} \beta(i,j)}{{}^{\delta}C_2}$$

IV. A New Vertex Ranking: Differential Core Ranking

In this section, we formulate a new method of ranking nodes. Let $G(V, E)$ be any graph. Let DCR_G represent the Differential Core Ranking of vertices in G. This list contains the nodes along with their Differential Core Measures in the decreasing order.

Let χ be any centrality measure. Let G_0 be the initial graph. Let G_1 be the graph obtained from G_0 after removal of nodes with the minimum degree. The change in χ centrality value of the nodes in G_0 is set as the attribute of the corresponding node. We then apply the above procedure starting with G_1. Let G_2 be the graph obtained from G_1 after the removal of nodes with the minimum degree. The change in the χ centrality value of the nodes in G_1 is added to the attribute of the corresponding node.

In general, let G_{i+1} be the graph obtained from G_i after the removal of nodes with the minimum degree. The change in the χ centrality value of the nodes in G_i is added to the attribute of the corresponding node. This procedure is repeated until there are no nodes left in G_i.

DCM_u denotes the centrality score of the node u. Higher the sum of changes in the χ centrality values of a node, higher is its importance in the network.

V. Network Reconstruction Algorithm

In this section of the paper, we describe our algorithm to predict the order of arrival of nodes in G_m. Our Algorithm is mainly divided into 4 subsections, as described below.

A. Generation of Synthetic Networks

The main focus of this section of the algorithm is to recreate the growth environment of the reference network G_m. Since the exact replication of G_m is not possible, we generate networks that are similar to G_m in certain characteristics. We refer to these set of networks as Synthetic Networks.

Let α be the number of Synthetic Networks generated. Let S_i and $chronology_{S_i}$ denote the Synthetic Network and the order of arrival of nodes in the corresponding S_i. In our experiments, we use BA model to generate S_i, with $|V_m|$ number of nodes and C_m connections. It is worth noting that every time we generate a Synthetic Network S_i, we keep track of the network growth by recording $chronology_{S_i}$. Since the Synthetic Networks are built on the same model as that of G_m, we hypothesize that the chronology of S_i is *similar* to the actual order of arrival of nodes in G_m. Hence, it is righteous to make use of $chronology_{S_i}$ in predicting the probable order of arrival of nodes in G_m.

B. Mapping and Derivation of Prediction Lists

The chronology of the Synthetic Networks S_i, where $1 \leq i \leq \alpha$, is known. In this section, we intend to derive an ordering of nodes in V_m, corresponding to each S_i. This ordering of nodes is the predicted order of arrival of nodes in G_m (during its inception), derived in accordance with $chronology_{S_i}$. We refer the node ordering corresponding to S_i as $PredList_i$.

We apply DCR, with χ as the base centrality measure, to G_m in order to obtain DCR_{G_m}. [refer to section IV] DCR_{G_m} is a list of vertex rankings sorted according to their DCM values. We apply DCR, with χ Centrality as the base centrality measure, to each S_i in order to obtain the

corresponding DCR_{S_i}.

Both DCR_{G_m} and DCR_{S_i} lists the vertices of G_m and S_i respectively in the decreasing of their importance. Earlier the position of a vertex in these lists, higher its importance in the corresponding network. A direct bijection mapping is carried out between DCR_{G_m} and DCR_{S_i}. This mapping maps the equi-important vertices in both the networks.

Mathematically, we define a mapping function as:
Let $f_{map} : V_{S_i} \rightarrow V_{G_m}$ be a direct bijection between V_{S_i} and V_{G_m}.
i.e, $f_{map}(u) = v$ where $u \in V_{S_i}, v \in V_{G_m}$ and $index_M(u) = index_N(v)$

We propose that the nodes of equal importance in G_m and S_i have the same chronological ranking. Since we know $chronology_{S_i}$, we deduce $PredList_{S_i}$ by replacing each vertex u in $chronology_{S_i}$ with $f_{map}(u)$. We repeat the above procedure for each S_i. At this stage, we have α prediction lists, denoted by $PredList_i$, each corresponding to a particular S_i.

Figures [1 to 4] illustrate an instance of Mapping of nodes between G_m and any $S_i : 1 \leq i \leq \alpha$.

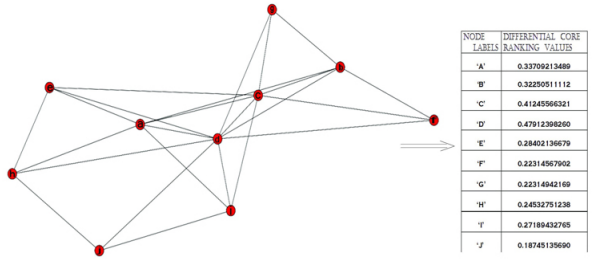

Figure 1. Applying Differential Core Ranking, with Betweenness Centrality as the base centrality, to G_m.

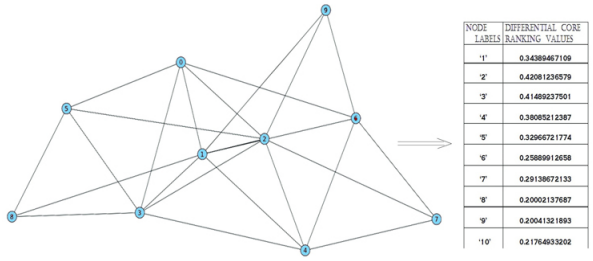

Figure 2. Applying Differential Core Ranking, with Betweenness Centrality as the base centrality, to one of the $S_i : 1 \leq i \leq \alpha$.

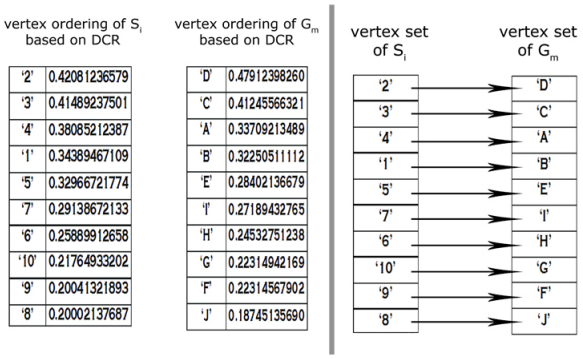

Figure 3. The diagram to the left indicates the vertex ordering based on decreasing Differential Core Ranking for V_{G_m} and V_{S_i}. The one on the right shows a direct bijection mapping of vertices between Lists.

$chronology_{S_i}$	1	2	3	4	5	6	7	8	9	10
$PredList_i$	'B'	'D'	'C'	'A'	'E'	'H'	'I'	'J'	'F'	'G'

Figure 4. Deduction of $PredList_i$ by reordering the nodes of V_m according to $chronology_{S_i}$.

C. Analysis of Prediction Lists and Construction of Directed Graph

In the previous section, we have deduced α number of Prediction Lists, $PredList_i : 1 \leq i \leq \alpha$. For every pair of vertices (u, v) , we find the order of occurrence of u and v in each $PredList_i$. Let $P_{(u,v)}$ denote the probability of u arriving before v during the inception of G_m. We compute $P_{(u,v)}$ as the fraction of the number of times u has occurred before v in the α Prediction Lists. We then construct a Directed Graph DG with vertex set $V_{DG} = V_m$, and an initial edge set $E_{DG} = \phi$. A directed edge from u to v in DG indicates that u has arrived before v during the construction of G_m.

For a pair of vertices (u, v):
if $P_{(u,v)} > 0.5$, then we say that u has arrived before v with a probability $P_{(u,v)}$. We put a directed edge from u to v with a weight $P_{(u,v)}$.
if $P_{(u,v)} < 0.5$, then we say that v has arrived before u with a probability $1 - P_{(u,v)}$. We put a directed edge from v to u with a weight $1 - P_{(u,v)}$.

D. Transformation of Directed Graph and Node Binning

In this section, we process DG obtained from the previous section to deduce the final prediction of order of arrival of nodes in G_m. But there is a fair possibility that DG can be a cyclic graph, which can make the prediction order ambiguous. Hence we intend to transform it into a Directed Acyclic Graph (DAG).

Input: Directed Graph DG.

Output: Directed Acyclic Graph DAG.
while DG contains cycles **do**
Remove the edge (u, v) with the least $P_{(u,v)} : (u, v) \in E_{DG}$.
end while

Ideally, the node that had arrived earliest should have zero in-degree. The next earliest node should have an in-degree equal to 1 and so on. Since we are probabilistically simulating the growth environment of G_m, this is not the case.

In the final step binning, we will find all the vertices v in DAG having the least in-degree and bunch them into a bin. The binned vertices are hypothesized to have arrived first and are removed from DAG. Later we iterate this process over till there are no nodes left in DAG. We obtain Final predicted bin ordering.

Algorithm to bin the nodes from DAG is presented below:

Input: Directed Acyclic Graph DAG
Output: Bin Ordering
$count \leftarrow 1$
while $|V_{DAG}| \neq 0$ **do**
$minInDeg \leftarrow arg\ min(InDegree(u))\ where\ u \in V_{DAG}$
Let $B_{count} \leftarrow \{u : \forall u \in V_{DAG}$ and $InDegree(u) = minInDeg\}$
Remove all the nodes in B_{count} from V_{DAG} i.e, $V_{DAG} \leftarrow V_{DAG} - B_{count}$
$Count \leftarrow Count + 1$
end while
Let $binOrdering \leftarrow [B_1, B_2, B_3, ...B_{Count}]$

$binOrdering$ gives the predicted chronological sequence of bins. The accuracy of this prediction, in contrast with accuracy of prediction using centrality measures, is discussed in the next section.

VI. Results and Discussions

A. Comparison between the predictions from Differential Core Ranking and Plain Centrality

Let χ be a base centrality measure. Let $Plain\chi_{G_m}$ denote the vertex ordering in the descending order of their χ centrality values. We apply DCR, with the same centrality χ as the base centrality, to the network G_m. Let $Differential\chi_{G_m}$ denote the vertex ordering in the descending order of their DCM values.

$list_{true}$ denotes the actual order of arrival of nodes in G_m (section II-C). Let the predicted order be denoted by $list_{pred}$. To compute the accuracy of our prediction, we define a new quality measure called $\eta(list_{true},\ list_{pred})$.

$$\eta(list_{true}, list_{pred}) = \frac{n_c}{|V_{G_m}|_{C_2}}$$

where n_c is the number of pairs in $list_{pred}$ that are in correct relative order with respect to $list_{true}$. To compare the prediction accuracy for the lists $Plain\chi_{G_m}$ and $Differential\chi_{G_m}$, we just compare the values of $\eta(list_{true}, Plain\chi_{G_m})$ and $\eta(list_{true}, Differential\chi_{G_m})$. In our experiments we consider the cases where χ represents Degree Centrality and Betweenness Centrality.

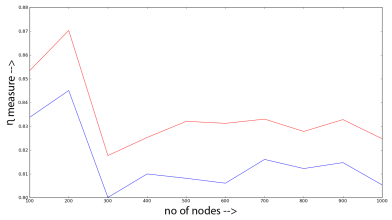

Figure 5. Plot representing the comparision of values of $\eta(list_{true}, Plain\chi_{G_m})$ (blue plot) and $\eta(list_{true}, Differential\chi_{G_m})$ (red plot) for varying number of nodes with χ : Degree Centrality

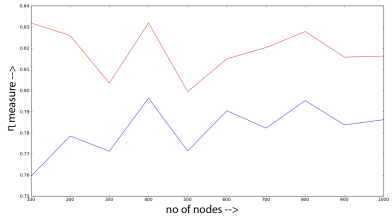

Figure 6. Plot representing the comparision of values of $\eta(list_{true}, Plain\chi_{G_m})$ (blue plot) and $\eta(list_{true}, Differential\chi_{G_m})$ (red plot) for varying number of nodes with χ : Betweenness Centrality

B. Prediction of arrival order in every node pair with an attached probability

We now present the analytical results that we have obtained, considering G_m as reference network. We have generated G_m using a BA model with 1000 nodes and 3 connections. We generate 50 synthetic networks. So, we set $\alpha = 50$. The analytical results thus obtained is given below:

R=range of $P_{(u,v)}$	No of edges whose $P_{(u,v)} \in R$ / $\lvert E_{DG} \rvert$	Fraction of pairs in the correct relative order with $list_{true}$
(0.5, 0.6]	0.216606606607	0.546827487407
(0.6, 0.7]	0.156592592593	0.652739778568
(0.7, 0.8]	0.137975975976	0.767770861446
(0.8, 0.9]	0.156284284284	0.864482988317
(0.9, 1.0]	0.308434434434	0.967824850873

Statistically, from the above table, we observe that the edges (u, v) having $P_{(u,v)}$ in $(0.5, 0.6]$ constitute around 20% of the edges. We also note that only around 50% of these edges are in the correct relative order with $list_{true}$. Since a large fraction of edges belonging to this range are in incorrect relative ordering, they contribute to the

cycle formation. Cycles introduce inconsistencies in node arrival order, hence they have to be removed. From our experiments, we have found out that DG will become acyclic when we remove the edges (u, v) continually in the increasing order until $P_{(u,v)} \approx 0.6$. We implement the same technique in section V-D to transform DG to DAG.

Based on the facts and figures from the table, we observe that the fraction of pairs that are in correct relative order with $list_{true}$ increases as the sampled range increases. Hence we conclude that, higher $P_{(u,v)}$ implies a stronger notion of relative ordering of (u, v).

C. Comparison between the predictions from DCR binning and Plain Centrality binning

The end result of our method (section V-D) is the ordering of the bins, referred to as $binOrdering_{DCR\chi}$. Let $\eta_{DCR\chi}$ denote the BQM score of $binOrdering_{DCR\chi}$, where χ refers to the base centrality measure for DCR.

Let $binOrdering_{\chi}$ denote the chronology of bins with χ as the base centrality. $binOrdering_{betweenness}$, $binOrdering_{eigen}$ and $binOrdering_{degree}$ denote the chronology of bins with χ set as Betweenness, Eigenvector and Degree Centralities respectively.

Let $\eta_{betweenness}$, η_{eigen} and η_{degree} denote the BQM scores of $binOrdering_{betweenness}$, $binOrdering_{eigen}$ and $binOrdering_{degree}$ respectively. Finally, we compare $\eta_{betweenness}$, η_{eigen}, η_{degree} and $\eta_{DCR\chi}$ where χ is the base centrality (refer section 4).

We perform the above said experiment multiple times for the reference graphs G_m of 1000 nodes and 3 connections. In our experiment, we have set $\alpha = 50$. For each experiment, we choose different base centralities and different G_m. We observe that the DCR method yields more accurate results compared to any other plain centrality based approaches. Figures 7 and 8 represents two of those instances and denotes the BQM scores for various binning methodologies.

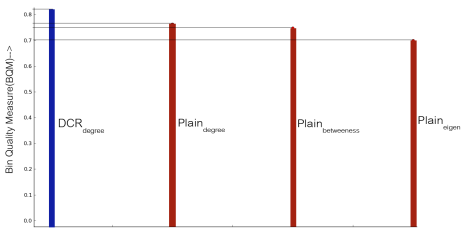

Figure 7. $\eta_{DCR_{degree}} = 0.804513946531$, $\eta_{degree} = 0.767615011251$, $\eta_{betweenness} = 0.759827243464$, $\eta_{eigen} = 0.695466553648$, number of bins=91

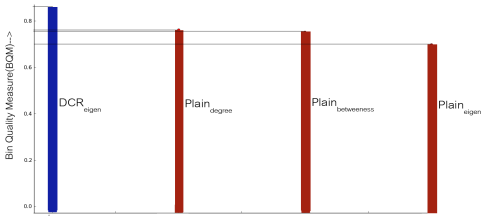

Figure 8. $\eta_{DCR_{eigen}} = 0.84654821986$, $\eta_{degree} = 0.7697124538121$, $\eta_{betweenness} = 0.753169421166$, $\eta_{eigen} = 6899122714632$, number of bins=77

VII. Conclusion

We presented a novel framework for uncovering the precursor of a SFN evolved by BA model. Our approach involves the synthesis of many such SFNs, mapping these SFNs with the reference network based on DCR score associated with the nodes and arriving at the final prediction order. We presented 3 results. 1. DCR based prediction, which proved to provide better predicted node arrival results than any other centrality based approaches. 2. Arrival order of every pair of nodes in a SFN, with an associated probability. We empirically proved that most of the node pairs with high probability indeed arrived in the order that we predicted. 3. We also proved that DCR based prediction, when applied in conjunction with the binning methodologies, offered a better accuracy compared to any other plain centrality based approaches.

References

[1] R. Albert and A. L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, January 2002.

[2] J.-D. Bancal and R. Pastor-Satorras. Steady-state dynamics of the forest fire model on complex networks. *The European Physical Journal B - Condensed Matter and Complex Systems*, 76:109–121, 2010. 10.1140/epjb/e2010-00165-7.

[3] A. L. Barabási and R. Albert. Emergence of Scaling in Random Networks. *Science*, 286(5439):509–512, October 1999.

[4] P. Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1):113–120, 1972.

[5] S. Borgatti and M. Everett. A graph-theoretic perspective on centrality. *Social Networks*, 28(4):466–484, October 2006.

[6] U. Brandes. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25:163–177, 2001.

[7] U. Brandes and T. Erlebach. Network Analysis. Methodological Foundations. *Network Analysis, Lecture Notes in Computer Science*, 3418, 2005.

[8] T. Gregory Dewey Fan Chung, Linyuan Lu and David J. Galas. Duplication models for biological networks. *Journal of Computational Biology.*, pages 677–687, October 2003.

[9] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.

[10] L. C. Freeman. Centrality in social networks: Conceptual clarification. *I. Social Networks*, 1:215–239, 1979.