NONUNIVERSALITY IN SEMI-DIRECTED BARABÁSI–ALBERT NETWORKS

M. A. SUMOUR* and M. A. RADWAN†

Physics Department, Al-Aqsa University
P. O. 4051, Gaza, Gaza Strip, Palestinian Authority

*msumo@msumoor@yahoo.com
†ma.radwan@yahoo.com

Received 7 May 2012
Accepted 20 July 2012
Published 27 August 2012

In usual scale-free networks of Barabási–Albert type, a newly added node selects randomly \( m \) neighbors from the already existing network nodes, proportionally to the number of links these had before. Then the number \( n(k) \) of nodes with \( k \) links each decays as \( 1/k^{\gamma} \) where \( \gamma = 3 \) is universal, i.e. independent of \( m \). Now we use a limited directedness in building the network, as a result of which the exponent \( \gamma \) decreases from 3 to 2 for increasing \( m \).

Keywords: General BA; directed BA; undirected BA; neighbors; Fortran programs.

PACS Nos.: 05.10.-a, 05.10. Cc.

1. Introduction

The Barabási–Albert network is growing such that the probability of a new site to be connected to one of the already existing sites is proportional to the number of previous connections to this already existing site: The rich get richer. In this way, each new site selects exactly \( m \) old sites as neighbors.

In directed Barabási–Albert networks,\(^1\) the network itself was built in the standard way, but when agents (spins) were put on the network nodes, the neighbor relations were such that if A has B as a neighbor, B in general does not have A as a neighbor.\(^2\)

The undirected Barabási–Albert network\(^3,4,5\) usually is grown in the same way, but then the neighbor relations were such that if A has B as a neighbor, B has A as a neighbor.\(^2\)

Now we introduce a directedness already when building the network.\(^3,4,5\) In the undirected Barabási–Albert network,\(^5,6\) if a new node selects \( m \) old nodes as neighbors, then the \( m \) old nodes are added to the Kertész list, and the new node is also added \( m \) times to that list.

\(^a\)There are also other network papers in the Science issue (24 July 2009).
Also in our semi-directed version, the new node makes connections with \( m \) randomly selected old nodes of the Kertész list. If one would only add the old nodes to the list, then only the initial core can be selected as neighbors, which is less interesting. But if one adds the \( m \) old nodes, plus only once (and not \( m \) times) the new node, then one has what we call here a semi-directed network. We deal here only with its structure, not with agents put onto its nodes. In this semi-directed Barabási–Albert model one can put in neighbor relations which are directed or undirected, for agents (spins) put onto the network nodes; but we do not do that here.

2. Data and Simulation

We use the FORTRAN program\(^7\) as in our appendix, with different \( m = 1, 2, 3, \ldots, 10 \), and the number of nodes (\texttt{maxtime}) = \( 4 \times 10^7 \) nodes.

First we calculate the degree distribution \( n(k) \). Figure 1 shows this number \( n(k) \) of nodes with \( k \) links each. Changes are seen with increasing number of neighbors \( m \), and the curvature shows that the exponent should be determined from the region of large \( k \) only.

Second, because of the strong scattering for large \( k \) or small \( n(k) \) we bin these raw data by powers of two, e.g. \( k = 4, 5, 6, 7 \) is summed to give one of the many data points (placed at the geometric means of the binned intervals) onto which we fitted the exponents \( \gamma(m) - 1 \), we use binned (summed) data from \( n(k) \) to plot double-logarithmically for \( m = 10 \) as we will show in Fig. 2.

Third, we plot the resulting slopes \( \gamma(m) - 1 \) versus \( 1/m \) to get Fig. 3, which makes clearer the possible extrapolation towards infinite \( m \) (\( m = \infty \), \( 1/m = 0 \)). May be the true exponents \( \gamma(m) \) equal \( 2 + 1/m \) since \( m = 1 \) should give the standard (undirected) exponent \( \gamma = 3 \). The deviations from this formula (curve in Fig. 3) are

![Distribution of n(k) of nodes with k links](image)

Fig. 1. Log–log plot of \( n(k) \) vs \( k \) for \( m = 1(+) \), \( 2(\times) \), and \( 10 (*) \), from left to right. The slopes of the straight lines approximating the data give the exponents \( \gamma(m) \) and vary with \( m \): Nonuniversality.
not larger than our systematic errors. As an alternative to the linear behavior also a power-law fit to $m > 1$ is shown. We see that the new power-law fit agrees very nicely with the data except for the standard BA model (undirected) case $m = 1$.

3. Conclusions

The exponent $\gamma$, for the decay of the number $n(k)$ of nodes with $k$ links each, was shown to change with increasing number of neighbors $m$. It remains a challenge to explain this nonuniversal variation with $m$, perhaps linear in $1/m$. 

Fig. 2. Distribution of binned $n(k)$ with $k$ links ($m = 10$, slope = 1.16), with fitting line (50 000 0000/ $k^{1.16}$).

Fig. 3. Slopes $\gamma - 1$ vs $1/m$ for $m = 1, 2, 3, \ldots, 10$, with power-law fit (0.96 + 0.80 $x^{0.59}$).
Acknowledgments

The authors thank D. Stauffer for many valuable suggestions, fruitful discussions and constructive advice during the development of this work, and the referee for useful help with the manuscript.

Appendix

parameter(nrun=1, maxtime=40000000 ,m=2,iseed=1,max=maxtime+m, 1   length=1+(1+m)*maxtime+m*(m-1))
integer*8 ibm
real*8 factor
dimension k(max), nk(131070), list(length), nklog(30)
data nk/131070*0/,nklog/30*0/
OPEN (UNIT=60,FILE='N(K)M2.DAT')
WRITE (60,*)'##(nrun, maxtime, m, iseed)',nrun, maxtime, m,iseed
ibm=2*iseed-1
factor=(0.25/2147483648.0d0)/2147483648.0d0
fac=1.0/0.69314
factor=0.5/2147483648.0d0
do 5 irun=1,nrun
do 3 i=1,m
   do 7 j=(i-1)*(m-1)+1,(i-1)*(m-1)+m-1
       list(j)=i
3    k(i)=m-1
   L=m*(m-1)
   if(m.eq.1) then
       L=1
       list(1)=1
   endif
   c All m initial sites are connected with each other
   do 1 n=m+1,max
      do 2 new=1,m
         ibm=ibm*16807
         j=1+(ibm*factor+0.5)*L
         if(j.le.0.or.j.gt.L) goto 4
         j=list(j)
         list(L+new)=j
2      k(j)=k(j)+1
      list(L+m+1)=n
      L=L+m+1
1      k(n)=1
   WRITE (60,*, '# (irun )', irun
C print *,# (irun ), irun
  do 5 i=1,max
    k(i)=min0(k(i),131070)
  5  nk(k(i))=nk(k(i))+1
  do 6 i=1,131070
  6  if(nk(i).gt.0) write (60,*), i,nk(i)
  do 9 i=1,131070
    j=alog(float(i))*fac
  9  nklog(j)=nklog(j)+nk(i)
  do 10 j=0,18
  10 write (60,*), sqrt(2.0)*2**j,nklog(j),j
stop
end

This program follows the program previously published in Ref. 8. Again the Kertész list is denoted by list here. For the usual (undirected) Barabási–Albert network, at and after label 2 one has:

2  k(j)=k(j)+1
  do 8 j=1,m
  8  list(L+m+j)=n
    L=L+2*m
  1  k(n)=m

These lines are now simplified to

2  k(j)=k(j)+1
  list(L+m+1)=n
    L=L+m+1
  1  k(n)=1

So, instead of listing $m$ times the new node $n$, it is listed only once, and thus the current length $L$ of the Kertész list is increased by only $m+1$ instead of $2m$.

The array nklog gives the binned sums, with bin boundaries for $k$ increasing as powers of two, in order to reduce the fluctuations in $n(k)$; it has nothing to do with the present semi-directedness.

References
M. A. Sumour & M. A. Radwan

