Al-Aqsa University Journal (Natural Sciences Series),Vol.19, No.1, Pages 50-62,Jan.2015
ISSN 2070-3155

Distribution of number of neighbors on semi-directed Barab'asi-Albert networks with many initial neighbors

Dr. M. A. Sumour *
Dr. F. W. S. Lima**
Dr. M. A. Radwan*
Dr. M. M. Shabat***

الملخص

توزيع عدد من الجيران على شبكات بربزاي- ألبرت شبه المباشرة مع العديد من الجيران المتقاربين

على شبكات بربزاي – ألبرت المباشرة وغير المباشرة، عند اختيار عقدة جديدة يكون لها عدد m عقدة كجيران، عندئذ العقد القديمة m تضاف على قائمة كيرتز والعقد الجديدة تضاف عدد m من المرات إلى هذه القائمة. هذه الروابط تتم مع عناصر مختارة بطريقة عشوائية من هذه القائمة. و عند إضافة العقد القديمة m للقائمة بالإضافة إلى ذلك العقدة الجديدة مرة واحدة وليس m من المرات، نحصل على شبكة غير مباشرة SDBA.

عند اختيار عدد الجيران على نسختين من SDBA1 وSDBA2 لشبكات بربزاي– ألبرت غير المباشرة، وجدنا ان SDBA2 لا تعطي القوانين المناسبة لعدد كبير من الجيران m ولكن تعطي قوانين مناسبة لعدد صغير من الجيران m ، بينما SDBA1 يعطي قوانين مناسبة لعدد كبير وصغير من الجيران mالنتائج الاسية التي تم الحصول عليها تتغير مع عدد الجيرانm .

Abstrct

On directed and undirected Barba'asi-Albert networks, when a new node has selected m old nodes as neighbors, then the m old nodes are added to the Kert'esz list, and the new node is also added m times to that list. These connections are made with m randomly selected elements of that list. If one adds to the list the m old nodes, plus only once and not m times the new node, one gets a semi-directed network (SDBA). Now we check the number of neighbors on two versions (SDBA1 and SDBA2) for
semi-directed Barba'asi-Albert networks.

We found that SDBA2 does not give proper power laws for large m but it does so for small m, whereas SDBA1 gives proper power laws for large and small m. The resulting exponents vary with m.

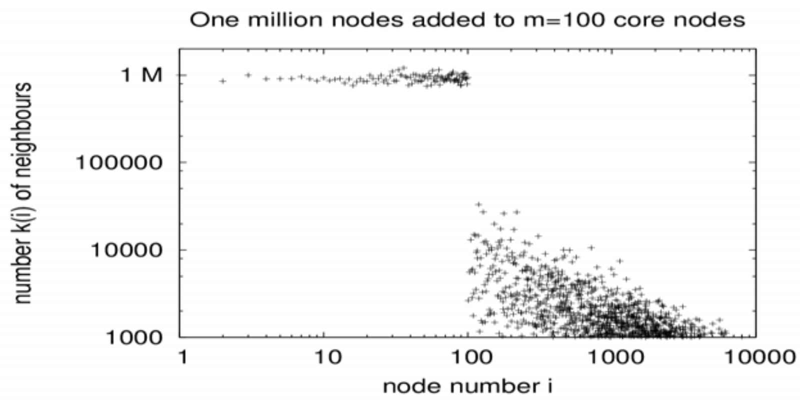
Keywords: Directed BA networks , Undirected BA network , Semi-directed BA network.

* Physics Department, Al-Aqsa University, Gaza, Gaza Strip, Palestinian Authority.
** Dietrich Stauffer Computational Physics Lab, Departamento de F´ısica, Universidade Federal do Piauı, 64049-550, Teresina - PI, Brazil.
***Physics Department, Islamic University P.o.Box 108, Gaza, Gaza Strip, Palestinian Authority.

1.Introduction:

The Barba'asi-Albert network [1,2]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 gets richer. In this way, each new site selects exactly m old sites as neighbors. In directed (DBA) and undirected(UBA)Barba'asi-Albert networks, the network itself was built in the standard way, but when agents (spins)were put on the network nodes[2,3,4]the neighbor relations were such that if A has B as a neighbor, B in general does not have A as a neighbor for DBA while it does have for UBA.

The present work continues the study of two semi-directed BA networks, SDBA1[5]and SDBA2[6]for much larger m than before. In these semi-directed networks the exponent **γ** for the power law governing the decay of the number n(k)of nodes having k neighbors, $n(k) \alpha\ 1/k^{\gamma}$, depends continuously on the parameter m, and we want to know its behavior for $m \to \infty$.

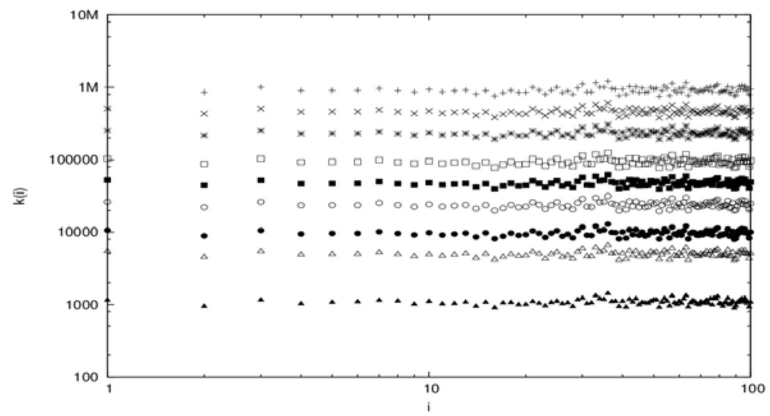


Figure(1):Number of neighbors k(i)influencing node (i) in SDBA2 networks with m=100 and N = 1 million.

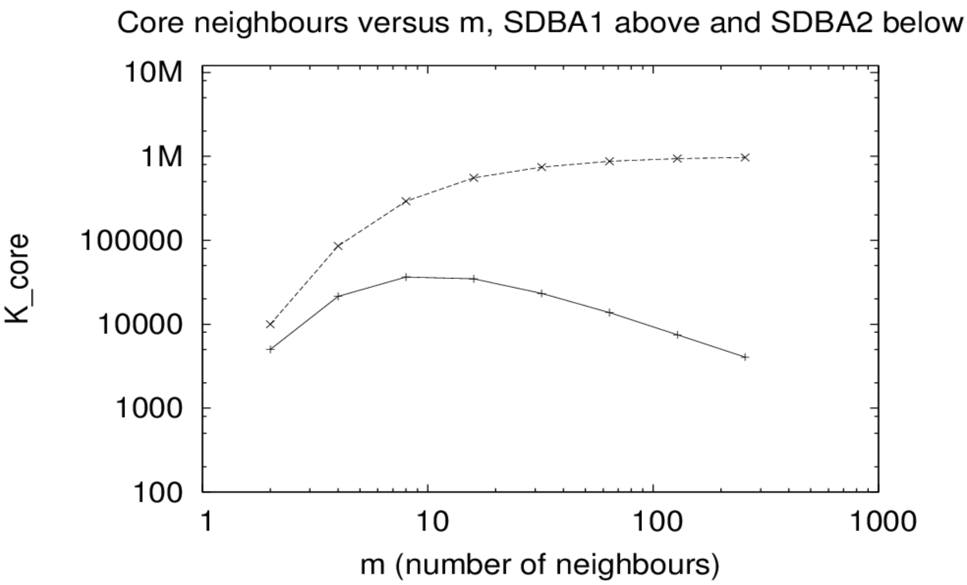
2. Models and Simulations:

2.1Directed(DBA)and undirected(UBA)Barba'asi-Albert network:

In the Barabasi-Albert network [1,2], if a new node selects m old nodes as neighbor, then the m old nodes are added to the Kert'esz list, and the new node is also added m times to that list. Connections are made with m randomly selected elements of that list.

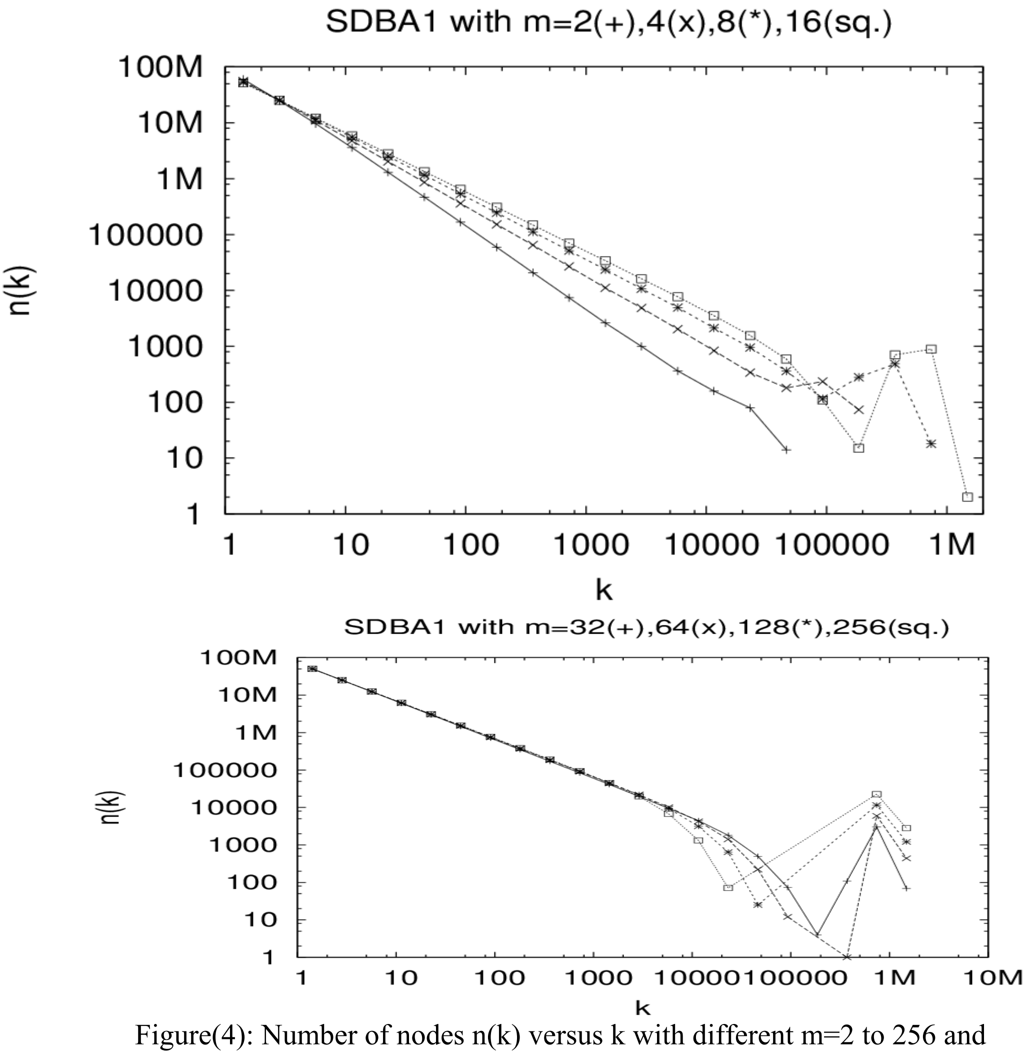


Figure(2): Number of neighbors k(i) versus the index of core nodes, at constant m=100 and different N

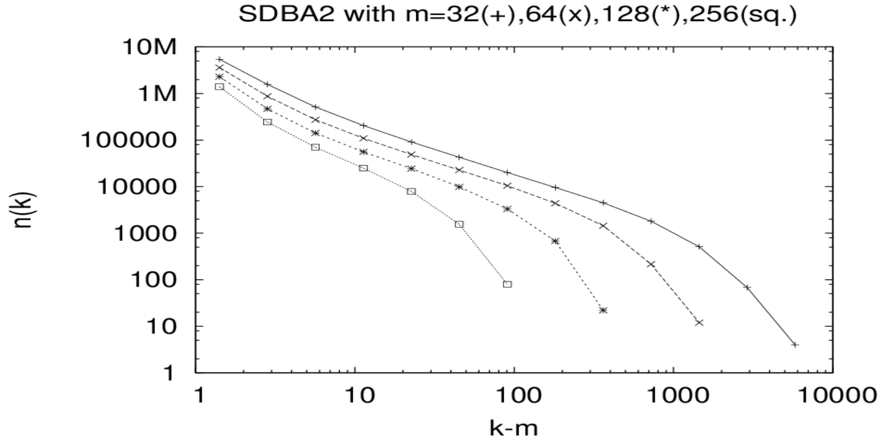


Figure(3): Average K_core versus m=2,4,8,16,32,64,128,and 256 with constant N=1,000,000 for SDBA1 and SDBA2.

In directed Barba'asi-Albert networks DBA, the network itself was produced in the standard way, but then the neighbor relations were such that if A has B as a neighbor, B in general does not have A as a neighbor. The undirected Barabasi-Albert network UBA 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. The growth process always starts with a fully connected core of m nodes.

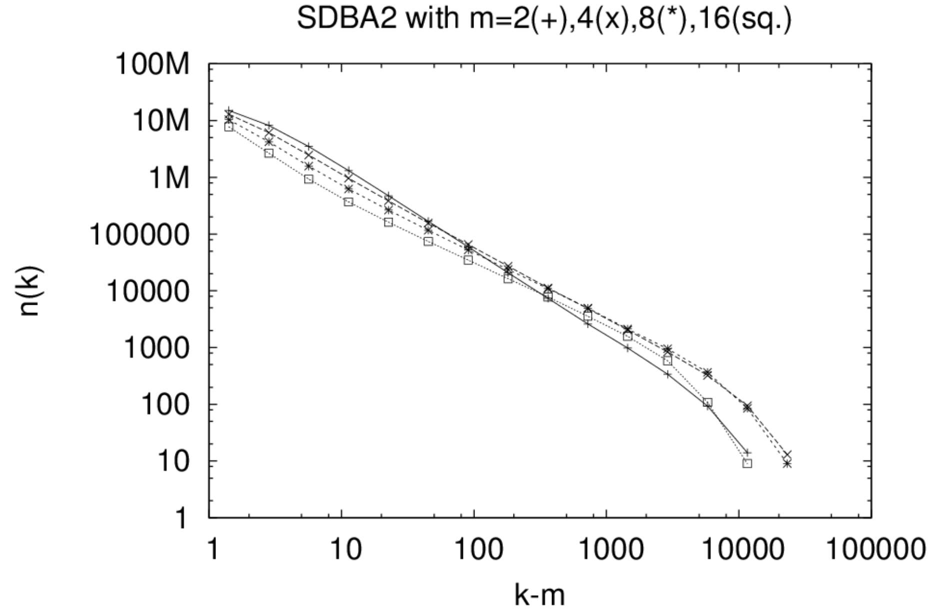


Figure(4): Number of nodes n(k) versus k with different m=2 to 256 and N=1,000,000for SDBA1.



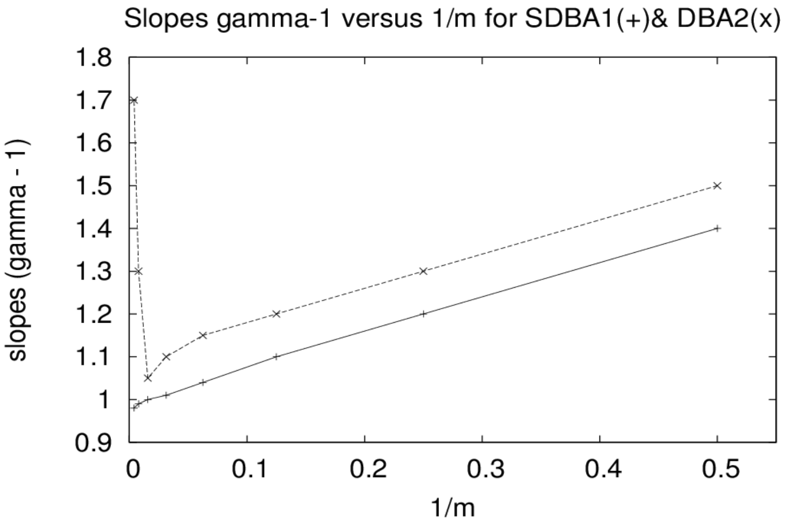
2.2 Semi-Directed Barba'asi-Albert network (SDBA):

If in building a BA network one would only add the old nodes to the Kert'esz list after a new node was added, then only the initial core can be selected as neighbors, which is not interesting. But if one adds to the list the m old nodes, plus only once and not m times the new node, one has a semi-directed network (SDBA).



Figure(5): Number of nodes n(k) versus k-m with different m=2 to 256 and N=1,000,000 for SDBA1.

The first version SDBA1 builds the network in the way of [5]. The new node selects m sites j which n will all influence, while n will be influenced only by the first selected j. Our second version SDBA2 inverts the direction of the spin interaction[6]:The new node n selects m sites j which will all influence n, while n will influence only the first selected j. Now we look at both SDBA1 and SDBA2 in the hope of finding the limit m $\rightarrow \infty$, more precisely for $1 << m << N$.



Figure(6): Plot of exponents γ -1 (slopes) versus 1/m for the SDBA1 and SDBA2 with m=2,4,8,16,32,64,128, and 256.

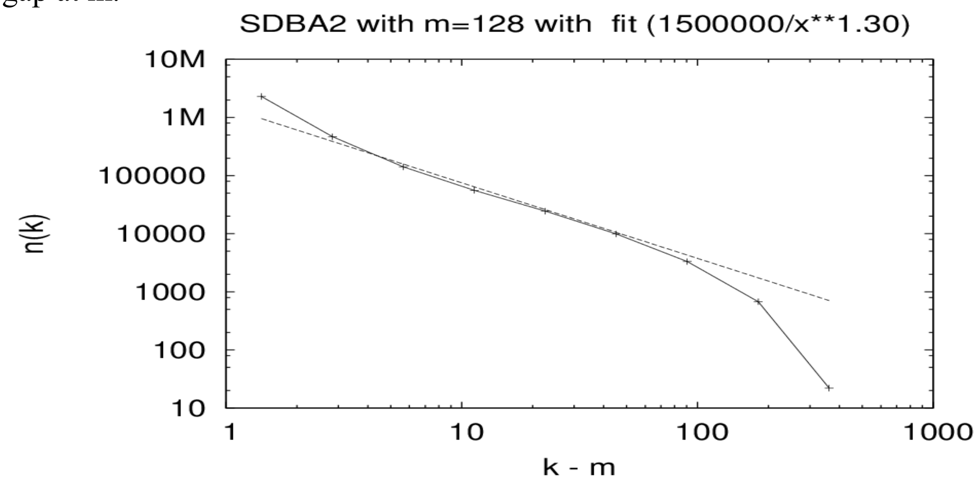
3. Results and discussion:

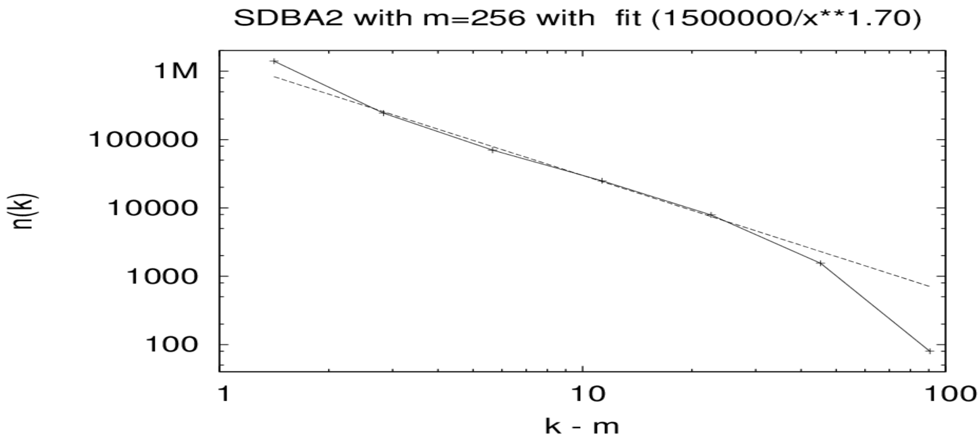
We used a simulation program that include two different programs Called SDBA1 and SDBA2 networks as shown in appendix (1-2). We determined powers of two and store in variable m=2,4,8,16,32,64,128, and 256, and we fixed the network size (called maxtime in the programs) N=1,000,000 and fixed an upper bond k_b=9 million for k variable. First we measured the distribution nodes of n(k) number influenced by k neighbors in both programs SDBA2 and SDBA1 we concluded that both programs are support the result in [5]. The program SDBA2 shown a clear gap in k(i) number of neighbors, as shown in figure(1).

The new nodes n added to the initial core of m sites have at least k(n)=m (here m=100) neighbors (see appendix 2 at label 1) most of result from the m core nodes, shown an important difference gap between SDBA1 and

SDBA2. Based on our simulation study for different sizes of lattice, by varying lattice size N (1,000,000, 500,000, 250,000, 100,000, 50,000, 10,000, 5,000, and 1,000) fixed m=100, the number of neighbors k(i) for the core nodes is shown in Figure(2).

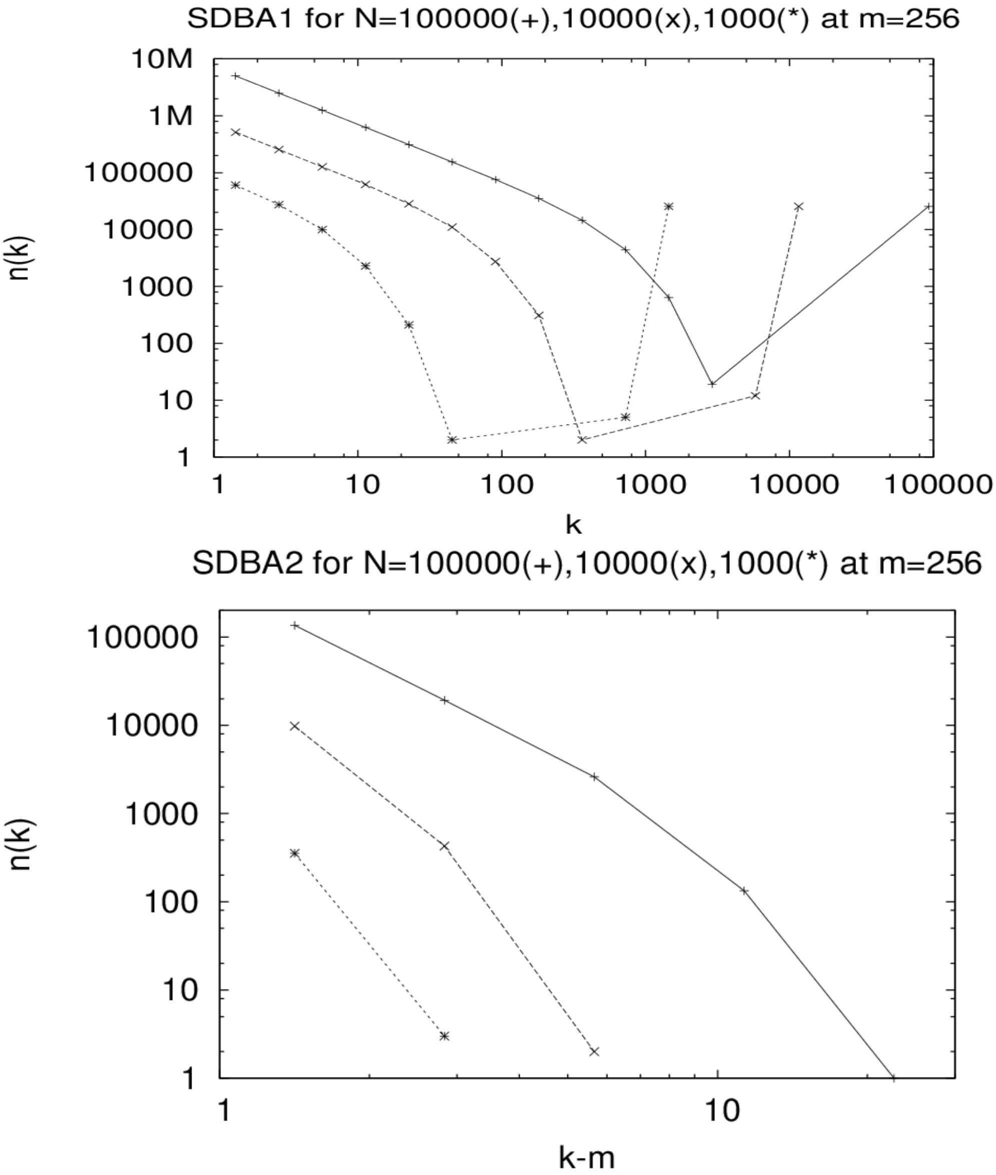
To plot the histograms n(k) of the numbers k of neighbors influencing a node, we bin them in powers of two. For SDBA2, because of the above gap, we count instead of k the number k-m of additional neighbors gained after the addition of a node to the network, in order to get smoother histograms. This means for SDBA2 we use only the right part of Figure(2) beyond the gap at m.





Figure(7): Number of nodes n(k) versus k-m links with m=128 and 256, at N=1,000,000 for SDBA2

In contrast, for SDBA1 these straight lines correspond to power laws without such a gap problem, we counted all k as usual and not the difference k-m.



Figure(8): Histograms of n(k) at smaller network sizes N for SDBA1 (top) and SDBA2 (bottom).

The average number k of neighbors for the m core nodes:

$$K_{core} = \sum_{i=1}^{m} k(i)/m,$$

is plotted in Figure(3),averaged over 100 samples. All following figures are also averaged over 100 samples.

As shown in figures(4 and 5),we can determine a nice slopes in log-log plots of n(k) versus k for SDBA1, while for large m, bad power law is seen for SDBA2. These slopes are summarized in Figure(6), which is the essence of main result of the research. The steep increase for SDBA2 near 1/m=0 is unreliable and due to the fact that Figure(5 bottom)does not give good power laws.

To see the problems for SDBA2 at large m we plotted as examples m=128 and 256 which resulted a bad straight line, thus we fit the curves by $1500000/x^{1.3}$ for m=128, and by $1500000/x^{1.7}$ for m=256, as listed in figure(7).

The last figure(8) shown the strong influence of the finite network size N on the tails of the histogram, in both SDBA1 and SDBA2. for both SDBA1 and SDBA2, only for large networks used in Figures (1,3,4,5) with N = 1 million do we get good power laws.

4. Conclusion:

We concluded that SDBA2 the initial m core nodes have anomalous neighborhoods and that SDBA2 does not given proper power laws for large m but does so for small m. In contrast, SDBA1 given a nice power laws and its exponent γ-1 approaches unity linearly in 1/m, meaning n(k) α $1/k^2$ for large m, as opposed to $1/k^3$ for standard BA networks at all m. In both SDBA1 and SDBA2 the exponents γ are not universal and a theoretical explanation for this continuous variation of the exponent should be searched for. The lack of good power laws for SDBA2(not SDBA1)might vanish if much larger networks can be simulated.

Acknowledgments:

The authors are grateful to Dietrich Stauffer for stimulating discussions and for a critical reading of the manuscript. F. W. S. L.acknowledges the Brazilian agency FAPEPI(Teresina-Piaui-Brasil) and CNPQ for its financial support and. This work also was supported the system SGI Altix 1350the computational park CENAPAD.UNICAMP-USP, SP-BRAZIL and Dietrich Stauffer Computational Physics Lab-TERESINA-PIAU\'I-BRAZIL.

References:

[1] A.-L. Barab´asi, R. Albert, Science 286, 509 (1999).

[2] A. Aleksiejuk, J.A. Ho lyst, D. Stauffer, Physica A 310, 269 (2002) .

[3] M. A. Sumour, M.M. Shabat, Int. J. Mod. Phys. C 16, 585 (2005).

[4] M. A. Sumour, M.M. Shabat, D. Stauffer, Islamic University Journal (Gaza) 14, 209 (2006). cond-mat/0504460 at www.arXiv.org.

[5] M. A. Sumour and M. A. Radwan, Int. J. Mod. Phys. C 23, article 1250062 (2013).

[6] M. A. Sumour and F. W. S. Lima, preprint .

Appendix(1):

Fortran program for Ising model on SDBA1 given in [5],without spins.

```
   parameter(nrun=100,maxtime=1000000,m=64,iseed=2,max=maxtime+m,
    1   kb=9000000,length=1+(1+m)*maxtime+m*(m-1))
      integer*8 ibm
      real*8 factor
      dimension nklog(0:30)
      dimension k(max), nk(kb), list(length)
      data nk/kb*0/,nklog/31*0/
      OPEN (UNIT=60,FILE='SDBA1-1NmA64.DAT')
      OPEN (UNIT=70,FILE='SDBA1-2mA64.DAT')
      WRITE (60,*)'#(nrun, maxtime, m, iseed)',nrun, maxtime,
      m,iseed
      ibm=2*iseed-1
      factor=(0.25d0/2147483648.0d0)/2147483648.0d0
      fac=1.0/0.69314
      do 5 irun=1,nrun
      do 3 i=1,m
      do 7 j=(i-1)*(m-1)+1,(i-1)*(m-1)+m-1
7      list(j)=i
3      k(i)=m-1
      L=m*(m-1)
      if(m.eq.1) then
      L=1
      List(1)=1
      k(1)=1
      endif
C     All m initial sites are connected with each other
      do 1 n=m+1,max
```

```
      do 2 new=1,m
4      ibm=ibm*16807
      j=1.d0+(ibm*factor+0.5d0)*L
      if(j.le.0.or.j.gt.L) goto 4
      j=list(j)
      list(L+new)=j
2      k(j)=k(j)+1
      L=L+m+1
      list(L)= n
1      k(n)=1
      write(60,*),'#(irun)', irun
      do 5 i= 1,max
5      nk(k(i))=nk(k(i))+1
      SUM=0
      do 6 i=1,m
      SUM=SUM+K(i)
6      write(60,*) i,k(i)
      do 9 i=1,kb
      j=alog(float(i))*fac
9      nklog(j)=nklog(j)+nk(i)
      AVERGESUM=SUM/m
      jmax=(1.0 + alog(float(kb))*fac)
      do 10 j=0,jmax
      if(nklog(j).ne.0)
       write(70,*)sqrt(2.0)*2**j,nklog(j),j,AVERGESUM
10    if(nklog(j).ne.0)print*,sqrt(2.0)*2**j,nklog(j),
     1 j,AVERGESUM
      stop
      end
```

Appendix(2):

Fortran program for Ising model on SDBA2 given in[6], without spins.

```
          parameter(nrun=100,maxtime=1000000,m=64,iseed=2,
1         max=maxtime+m kb=9000000,length=1+
     1     (1+m)*maxtime+m*(m-1))
      integer*8 ibm
      real*8 factor
      dimension nklog(0:30)
      dimension k(max), nk(kb), list(length)
      data nk/kb*0/,nklog/31*0/,k/max*0/
```

Distribution of number of neighbors ….

```
      OPEN (UNIT=60,FILE='SDBA2-1M64.DAT')
      OPEN (UNIT=70,FILE='SDBA2-2M64.DAT')
      WRITE (60,*)'#(nrun, maxtime, m, iseed)',nrun,
     1  maxtime,   m,iseed
      ibm=2*iseed-1
      factor=(0.25d0/2147483648.0d0)/2147483648.0d0
      fac=1.0/0.69314
      do 5 irun=1,nrun
      do 3 i=1,m
      do 7 j=(i-1)*(m-1)+1,(i-1)*(m-1)+m-1
7      list(j)=i
3      k(i)= m-1
      L= m*(m-1)
      if(m.eq.1) then
      L=1
      List(1)=1
      k(1)=1
      endif
C      All m initial sites are connected with each other
      do 1 n=m+1,max
      do 2 new=1,m
4      ibm=ibm*16807
      j=1.d0+(ibm*factor+0.5d0)*L
      if(j.le.0.or.j.gt.L) goto 4
      j=list(j)
      k(n)=k(n)+1
      if(new.eq.1) k(j)=k(j)+1
      if(k(j).gt.kb .or. k(n).gt.kb) stop 9
      list(L+new)=j
2      continue
      L=L+m+1
      List(L)=n
1      k(n)=m
      write(60,*) '# (irun )', irun
      do 5 i= m+1,max
5      if(k(i).gt.m) nk(k(i)-m)=nk(k(i)-m)+1
      SUM=0
```

```
      do 6 i=1,m
      SUM=SUM+K(i)
6      write(60,*) i,k(i)
      do 9 i=1,kb
      j=alog(float(i))*fac
9      nklog(j)=nklog(j)+nk(i)
      AVERGESUM=SUM/m
      jmax=(1.0 + alog(float(kb))*fac)
      do 10 j=0,jmax
      if(nklog(j).ne.0)
      write(70,*)sqrt(2.0)*2**j,nklog(j),j,AVERGESUM
10     if(nklog(j).ne.0)print*,sqrt(2.0)*2**j,
     1 nklog(j),j,AVERGESUM
      stop
      end
```