Describing network evolution using probabilistic inductive classes

Vladimir Batagelj

IMFM Ljubljana and IAM UP, Koper

COSTnet'17 conference
Palma de Mallorca, October 25-27, 2017

Outline

Generating large sparse random graphs and networks

For studying and predicting behaviour of Internet, communication, transportation, biological, social and other types of networks the simulation approach is usually used. As a support for it we need efficient generators of networks of the corresponding types [6, 12].

Large networks are usually (very) sparse – *Dunbar's number* [14]. Therefore the average degree in the network is not large. For this reason the standard (based directly on the definitions) algorithms for generating random graphs of selected type can be inefficient. In Batagelj and Brandes [5] we presented some fast algorithms for generating large sparse random graphs and networks of different types.

Generation of random graphs of Gilbert's type

For example, the generation of random graphs of Gilbert's type is equivalent to the filling of lower triangle of graph's adjacency matrix with Bernoulli sequence with parameter p of length $\binom{n}{2}$. In generating large and sparse such graphs we can replace it with puting the value 1 in positions determined by the corresponding geometrically distributed steps. This gives us much faster generator – see next slide.

Since in large sparse networks the probability p is very small in the R function `Gilbert` the parameter `p` is replaced by a more intuitive *average degree* `ad` and computed internally using the relation

$$ad = p \cdot (n-1)$$

obtained from

$$ad = \frac{1}{n}\sum_{v \in V} \deg(v) = \frac{2m}{n} \quad \text{and} \quad p = \frac{2m}{n(n-1)}$$

Generation of random graphs of Gilbert's type in R

```
Gilbert <-
# generates a random undirected graph of Gilbert's type
# on n nodes with expected average degree ad and writes
# it on the file fnet in Pajek's format.
# based on ALG.1 from: V. Batagelj, U. Brandes:
#   Efficient generation of large random networks
function(fnet,n,ad){
  net <- file(fnet,"w"); cat("*nodes",n,
    "\n% random Gilbert's graph / n = ",n," ad = ",ad,
    "\n*edges\n",file=net)
  logQ <- log(1-ad/(n-1));  v <- 1;  w <- -1
  repeat{
    w <- w + 1 + trunc(log(1-runif(1,0,1))/logQ)
    while (w >= v) {w <- w-v; v <- v+1}
    if (v >= n) break
    cat(v+1,w+1,'\n',file=net)
  }
  close(net)
}

Gilbert("gilbert.net",100,3.0)
```

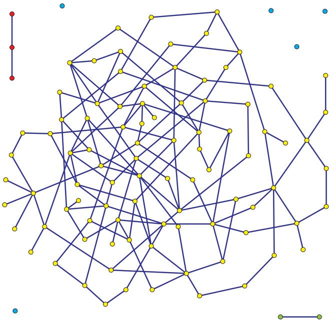
Generation of random scale-free networks

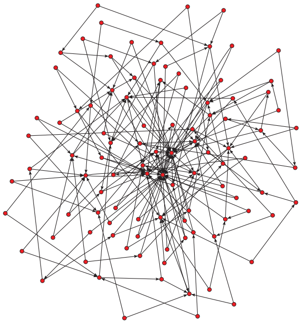
Similary, representing edges with pairs of nodes and observing that the number of copies of a node in a table equals its degree, we get the following fast generator of scale-free graphs. Examples of graphs generated with both algorithms are displayed on the following slide.

Generation of random scale-free networks

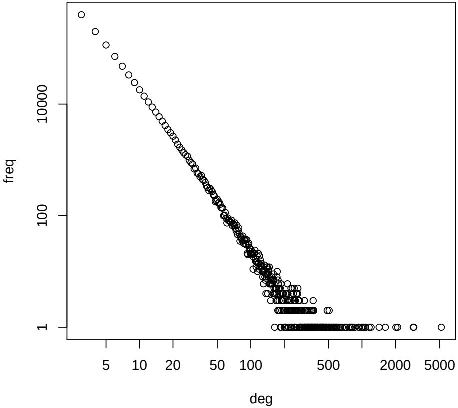
```
ScaleFreeBrstNet <-
# generates a random directed scale free graph
# on n nodes with d attachments to existing nodes
# and stores it on the file  fnet  in Pajek's format
# based on ALG.5 from: V. Batagelj, U. Brandes:
#   Efficient generation of large random networks
function(fnet,n,d){
  net <- file(fnet,"w"); cat("*vertices",n,"\n",file=net)
  k <- 0; m <- (n-d)*d; L <- rep(0,2*m); v <- d+1
  cat('% random scale free graph / n = ',n,' d = ',d,'\n',file=net)
  for(u in 1:d){k <- k+1; L[k] <- v; k <- k+1; L[k] <- u}
  for(v in (d+2):n) {
    repeat{S <- unique(L[sample(1:k,d)]); if(length(S)==d) break}
    for(i in 1:d) { k <- k+1; L[k] <- v;  k <- k+1; L[k] <- S[i]}
  }
  cat("*arcs\n",file=net)
  for (i in 1:m) cat(L[[2*i-1]],L[[2*i]],'\n',file=net)
  close(net)
}

ScaleFreeBrstNet("scaleFree.net",100,2)
```

Random graph of Gilbert's type, $n = 100$, $ad = 3$; and scale-free random graph, $n = 100$, $d = 2$





Degree distribution of scale-free random graph, $n = 1000000$, $d = 3$

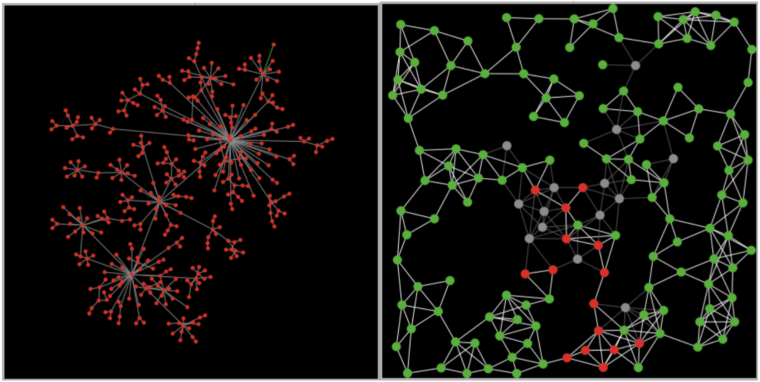


Probabilistic inductive classes of graphs

A random network can be viewed also as a result of an *evolution* process starting from some simple network in which next network is obtained from the current network using some (local) transformation. Some such models are implemented in programming language Netlogo [22] (see next slide).

The class of graphs/networks that can be obtained in this way can be described using *inductive definitions* (Curry [7], Batagelj [1]) or in more formalized setting using Lindenmayer's systems [20] or graph grammars (Ehrig et al. 1991 [9]). In our research we prefer the less formal inductive definitions because they are easier to adapt to specific characteristics of the classes of our interest.

Netlogo: evolution of scale-free network and spreading of the virus



Hilbert's space-filling curve

inductive definition [4]

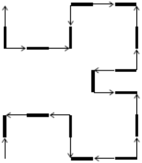
The nonterminals are represented by an arrow and the terminals by heavy lines which remain a permanent part of the objects. The axiom AH and the rules RH1 and RH2 should be understood as embedded in the plane.

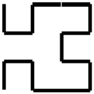
AH. 

RH1.  ::= 

RH2.  ::=

We start with AH  by RH1 we obtain  and by RH2 we obtain H_1  , but by RH1

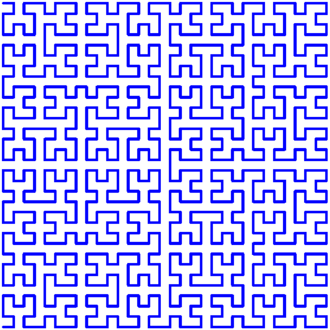


and by RH2 we obtain H_2 

etc.

Hilbert's space-filling curve

Logo program



```
TO Hilb :n :a :h
  IF :n = 0 [ STOP ]
  RT :a Hilb :n-1 (-:a) :h FD :h LT :a Hilb :n-1 :a :h FD :h
  Hilb :n-1 :a :h LT :a FD :h Hilb :n-1 (-:a) :h RT :a
END
TO Hilbert
  PU SETPOS [-150 -150] PD SETPC [000 000 255] Hilb 5 90 10
END
```

Inductive classes of graphs

In graph theory the inductive descriptions of many classes of graphs are known [3]. They are often used in proving theorems using the *inductive generalization*.

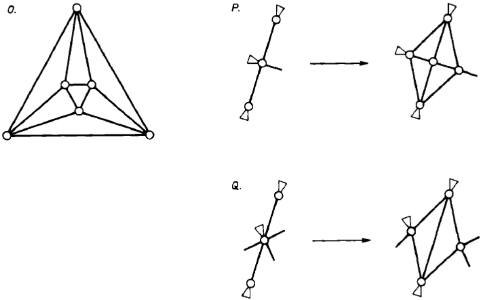
In order to show that every object from the inductive class has a certain property P it is sufficient to establish that

- every object from the basis has the property P ;
- the generating rules preserve the property P .

Example: even triangulations[2]

A graph is a *triangulation* (of the plane/sphere) iff it can be embedded in the sphere such that all its faces are triangles. A triangulation is *even* iff the degree of every node is even.

The inductive class $T_e = Cn(O; P, Q)$ is equal to the class of all even triangulations of the sphere.



Probabilistic inductive classes of graphs

The notion of inductive class of graphs can be extended by assigning probabilities to events in the evolution process. In the paper (Kejžar et al. [15]) we presented the following definition.

A *probabilistic inductive class of graphs* (*PICG*), $\mathcal{I}$, is given by:

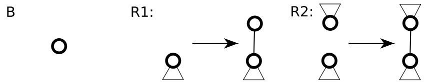
1. class $\mathcal{B}$ of initial graphs, the *basis* of PICG,
2. probability distribution specifying how the initial graph is chosen from class $\mathcal{B}$,
3. class $\mathcal{R}$ of generating rules, each with distinguished *left element* to which the rule is applied to replace it with the *right element*,
4. probability distribution specifying how the rules from class $\mathcal{R}$ are applied, and, finally,
5. a set of probability distributions specifying how the left elements for every rule in class $\mathcal{R}$ are chosen.

Probabilistic inductive classes of graphs

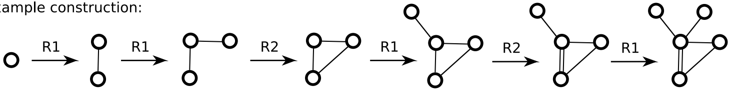
A random graph is obtained by starting from some randomly selected basic graph from the basis $\mathcal{B}$ and applying on it a randomly selected generating rules from $\mathcal{R}$ on randomly selected subgraph isomorphic to the rule's left element. On the so obtained graph the next randomly selected rule is applied, and so on. The PICG $\mathcal{I}$ consists exactly of graphs that can be obtained in this way in a finite number of steps. The sequence of graphs corresponding to these steps, enriched with the information about the applied rule, is called the *construction sequence* of a graph from the class.

On the next slide a simple ICG $\mathcal{I}(B; R1, R2)$ and an example of construction sequence are presented.

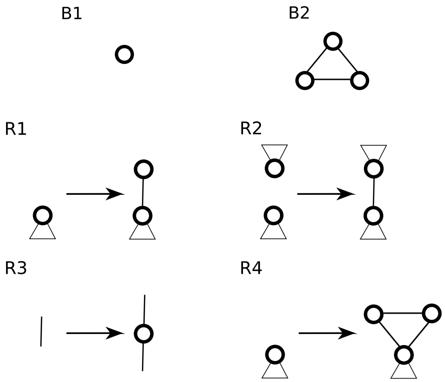
Basic graphs and rules



Example construction:



Probabilistic inductive classes of graphs – examples



Probabilistic inductive classes of graphs – examples

For the base graphs and rules from the previous slide the ICG $\mathcal{I}(B1; R1, R2)$ is the class of all connected (undirected) graphs, $\mathcal{I}(B2; R2, R3)$ is the class of all 2-node-connected graphs, and $\mathcal{I}(B2; R2, R3, R4)$ is the class of all 2-edge-connected graphs.

In our paper [15] we analyzed these three inductive definitions for the case when the generating rules have constant probabilities to be selected and the left part subgraph is selected with the uniform probability among available isomorphic subgraphs.

For such relatively simple definitions theoretical answers to some questions (degree distribution) can be obtained using the mean-field approach from theoretical physics. For more complicated definitions it seems that the only way to get some approximate answers is the simulation approach.

Probabilistic inductive classes of graphs comments

The number of classes that can be described as PICGs depends on the limitations we impose on the language for expressing the rules. In general we can allow also parametrized schemes of rules that produce rules only after specifying the values of parameters – they are finitely describing possibly infinite sets of rules. If needed we can also introduce a precedence among groups of rules – the rules with lower precedence are applied only when no rule with higher precedence can be applied.

In December 2017 Jason Vallet from Bordeaux will defend his PhD thesis in which he is proposing a detailed elaboration of rule selection mechanism [21].

Probabilistic inductive classes of graphs comments

In networks the graph structure is enriched by values in nodes and/or on links. Often they can be treated as colors. Taking a given network as a base network and introducing the rules that change colors, we can use PICGs also for studying different processes on networks – for example balancing in the signed networks [13], spreading of viruses, voting, etc.

An interesting question to be solved is also how to estimate the probabilities of generating rules from the realized graphs/networks. Recently [17] I learned about the Approximate Bayesian computation approach that could provide a solution.

References I

- Batagelj, V. (1985). Inductive classes of graphs. Proceedings of the Sixth Yugoslav Seminar on Graph Theory, Dubrovnik, 43-56. WWW
- Batagelj, V: An improved inductive definition of two restricted classes of triangulations of the plane. SKUPIEN, Zdzislaw, BOROWIECKI, Mieczyslaw (Eds.). Combinatorics and graph theory, (Banach Center Publications, vol. 25). Warszawa: Panstwowe wydawnictwo naukowe, 1989, 11-18 PDF
- Batagelj, V.: Inductive classes of graphs: Definitions and Bibliography. 1996 WWW
- Batagelj, V.: Drawing space-filling curves in logo. EuroLogo, 1997 PDF
- Batagelj, V. and Brandes, U. (2005) *Efficient Generation of Large Random Networks*. Physical Review E 71, 036113, 2005.
- BRITE: Boston University Representative Internet Topology Generator, WWW
- Curry, H.B. (1963) Foundations of mathematical logic. McGraw-Hill, New York.

References II

- Dorogovtsev, S.N., Mendes, J.F.F. (2003) *Evolution of Networks: From Biological Nets to the Internet and Www*. Oxford University Press.
- Ehrig, H., Kreowski, H. J., Rozenberg, G., (eds.) (1991). Graph Grammars and Their Application to Computer Science, Lecture Notes in Computer Science, vol. 532. Berlin and Heidelberg: Springer-Verlag.
- Erdős, P. and Rényi, A. (1959). "On Random Graphs. I.". Publicationes Mathematicae 6: 290-297.
- Gilbert, E.N. (1959) Random Graphs. Annals of Mathematical Statistics. Volume 30, Number 4 , 1141-1144.
- GT-ITM: Georgia Tech Internetwork Topology Models WWW
- Heider, F. (1946). Attitudes and cognitive organization. Journal of Psychology, 21, 107-112.
- Hill, R.A. and Dunbar, R.I.M. (2002) Social network size in humans, Human Nature, 14(1): 53-72.
- Kejžar, N., Nikoloski, Z., Batagelj, V.: Probabilistic Inductive Classes of Graphs. Journal of Mathematical Sociology 32: 85-109, 2008.

References III

- Kolaczyk, E.D. *Statistical Analysis of Network Data: Methods and Models*. Springer, Berlin 2009.
- Mira, A., Onnela, J.P., Peluso, S., Muliere, P., Caimo, A. : Inference in Statistical and Mechanistic Network Models. BoB, December 8, 2015 PDF
- de Nooy, W., Mrvar, A., Batagelj, V. (2005) *Exploratory Social Network Analysis with Pajek*. Cambridge University Press.
- Pennock, D.M., Flake, G.W., Lawrence, S., Glover, E.J., Giles, C.L. (2002) *Winners don't take all: Characterizing the competition for links on the web*. PNAS 99(8), 5207-5211.
- Prusinkiewicz, P., Lindenmayer, A.: The Algorithmic Beauty of Plants. Springer, 1991.
- Vallet, J.: Where Social Networks, Graph Rewriting and Visualisation Meet: Application to Network Generation and Information Diffusion. University of Bordeaux, thesis, 2017.
- Wilensky, U. (1999) Netlogo. Center for Connected Learning and Computer-Based Modeling. Northwestern University, Evanston, IL.

Acknowledgments

This work was supported in part by the Slovenian Research Agency (research program P1-0294 and research project J7-8279).

The authors attendance on the conference was supported by the COST Action CA 15109 – COSTnet.