Heterogeneous clustered random graphs

T. House

Warwick Mathematics Institute, University of Warwick - Coventry, CV4 7AL, UK

received 5 December 2013; accepted in final form 12 March 2014
published online 28 March 2014

PACS 89.75.Hc – Networks and genealogical trees
PACS 02.70.Uu – Applications of Monte Carlo methods
PACS 89.75.-k – Complex systems

Abstract – A graph Hamiltonian is proposed that allows creation of random networks close to specified connectivity, degree heterogeneity, and clustering.

Introduction. – Networks are now a ubiquitous tool in modelling complex systems that consist of multiple interacting components [1,2]. In the study of complex networks, random graph models form one of the key analytical tools, starting with the Erdős-Rényi model of network connectivity [3]. Study of real data on complex networks has consistently shown that these are both more heterogeneous and more clustered than the simplest random graph models would suggest [2].

Of these two, modelling heterogeneity has proved to be more straightforward with approaches such as the configuration model [4], preferential attachment [5], graph Hamiltonians [6] and combinatorially exact sampling [7] proving very successful. Currently quite a variety of random graph models with significant clustering have been proposed and successfully applied, including the Watts-Strogatz “small world” [8], triangular configuration model [9], and locally clustered approaches [10]. A fundamental property of all these models is, however, that they attribute zero likelihood to the majority of observed networks —for example, the model proposed by Newman [9] stipulates that no triangles may overlap, which is seldom true for real networks. This is not a problem if the aim is to obtain analytical results and insight, but in other contexts (mainly statistical inference) is a major obstacle.

Graph Hamiltonian. – Let \( \mathcal{G} \) be the set of all undirected graphs with \( N \) nodes. We consider a graph \( G \in \mathcal{G} \) with nodes indexed \( i, j, \ldots \), and links defined by the adjacency matrix \( G = (G_{ij}) \). Here, \( G_{ij} = G_{ji} = 1 \) if \( i \) and \( j \) are connected and \( G_{ij} = 0 \) otherwise; by convention we set \( G_{ii} = 0 \).

We write \( k_i \) for the degree of node \( i \), which also lets us define expectations of functions of degree:

\[
   k_i := \sum_j G_{ij}, \quad \langle f(k) \rangle := \frac{1}{N} \sum_i f(k_i).
\]

Analogously, we define the number of triangles node \( i \) participates in, \( t_i \), and related expectations, as

\[
   t_i := \sum_{j,k} G_{ij}G_{ik}G_{kj}, \quad \langle f(t) \rangle := \frac{1}{N} \sum_i f(t_i).
\]

Other relevant quantities, defined at the level of the whole network, are the number of links, number of lines and...
number of triangles, which are given, respectively, by

\[
M[G] := \sum_{i,j} G_{ij} = N\langle k \rangle,
\]

\[
L[G] := \sum_{i,j,k \neq j} G_{ij} G_{ik} = N\langle k(k - 1) \rangle,
\]

\[
T[G] := \sum_{i,j,k} G_{ij} G_{jk} G_{ki} = N\langle t \rangle.
\]

Note that this method of counting is more common for directed than undirected graphs, but will yield simpler expressions later. Our aim is to obtain a way to generate networks with specified values of these quantities.

We will define a probability measure over \( G \) in a Hamiltonian framework so that the probability given to \( G \in \mathcal{G} \) is

\[
\pi(G|\theta) = \frac{e^{-H(G;\theta)}}{Z_\theta}, \quad \text{where} \quad Z_\theta = \sum_{G \in \mathcal{G}} e^{-H(G;\theta)}.
\]

Here \( H \) is called the Hamiltonian, \( \theta = (\theta_a) \) is a vector of model parameters, and \( Z \) is the partition function. We will also write \( \tilde{\pi} \) for the ensemble probability of a network having properties \( F[G] = (F_a[G]) \),

\[
\tilde{\pi}(f|\theta) = \sum_{G \in \mathcal{G}} \pi(G|\theta) \prod_a \delta_{f_a,F_a[G]}.
\]

Standard exponential random graph models invoke maximum entropy at given mean to motivate a linear Hamiltonian \([2,12,13]\), taking the form

\[
H(G;\theta) = \sum \theta_a F_a[G],
\]

where the parameter vector \( \theta \) is chosen to give the desired mean values to the network properties \( F_a \). For the simplest of these, the Bernoulli random graph \((H = \theta M)\) we can impose mean link density \( \rho \) and use the independence of links in the model to solve analytically, giving

\[
\pi(G|\theta) = \frac{e^{-\mathcal{M}(G)\theta}}{(1+e^{-\theta})^{N(N-1)}},
\]

\[
\tilde{\pi}(M|\theta) = (N(N-1))/(1 - \rho)^{N(N-1)-M} \rho^M,
\]

where \( \theta = \ln((1 - \rho)/\rho) \). The difference between these distributions arises due to combinatorial factors (i.e. the binomial coefficient) and is shown graphically in fig. 1.

One known pathology of linear Hamiltonians is that models based on them often exhibit degeneracies \([2,14]\), leading Park and Yook \([20]\) to propose a non-linear optimisation Hamiltonian for degree and clustering at an individual level:

\[
H(G;\beta_d,\beta_c,\Phi) = \sum_i \left( \beta_d k_i - g_i \right) + \beta_c \left( t_i - \frac{1}{2} \gamma_i k_i(k_i - 1) \right).
\]

This approach avoids the degeneracy problems of linear Hamiltonians. Other approaches make use of a non-linear Hamiltonian for the overall number of triangles in the network \([15,17]\),

\[
H(G;\beta,\Theta) = \beta \langle T[G] - \Theta \rangle,
\]

where \( \Theta \) is the desired number of triangles in the network. Such a model is designed to work with a sampling strategy based on applying edge swaps to a network that already has the desired degree distribution.

Both of these approaches require a large number of parameters to work: \(8\) has \(2(N+1)\) explicit model parameters and \(8\) has \(2\) explicit parameters, but with the \(N\) node degrees constrained or known. Our aim here is to derive a graph Hamiltonian that will appropriately constrain network connectivity, heterogeneity and transitivity, with \(O(1)\) rather than \(O(N)\) parameters.

To do this, let us first consider what might be a natural functional form, since there is no a priori reason why the absolute value should be used. If we want to adopt an optimising approach, then in place of \(6\) we would write

\[
H(G;\Phi) = \sum_a \Phi_a(F_a[G]),
\]
where the constraint on $\Phi_a$ is that it should take its minimum at a specified value, and hence we constrain the \textit{mode} rather than the \textit{mean} of the network statistic $F_a[G]$ under the probability measure. If $\Phi_a$ are quadratic functions, then we are aiming for a sampling regime such that in the thermodynamic limit the marginal distributions for each network property $F_a[G]$ near the mode are normal (\textit{i.e.} the maximum entropy distribution of given mean and variance) making this a natural choice.

Now let us consider a relatively small set of target network statistics that capture heterogeneity and clustering, as shown in fig. 2. First, let $\theta_m \in [0, 1]$ be the target link density -- \textit{i.e.} the desired proportion of possible links that are actually present. Secondly, let $\theta_l \in [0, \infty)$ be the target value of $(\langle k^2 \rangle - \langle k \rangle^2) / \langle k \rangle$. Finally, let $\theta_t \in [0, 1]$
be the desired network clustering coefficient [2]. Putting these together with the quadratic form for (10) above gives the following six-parameter Hamiltonian:

\[
H(G; \theta, \beta) = \beta_m (M[G] - \theta_m N(N-1))^2 \\
+ \beta_l (L[G] - M[G]((\theta_l - 1) + N^{-1}M[G]))^2 \\
+ \beta_t (T[G] - \theta_t L[G])^2
\]  

(11)

It then remains to motivate values for the parameters \( \beta \). In general, higher values of these parameters will attribute lower probabilities to networks further from the mode. From the form of (11) we have natural choices,

\[
\beta_m = 1/(2\sigma^2), \\
\beta_l = (\theta_l + \theta_m(N-1) - 1)\beta_m, \\
\beta_t = \theta_t\beta_l.
\]  

(12)

There is then the question of what value of the standard deviation in the number of links, \( \sigma \), is most appropriate.

**Simulation.** – In general, the sum over the set of all graphs, \( \mathcal{G} \), needed to calculate the partition function in (4) involves \( 2^{N(N-1)/2} \) possibilities making this impractical. Monte Carlo methods—in particular the Metropolis-Hastings algorithm [21,22]—can be used to overcome this difficulty. Here a current graph \( G \) is stored in memory, and a new graph \( G' \) proposed with probability \( e^{q(G';G)} \). Then the new graph is accepted if

\[
\ln(\text{rand}) < q(G; G') - q(G'; G) - \Delta H(G', G).
\]  

(13)

Here, \( \text{rand} \) is a random number picked from the uniform distribution \( U(0, 1) \), and \( \Delta H \) is the proposed change in the Hamiltonian. For most moves in graph space (e.g. adding/deleting links and edge swaps) the probability of
a proposal is equal to the probability of proposing its reversal and so the $q$ terms cancel. The stationary distribution of the Markov chain defined by (13) will be (4) provided the process is ergodic. As we discussed above for the Bernoulli random graph, this is no guarantee that the measure $\pi$ over network properties will be as desired. Suppose we could calculate the probability $q(F'; F)$ that a graph with properties $F'$ will be proposed from one with $F$, and consider the Markov chain defined by the acceptance criterion

$$\ln(\text{rand}) < Q - \Delta H(G', G), \quad \text{where}$$

$$Q := \tilde{q}(F|G); F(G') - \tilde{q}(F'|G') \tilde{q}(F|G).$$

(14)

The stationary distribution of this chain will not have stationary distribution $\pi$, but will have the distribution $\pi$ over network properties that we have designed our Hamiltonian to produce. Exact calculation of $Q$ will in general involve extremely complex combinatorial considerations, making it computationally unfeasible. It is possible, however, to make a judicious choice of $\beta$ in (11) so that $\Delta H \gg Q$, meaning that we can approximate

$$Q \approx 0.$$  

(15)

Where this approximation is good, the criteria for acceptance (13) and (14) are equivalent, and we expect the network measure defined by (11) to yield distributions over network properties that are approximately normal as desired. The approximation (15) will be more accurate for larger values of the precision parameters $\beta$, since as these parameters are increased the (tractable) differences in the Hamiltonian will become larger while the (intractable) combinatorial factors remain constant. On the other hand, overly large differences will worsen the computational performance of the Metropolis-Hastings scheme [23]. This means that there is a trade-off between precision and the ability to sample networks efficiently. If we were to implement a Monte Carlo scheme of choosing distinct pairs $(i, j \neq i)$ and then proposing $G_{ij} \rightarrow 1-G_{ij}, G_{ji} \rightarrow 1-G_{ji}$, then each proposal involves a change in $M$ of 2. Therefore, a natural scale for the standard deviation associated with $M$ is $\sigma = O(1)$, and the precise choice $\beta_m = 1/4$, with other parameter values following from (12), is used here since these were found to perform well.

Figure 3 shows $4 \times 10^3$ samples from the model for three different parameter choices, which were generated in 90 seconds on a modern laptop and have properties close to those desired with errors arising due to the approximation (15). In general, convergence of the algorithm is smooth and it does not become trapped in local minima. Figure 4 shows that as would be expected the mixing times starting with the empty graph are between the order of pairs of nodes, $O(N^2)$, and the next polynomial order $O(N^3)$. The question of whether better performance could be obtained using advanced Monte Carlo methods [23] remains open.

**Power grid randomisation.** To apply the approach suggested to a real-world dataset, parameters were taken from the power grid network considered by Watts and Strogatz [8]: $N = 4941$, $\theta_m = 5.4 \times 10^{-4}$, $\theta_t = 1.2$, $\theta_i = 0.10$. One hundred simulations were run starting from the graph $G_{i,j} = 0$, $\forall i, j$ until convergence. The results of these are shown in fig. 5, showing that the approach is slightly biased due to the approximation (15), but for practical purposes delivers random samples of networks with the desired statistics to high accuracy.

Note that for the original power grid network, the mean shortest path length is 19 and the network consists of one giant component. For the simulated networks, the mean finite shortest path length is 7.5, but on average 26% of shortest paths are infinite (i.e., the nodes are in different components). This indicates how the real network is structured to preserve connectivity at the expense of path length when compared to a random graph, even when adjusting for connectivity, heterogeneity and clustering.
To emphasise the usefulness of the random graph model independently of the simulation method used, the initial network was also randomised using edge swaps, and then edge swaps were used to reintroduce clustering using the Hamiltonian (11), with results shown in fig. 6, which are very similar to those of fig. 5, but with degree distribution unchanged compared to the original network.

Conclusions. – The approach presented here gives a general recipe for complex random graph models. First, decide which network properties are to be specified and constrained for the scientific problem in hand. Secondly, determine what conditional relationships hold between these properties and add any that are missing: in the model considered above, both triangles and lines are included since triangles are closed lines. Thirdly, write down an appropriate optimising Hamiltonian. Finally, tune the precision parameters to obtain the appropriate balance between accuracy and sampling efficiency.

***

Work funded by the Engineering and Physical Sciences Research Council. I would like to thank two anonymous referees for constructive comments that have improved this manuscript.

REFERENCES

[1] Boccaletti S., Latora V., Moreno Y., Chavez M. and Hwang D.-U., Phys. Rep., 424 (2006) 175.
[2] Newman M. E. J., Networks: An Introduction (Oxford University Press) 2010.
[3] Erdős P. and Rényi A., Publ. Math. Inst. Hung. Acad. Sci., 5 (1960) 17.
[4] Molloy M. and Reed B., Random Struct. Algorithms, 6 (1995) 161.
[5] Barabási A. L. and Albert R., Science, 286 (1999) 509.
[6] Yook S.-H. and Park J., EPL, 93 (2011) 38001.
[7] del Genio C. I., Kim H., Toroczkai Z. and Bassler K. E., PLoS ONE, 5 (2010) e10012.
[8] Watts D. J. and Strogatz S. H., Nature, 393 (1998) 440.
[9] Newman M. E. J., Phys. Rev. Lett., 103 (2009) 1.
[10] Bollobás B., Janson S. and Riordan O., Random Struct. Algorithms, 38 (2011) 269.
[11] Strauss D., SIAM Rev., 28 (1986) 513.
[12] Snijders T. A. B., Pattison P. E., Robins G. L. and Handcock M. S., Sociol. Methodol., 36 (2006) 99.
[13] Robins G., Pattison P., Kalish Y. and Lusher D., Soc. Netw., 29 (2007) 173.
[14] Park J. and Newman M. E. J., Phys. Rev. E, 72 (2005) 026136.
[15] Milo R., Shen-Orr S., Itzkovitz S., Kashtan N., Ciklovskii D. and Alon U., Science, 298 (2002) 824.
[16] Bansal S., Khandelwal S. and Meyers L. A., BMC Bioinform., 10 (2009) 405.
[17] Foster D., Foster J., Paczuski M. and Grassberger P., Phys. Rev. E, 81 (2010) 046115.
[18] House T. and Keeling M. J., PLoS Comput. Biol., 6 (2010) e1000721.
[19] Coolen A. C. C., Martino A. and Annibale A., J. Stat. Phys., 136 (2009) 1035.
[20] Park J. and Yook S.-H., J. Stat. Mech.: Theory Exp. (2011) P08008.
[21] Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A. H. and Teller E., J. Chem. Phys., 21 (1953) 1087.
[22] Hastings W. K., Biometrika, 57 (1970) 97.
[23] Brooks S., Gelman A., Jones G. L. and Meng X.-L. (Editors), Handbook of Markov Chain Monte Carlo (CRC Press) 2011.