Generating conditional uniform random networks by optimization procedures

Stefano Nasini
stefano.nasini@upc.edu

Jordi Castro
jordi.castro@upc.edu

Dept. of Statistics and Operations Research
Universitat Politècnica de Catalunya
Jordi Girona 1–3
08034 Barcelona

Abstract

Complex networks is a recent area of research motivated by the empirical study of real-world networks, such as social relations, protein interaction, neuronal connections, etc. As closed-form probabilistic models of networks are often not available, the ability of randomly generating networks verifying specific constraints might be useful. The purpose of this work is to develop optimization-based procedures to randomly generate networks with structural constraints, within the probabilistic framework of conditional uniform models. Based on the characterization of families of networks by means of systems of linear constraints, polynomial-time methods to generate networks with specified structural properties are constructed.

Key words: Conditional Random Networks, Linear Programming, Interior Point Methods, Block-angular problems, Social Networks.

1 Extended Abstract

After observing a real-world network, a conditional uniform model [1] might be used to check whether empirical features (for example, the number of closed triangles, or the number of connected paths) are significantly unlikely under a uniform distribution of all networks having fixed and well known structural properties (for example, uniform probability distribution of all networks with fixed density). Comparing the observed network with an ensemble of such networks allows one to detect deviations from randomness in network properties.

Let x_{ij} be the (i,j) component of the adjacency matrix (AM, from now on) of a simple graph. Then, the set G_n of all simple graphs with n nodes is characterized by the integer points of the polytope $\varphi = \{(x_{12},\ldots,x_{n-1n}) \in [0,1]^{n(n-1)} : x_{ij} - x_{ji} = 0, i = 1\ldots n-1,\ i < j \leq n\ \}$, i.e., $\varphi \cap \mathbb{Z}^{n(n-1)}$. Adding nonnegative slacks, the representation of φ in standard matrix form is

$$\left[\begin{array}{c|c|c|c} I & -I & & \\ \hline I & & I & \\ \hline & I & & I \end{array}\right] \left[\begin{array}{c} \mathbf{x} \\ \hline \mathbf{s} \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \mathbf{e} \\ \hline \mathbf{e} \end{array}\right] \qquad \left[\begin{array}{c} \mathbf{x} \\ \hline \mathbf{s} \end{array}\right] \geq \mathbf{0}. \tag{1}$$

Matrix I is the $n(n-1)/2 \times n(n-1)/2$ identity matrix, $\mathbf{e} = (1,\ldots,1)^T \in \mathbb{R}^{n(n-1)/2}$, $\mathbf{x}^T = [x_{12},\ x_{13},\ \ldots,\ x_{1n},\ x_{23},\ x_{24},\ \ldots,\ x_{2n},\ \ldots,\ x_{45},\ \ldots,\ x_{(n-1)n},\ x_{21},\ x_{31},\ x_{32},\ x_{41},\ \ldots,\ x_{n(n-1)}]$ and $\mathbf{s}^T = [s_{12},\ s_{13},\ \ldots,\ s_{1n},\ s_{23},\ s_{24},\ \ldots,\ s_{2n},\ \ldots,\ s_{45},\ \ldots,\ s_{(n-1)n},\ s_{21},\ s_{31},\ s_{32},\ s_{41},\ \ldots,\ s_{n(n-1)}]$.

The coefficient matrix of (1) is totally unimodular (TU, from now on), such that all extreme points of φ are integer. Moreover, since φ is a subset of the $n(n-1)$-dimensional unit cube, there couldn't be any integer point in the interior, so that each graph with n nodes, whose AM is given by an integer solution of (1), must be associated to a unique extreme point of φ and, conversely, the latter is by construction associated to a unique AM. Sufficient conditions for a matrix to be totally unimodular are shown in the appendix of [4].

We proved that, due to the total unimodularity of their matrix structure, many interesting families of networks could be characterized by extreme points of polytopes:

- undirected networks conditioned to the total density;
- directed networks conditioned to the total density;
- undirected networks conditioned to the within & between group densities;
- undirected networks conditioned to the within group densities;
- undirected networks conditioned to the between group densities;
- directed networks conditioned to the within & between group densities;
- directed networks conditioned to the within group densities;
- directed networks conditioned to the between group densities;
- undirected networks conditioned to the within group densities and the total density;
- undirected networks conditioned to between group densities and the total density;
- directed networks conditioned to the within group densities and the total density;
- directed networks conditioned to the between group densities and the total density;
- directed networks conditioned to the in-&-out-degree sequences;
- undirected networks conditioned to the lower bound of the degree sequence range and density;
- directed networks conditioned to the diad count;
- directed networks conditioned to the number of mutual links;

The fact that we are able to associate to each network verifying specified constraints an extreme point of a polytope, allows to generate random networks by solving linear programs.

It may be seen that the systems of linear constrains associated to the families of networks listed above have a primal block-angular structure, as the following

$$\min \quad \sum_{i=0}^{k} {c^i}^T x^i \tag{2a}$$

$$\text{subject to} \quad \begin{bmatrix} N_1 & & & & \\ & N_2 & & & \\ & & \ddots & & \\ & & & N_k & \\ L_1 & L_2 & \dots & L_k & I \end{bmatrix} \begin{bmatrix} x^1 \\ x^2 \\ \vdots \\ x^k \\ x^0 \end{bmatrix} = \begin{bmatrix} b^1 \\ b^2 \\ \vdots \\ b^k \\ b^0 \end{bmatrix} \tag{2b}$$

$$\mathbf{0} \leq x^i \leq \mathbf{1} \quad i = 0, \dots, k. \tag{2c}$$

Matrices $N_i \in \mathbb{R}^{m_i \times n_i}$ and $L_i \in \mathbb{R}^{l \times n_i}$, $i = 1, \dots, k$, respectively define the block-diagonal and linking constraints, k being the number of blocks. Vectors $x^i \in \mathbb{R}^{n_i}, i = 1, \dots, k$, are the variables for each block. $x^0 \in \mathbb{R}^l$ are the slacks of the linking constraints. $b^i \in \mathbb{R}^{m_i}, i = 1, \dots, k$, is the right-hand-side vector for each block of constraints, whereas $b^0 \in \mathbb{R}^l$ is for the linking constraints. The upper bounds for each group of variables are defined by $u^i, i = 0, \dots, k$.

Specialized versions of the simplex method, interior point methods and decomposition techniques have been proposed for these particular structures [2, 3].

Let $A \in \mathbb{R}^{m' \times n'}$ be the coefficient matrix associated to one of the polytopes whose extreme points are bijectively related to the aforementioned families of networks. By the Fundamental Theorem of Linear Programming, given a $\mathbf{c} \in \mathbb{R}^{|V|(|V|-1)}$, there exists a network g belonging to the specified family of networks such that $g \in \text{argmax}_{\mathbf{x}}\, \mathbf{c}^T\mathbf{x}$, s.t. $\mathbf{x} \in \varphi$. This means that, if $\mathbf{c}$ is a properly defined random vector, the probability distribution of a network in the specified family can be obtained by associating to a given $\mathbf{c} \in \mathbb{R}^{|V|(|V|-1)}$ the set $\{\mathbf{x} \in \varphi : \mathbf{x} \in \text{argmax}_{\mathbf{x}}\, \mathbf{c}^T\mathbf{x}\}$.

Clearly, if each network is an optimal solution of a LP with randomly generated objective function, we cannot state that every network has the same probability of being selected, i.e. we are not authorized to regard the sample of networks generated this way as draw from a uniform distribution. After fixing a random objective function the probability of obtaining a network depends on geometrical properties of the associated polytope (the polytope whose extreme points characterize the family). It seems in any case reasonable to assume that such probability does not depend on any structural property of the network, except for the ones defined by the specified constraints.

If we have a given extreme point $\mathbf{x}_0$ of φ, we might obtain another extreme point $\mathbf{x}_1$ of φ by fixing r variables and optimizing, with a given objective cost vector $\mathbf{c} \in \mathbb{R}^{N-r}$, the remaining $N - r$ variables.

To clarify this method, consider a primal block-angular problem with R blocks and let $\mathfrak{r} \subset \{1, 2, \ldots, R\}$. Let $\mathbf{x}_{(\mathfrak{r})}$ be the vector of decision variables associated with blocks indexed by $\mathfrak{r}$ and $\mathbf{x}_{(\mathfrak{r}^c)}$ be the vector of decision variables associated with the $|\mathfrak{r}^c| = R - |\mathfrak{r}|$ remaining blocks. Similarly, let $A_{(\mathfrak{r})}$ represent the submatrix of A, whose columns corresponds to the blocks indexed by $\mathfrak{r}$ and $A_{(\mathfrak{r}^c)}$ the submatrix of A, whose columns corresponds to the blocks indexed by $\mathfrak{r}^c$.

Algorithm 1 illustrates the iterative procedure to generate a sequence of networks by randomly selecting $\mathbf{c} \in \mathbb{R}^{N-r}$ and the blocks to be optimized $\mathfrak{r}$.

Algorithm 1 R-blocks chain

1: Let $k = 0$ $\mathbf{x}^k$ be an initial extreme point
2: **repeat**
3: Randomly select $\mathbf{c} \in \mathbb{R}^{N-r}$ and $\mathfrak{r} \subset \{1, 2, \ldots, R\}$;
4: Construct $A_{(\mathfrak{r}^c)}$, $A_{(\mathfrak{r})}$, $\mathbf{x}_{(\mathfrak{r}^c)}$ and $\mathbf{x}_{(\mathfrak{r})}$;
5: Solve the associated subproblem;
6: Update $\mathbf{x}_{(\mathfrak{r})}^{k+1}$ using the optimal solution;
7: $k = k + 1$;
8: **until** $k \geq K$

Another way to generate extreme points of a polytope is to produce a sequence of nondegenerate bases. Consider a partition of the set of columns of A into two groups: the basic columns $\mathcal{B}$ (linearly independent, $|\mathcal{B}| = rank(A)$), and the nonbasic ones $\mathcal{N}$. The components of $\mathbf{x}$ associated to the non-basic columns ($\mathbf{x}_{\mathcal{N}}$) are fixed at their limits, namely either 0 or 1, and the variables associated to basic columns ($\mathbf{x}_{\mathcal{B}}$) are computed as $\mathbf{x}_{\mathcal{B}} = B^{-1}(\mathbf{b} - N\mathbf{x}_{\mathcal{N}})$, where B and N are the matrices of columns of A associated to $\mathcal{B}$ and $\mathcal{N}$. Denoting as e_q the q-th column of the identity matrix of dimension $|\mathcal{N}|$, $q = 1, \ldots, |\mathcal{N}|$, if $|\varphi \cap \mathbb{Z}^{n(n-1)}| > 1$ (i.e, there is more than one extreme point in φ), from an extreme point $x_k \in \varphi$ we can compute another extreme point as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda \begin{bmatrix} B^{-1}Ne_q \\ -e_q \end{bmatrix}. \tag{3}$$

Indeed (3) is equal to the standard simplex iteration. Using (3) we can obtain a finite sequence

of K extreme points $\{\mathbf{x}_k\}_{k=0}^{K}$. (From now on we shall use $\delta_{\mathcal{B}}(q)$ to denote in a more compact notation the direction of movement of (3).)

It can be shown for the polytopes described in this paper (subsets of a finite-dimensional hypercube with integer extreme points) that the step length $\lambda \in \{-1, 0, 1\}$.

The sequence (3) requires the repeated selection of $q \in \{1, \ldots, |\mathcal{N}|\}$ and $\lambda \in \{-1, 0, 1\}$. Thus, if we let q be a random variable, (3) gives rise to a Markov Chain over the extreme points of φ. Algorithm 2 provides an iterative procedure to simulate using that Markov chain.

Algorithm 2 Pivoting sequence

1: Let $\mathbf{x}_0$ be an initial extreme point
2: **repeat**
3: Randomly select $q \in \{1, \ldots, |\mathcal{N}|\}$;
4: compute the feasible step-length $\lambda \in \{-1, 0, 1\}$ associated with q;
5: **if** $\lambda \neq 0$ **then**
6: Update $\mathcal{B}^{-1}$ and $\mathcal{N}$ and compute $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda\delta_{\mathcal{B}}(q)$; $k = k + 1$;
7: **end if**
8: **until** $k \geq K$

The two procedures to generate extreme points have been implemented, for most of the aforementioned families of networks. The comparison of our computational results with the few available analytical solutions confirms that both procedures provide coherent numerical results, matching the known theoretical values. When analytical results are not available, the two procedures give rise to the same results for large sample sizes and differ only in terms of efficiency, since many bases obtained by the pivoting method do not produce new networks due to the high degeneracy of the problems. Finally, the method based on the sequential random selection of K blocks is (1 efficient, and (2 allows to obtain unbiased information of real-world network properties.

References

[1] Bollobas, B., (1985), *Random Graphs*. Cambridge University Press.

[2] Castro, J., (2000), A specialized interior-point algorithm for multicommodity network flows, *SIAM J. Optim.*, 10, 852-877.

[3] Castro, J., (2003), Solving difficult multicommodity problems through a specialized interior-point algorithm, *Ann. Oper. Res.*, 124, 35-48.

[4] Heller, I., Tompkins, C.B., (1956), 'An Extension of a Theorem of Dantzig's', in K. Kuhn, A. Tucker, *Linear Inequalities and Related Systems*, Princeton University Press, Princeton, 247-254.