

Mathematical programming approaches for classes
of random network problems

Jordi Castro and Stefano Nasini
Dept. of Statistics and Operations Research
Universitat Politècnica de Catalunya
Jordi Girona 1-3, 08034 Barcelona, Catalonia
`jordi.castro@upc.edu` `stefano.nasini@upc.edu`
Report DR 2013-01

July 2013; updated March 2014, November 2014, January 2015, March 2015

Report available from <http://www-eio.upc.es/~jcastro>

Mathematical programming approaches for classes of random network problems

Jordi Castro*, Stefano Nasini

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

Abstract

Random simulations from complicated combinatorial sets are often needed in many classes of stochastic problems. This is particularly true in the analysis of complex networks, where researchers are usually interested in assessing whether an observed network feature is expected to be found within families of networks under some hypothesis (named conditional random networks, i.e., networks satisfying some linear constraints). This work presents procedures to generate networks with specified structural properties which rely on the solution of classes of integer optimization problems. We show that, for many of them, the constraints matrices are totally unimodular, allowing the efficient generation of conditional random networks by specialized interior-point methods. The computational results suggest that the proposed methods can represent a general framework for the efficient generation of random networks even beyond the models analyzed in this paper. This work also opens the possibility for other applications of mathematical programming in the analysis of complex networks.

Key words: Linear Programming, Interior-Point Methods, Integer Programming, Complex Networks, Social networks analysis, Total unimodularity, Central path

1. Introduction

The use of random simulation is quite common when statistically studying properties of highly combinatorial sets. In many of those cases, closed-form expressions are hard to be found and the availability of *efficient* and *correct* simulation procedures might be of remarkable importance.

This is particularly true in the analysis of complex networks, an interdisciplinary field which brings together tools and methods from discrete mathematics and computer science with a great concern toward empirical applications, among others, in business, marketing, epidemiology, engineering, etc. Researchers are often interested in assessing the hypothesis of whether a particular network property is likely to appear under a uniform distribution of all networks verifying given constraints, named *conditional random networks* [4]. In the absence of closed-form expressions (as it is often the case for most of random network models), large random samples of networks satisfying particular properties are required to test these hypotheses. This work introduces novel procedures to generate this sample, based on linear and integer optimization. They result in a general approach for random network simulation, which outperforms in versatility some currently available methods.

*Corresponding author

Email addresses: `jordi.castro@upc.edu` (Jordi Castro), `stefano.nasini@upc.edu` (Stefano Nasini)

Following the standard notation [2], a graph $G = (V, E)$ is defined by a finite set V of n nodes, and a set of m pairs of them $E \subseteq V \times V$, named edges or arcs. A graph can be represented by a $n \times n$ binary matrix X , called *adjacency matrix* (AM from now on), whose (i, j) -entry, x_{ij} , is equal to 1 if there is a link between nodes i and j , and 0 otherwise. We will assume the graph has no loops, so that the diagonal of X is null. A network is a graph whose arcs or nodes have associated numerical values (arc costs, arc capacities, node supplies, etc). In this work we will make no distinction and the two terms “graph” and “network” will be used as synonyms.

The study of random graphs begins with the seminal work of P. Erdős and A. Rainyi [11], who considered a fixed set of nodes and an independent and equal probability of observing edges among them. There are two closely related variants of the Erdős-Rainyi model:

- *the $G(n, p)$ model*, where a network is constructed by connecting nodes randomly with independent probability p ;
- *the $G(n, m)$ model*, where a network is chosen uniformly at random from the collection of all graphs with n nodes and m edges.

Both models possess the considerable advantage of being exactly solvable for many of their average properties: clustering coefficient, average path length, giant component, etc. (For more details about network properties, see Bollobas [4], and Wasserman and Faust [34].) In other words, the expectation of many structural properties of networks generated by the Erdős-Rainyi processes is analytically obtainable. Conditional uniform models can be seen as a generalization of the $G(n, m)$ model, when the conditioning information is not necessarily the number of edges but whatever other arbitrary network property. Unfortunately, in this case we have very few analytical results and simulation is required to obtain empirical distributions of their average properties.

Although other Operations Research tools have been used in the context of social networks [3, 13], as far as we know, this work is the first attempt to use linear and integer optimization for the generation of several classes of conditional random graphs. Previous approaches, developed within the fields of mathematical and computational sociology, were ad hoc procedures for some particular types of networks, in general difficult to generalize and not very efficient. For instance, the distribution of all networks conditioned to the nodes in- and out-degree have difficult combinatorial properties, as its analytical study involves binary matrices with fixed marginal rows and columns. In this respect, some combinatorial results have been obtained by Ryser [29], who derived necessary and sufficient conditions for two vectors of non-negative integers to constitute the row sums and column sums of some zero-one matrix. On the other hand, ways to generate uniform random networks with given degree distribution were developed in [31, 26, 8, 28, 33], although they were computationally expensive and prohibitive for very large AMs.

In practice one would like to go even further in conditioning, which however leads to self-defeating attempts because of combinatorial complexity. This work provides a general methodological framework to generate networks with constraints, representing structural features we wish to control for.

Let x_{ij} be entries of the AM of either a directed or undirected graph with no loops or multiples edges. The AM is an element of the set of binary matrices

$$\chi = \{x_{ij} \in \{0, 1\}, (i, j) \in H\}, \tag{1}$$

$$\begin{array}{ll} \text{where} & H = \{(i, j) : 1 \leq i \leq n - 1, i < j \leq n\} & \text{for undirected graphs} \\ \text{or} & H = \{(i, j) : 1 \leq i \leq n, 1 \leq j \leq n, i \neq j\} & \text{for directed graphs.} \end{array}$$

The continuous relaxation of χ , name it $CR(\chi)$, is obtained by replacing $x_{ij} \in \{0, 1\}$ by $x_{ij} \in [0, 1]$, in (1). Clearly, all extreme points of $CR(\chi)$ are integer. If we consider a conditional graph by adding extra linear constraints to χ , then $CR(\chi)$ may contain fractional extreme points, unless its constraints matrix is totally unimodular (TU, from now on). As shown in [15], the next theorem provides sufficient conditions for a matrix to be TU:

Theorem 1. *Let $A \in \{-1, 0, 1\}^{m \times n}$ be a matrix obtained by elementary operations of $B \in \mathbf{Z}^{m \times n}$ and consider a partition of the rows of A in two disjoint sets \mathcal{J}_1 and \mathcal{J}_2 . The following three conditions together are sufficient for B to be TU:*

1. *Every column of A contains at most two non-zero entries, which are either 1 or -1 .*
2. *If two non-zero entries in a column of A have the same sign, then the row of one is in \mathcal{J}_1 , and the other in \mathcal{J}_2 .*
3. *If two non-zero entries in a column of A have opposite signs, then the rows of both are either in \mathcal{J}_1 or \mathcal{J}_2 .*

The above theorem will be extensively used in next section. More details on unimodularity in integer programming can be found in [30]. If the constraints matrix of $CR(\chi)$ is TU, each extreme point of $CR(\chi)$ represents a graph. Therefore, it is possible to generate a bunch of graphs by merely solving linear programs (LP) with random gradients in the objective function, or by non-degenerated simplex pivoting, starting from a given initial extreme point [24]. Moreover, they can be generated in polynomial time if interior-point methods are used [35].

The paper is organized as follows. Section 2 is devoted to the characterization of the convex hull of polytopes associated to some families of conditional random networks. We will differentiate between families whose constraints are TU, and those which may give rise to fractional AMs. Supported by these results, Section 3 presents two particular procedures for the generation of conditional random networks, and analyzes the probability distribution of the LP solutions. Section 4 illustrates these techniques using some real-world data sets.

Throughout the paper we denote the vector of variables associated to the components of the AM as either $\mathbf{x}^T = [x_{12}, \dots, x_{1n}, x_{23}, \dots, x_{(n-1)n}, x_{21}, \dots, x_{n(n-1)}]$ (i.e., the rowwise upper triangle of AM followed by its columnwise lower triangle) for directed graphs, or $\mathbf{x}^T = [x_{12}, \dots, x_{1n}, x_{23}, \dots, x_{(n-1)n}]$ (only the rowwise upper triangle of AM) for undirected graphs.

2. Total unimodularity of constraints from some conditional random networks

Let χ be the set of AMs of a family of either directed or undirected networks with n nodes, and let $CR(\chi)$ be its continuous relaxation. For about twenty families of networks the extreme points of $CR(\chi)$ can be seen to be integer. Although making an extensive list of all these families is out of the scope of this work, some of the most relevant ones will be discussed in the following subsections.

Next Proposition 1, which provides a sufficient condition for the existence of a bijection between extreme points of $CR(\chi)$ and the set of feasible networks, will be useful to show that some constraints matrices are TU.

Proposition 1. *For a given family of either directed or undirected networks with n nodes, let $F \in \mathbb{R}^{l \times m}$, be a matrix of $l \leq m$ linear constraints characterizing the family of networks under consideration, where $m = n(n-1)$ or $m = n(n-1)/2$ for, respectively, directed and undirected networks. Let $CR(\chi) = \{\mathbf{x} \in [0, 1]^m : F\mathbf{x} = \mathbf{b}\}$ be the continuous relaxation of the*

constraints. If \mathbf{b} is integer and F can be reduced by elementary row operations to a matrix, call it F' , with a unique unitary element (either $+1$ or -1) per column and all the elements of the same row with the same sign, then there is a bijection between the extreme points of $CR(\chi)$ and the set of networks under consideration.

Proof. In standard form, the system of linear constraints associated to $CR(\chi)$ is

$$\left[\begin{array}{c|c} I & I \\ \hline F' & \end{array} \right] \begin{bmatrix} \mathbf{x} \\ \mathbf{s} \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{b} \end{bmatrix}, \quad \begin{bmatrix} \mathbf{x} \\ \mathbf{s} \end{bmatrix} \geq 0. \quad (2)$$

From Theorem 1, the constraints matrix of (2) is TU by considering the following partition of rows: set the first m rows (associated to the identities) in \mathcal{J}_1 ; if elements of row i of F' are negative, then set this row in \mathcal{J}_1 ; otherwise, if they are positive, set the row in \mathcal{J}_2 . Therefore, all extreme points of $CR(\chi)$ are integer and they correspond to the AM of a network. In addition, no integer point may be located in the interior of $CR(\chi)$ since it is a subset of the unit hypercube, completing the proof. \square

In some cases there is no bijective relation between a family of conditional random networks and the extreme points of its polyhedron, since some basic solutions may be fractional. However, if we can ensure that no integer solution is in the interior of the polyhedron, this injective relation (i.e., any random network is associated to an extreme point, but not the opposite) is still useful, whenever some kind of acceptance-rejection technique is considered for fractional solutions. This is the case, for instance, of networks conditioned to the within group densities and in-and-out-degrees, which will be considered in Subsection 4.3.

2.1. Basic models of networks conditioned to linear constraints

One of the simplest cases is that of networks conditioned to the density d . The following result is immediate by noting that the system of linear constraints characterizing $CR(\chi)$ verifies the hypotheses of Proposition 1:

Proposition 2. *Let $CR(\chi) = \{\mathbf{x} \in [0, 1]^m : \sum_{(i,j) \in H} x_{ij} = d\}$, where H defined in (1) either relates to a directed or undirected graph. Then there is a bijection between the extreme points of $CR(\chi)$ and the set of graphs with n nodes and density d .*

In some situations, nodes might be partitioned into g different groups, $\gamma_1, \dots, \gamma_g$ and our interest might be to keep the within-group densities fixed when simulating random networks. Let Γ be the set of such groups and consider a function, $\theta : V \times V \rightarrow \Gamma \times \Gamma$, associating to each pair of nodes the pair of groups they belong to. The density constraint between group γ_k and γ_h , $1 \leq k \leq g$, $k \leq h \leq g$, is $\sum_{(i,j) \in H: \theta(i,j) = (\gamma_k, \gamma_h)} x_{ij} = d_{kh}$, where d_{kh} is a non-negative integer. Note that when $k = h$ we have a within group density constraint, otherwise a between group density constraint. Since the linear constraints characterizing these networks verify the hypotheses of Proposition 1, the following proposition holds:

Proposition 3. *Let $CR(\chi) = \{\mathbf{x} \in [0, 1]^m : \sum_{(i,j) \in H: \theta(i,j) = (\gamma_k, \gamma_h)} x_{ij} = d_{kh}, 1 \leq k \leq g, k \leq h \leq g\}$. Then there is a bijection between the extreme points of $CR(\chi)$ and the set of graphs with n nodes and within and between group densities d_{kh} .*

Another widely studied family of networks is that of undirected networks with fixed degree sequence [20, 21, 22], whose associated set of AMs is $\chi = \{\mathbf{x} \in \{0, 1\}^m : \sum_{j=1}^{i-1} x_{ji} + \sum_{j=i+1}^n x_{ij} = f_i, i = 1 \dots n\}$, where f_i is the degree of node i . Denoting these linear constraints as $F\mathbf{x} = \mathbf{f}$, we see that each column of F has two $+1$, thus it does not verify the hypothesis of Proposition 1, and the constraints matrix is not TU.

However, if we only add the constraints associated to the degrees of two particular nodes—with optionally the constraint associated to the number of edges (density)—the resulting matrix is TU by Proposition 1. The information we are conditioning in this case can be seen as a lower bound of the distance between the maximum and minimum degrees. Next proposition summarizes this result.

Proposition 4. *Let $i_1, i_2 \in V$ be two nodes with degrees f_{i_1} and f_{i_2} , \tilde{f}_{i_1} and \tilde{f}_{i_2} their degrees without considering the arcs (i_1, i_2) and (i_2, i_1) , and d the total number of edges in the network. Let $J(k, h) = \{j : 1 \leq j \leq n, j \neq k, j \neq h\}$ and $CR(\chi) = \{\mathbf{x} \in [0, 1]^m : \sum_{(i,j) \in H} x_{ij} = d; \sum_{j \in J(i_1, i_2)} x_{ij} = \tilde{f}_i, i = i_1, i_2\}$. Then, there is a bijection between the extreme points of $CR(\chi)$ and the set of graphs with n nodes, d edges and degree range greater than or equal to $|f_{i_1} - f_{i_2}|$. This same result holds if the density constraint $\sum_{(i,j) \in H} x_{ij} = d$ is removed from $CR(\chi)$.*

2.2. Conditional edge-colored networks

The researcher might sometimes be interested in studying networks whose edges are associated to a categorical value (color), generally known under the name of edge-colored networks. The study of edge-colored graphs (i.e., graphs with different types of edges) has given rise to important developments during the last few decades. From the point of view of applicability, problems arising in molecular biology are often modeled using edge-colored graphs [25], and the problem of interpersonal ties in social networks might be also modeled considering different types of arcs [14].

Let \mathcal{C} be a given set of colors, $|\mathcal{C}| \geq 2$. Formally, an edge-colored graph is a tuple $G_{\mathcal{C}} = (V, E, \tau)$, where V and E are the sets of nodes and edges, respectively, and $\tau : E \rightarrow \mathcal{C}$ a function assigning a color to each edge. They are, in some sense, related to multicommodity networks. Edge-colored graphs can be modeled as

$$\begin{aligned} \sum_{c=1}^{|\mathcal{C}|} x_{ij}^c &\leq 1 & (i, j) \in H \\ x_{ij}^c, x_{ji}^c &\in \{0, 1\} & (i, j) \in H, c = 1, \dots, |\mathcal{C}| \end{aligned} \quad (3)$$

where x_{ij}^c is 1 if an arc with color c from node i to node j exists, and 0 otherwise, and H was defined in (1). The first set of constraints of (3)—multicommodity or generalized upper bounding constraints—complicate the structure of the constraints matrix for some structural properties, such as the total number of edges, the number of edges per color, and the lower bound of the degree range.

In the case of edge-colored networks conditioned to having d_c edges per color c , the constraints $\sum_{(i,j) \in \mathcal{H}^2} x_{ij}^c = d_c, c = 1, \dots, |\mathcal{C}|$ should be included.

Proposition 5. *By adding within-color densities constrains the coefficient matrix associated to system (3) becomes*

$$\left[\begin{array}{cccc|cccc} I & & & & I & & & \\ & I & & & & I & & \\ & & \ddots & & & & \ddots & \\ & & & I & & & & I \\ \hline F & & & & & & & \\ & F & & & & & & \\ & & \ddots & & & & & \\ & & & F & & & & \\ \hline G & G & \dots & G & I & & & \end{array} \right] \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{x}^2 \\ \vdots \\ \mathbf{x}^{|\mathcal{C}|} \\ \mathbf{s}^1 \\ \mathbf{s}^2 \\ \vdots \\ \mathbf{s}^{|\mathcal{C}|} \end{bmatrix} = \begin{bmatrix} \mathbf{e} \\ \mathbf{e} \\ \vdots \\ \mathbf{e} \\ \mathbf{d} \\ \mathbf{e} \end{bmatrix}. \quad (4)$$

where $\mathbf{d} \in \mathbb{Z}^{|\mathcal{C}|}$ is the vector of within-color densities, $F = \mathbf{e}^T$ and $G = I$. As matrix $F' = [\mathbf{e} \ I]^T$ verifies the conditions of Proposition 1, the coefficient matrix of (4) is TU.

Note that the slack of the inequality of $\sum_{c=1}^{|\mathcal{C}|} x_{ij}^c \leq 1$ is obtained in (4) by defining an auxiliary color with no specified within-color density, c_a and an extended set of colors $\mathcal{C}^* = \mathcal{C} \cup \{c_a\}$.

The two models of networks described in propositions 3 and 5 can be combined to obtain another family of network which is also characterized by a TU system. Consider an edge-colored network where nodes are partitioned into g different groups: $\gamma_1, \dots, \gamma_g$. Since the connections within members of the same group might have different colors, our interest is to keep the within-color and between-group (or within-group) densities fixed when simulating random networks. The number of c -color edges between group γ_k and γ_h , is $\sum_{(i,j) \in \mathcal{H}^2: \theta(i,j)=(\gamma_k, \gamma_h)} x_{ij}^c = d_{kh}^c$, where d_{kh}^c is a non-negative integer, for $c \in \mathcal{C}$, $1 \leq k \leq g$, $k \leq h \leq g$. Note that when $k = h$ we have a within group density constraint, otherwise a between group density constraint. This system of linear constraints verifies the hypotheses of Proposition 1.

Proposition 6. Consider the edge-colored undirected graph conditioned to the within-color and within-group densities. Let $CR(\chi)$ be the subset of the m -dimensional unitary cube (that is $\mathbf{x} \in [0, 1]^m$), verifying the following system

$$\begin{aligned} \sum_{c \in \mathcal{C}} x_{ij}^c &\leq 1 & (i, j) \in \mathcal{H}^2 \\ \sum_{\theta(i,j)=(\gamma_k, \gamma_k)} x_{ij}^c &= d_k^c & c \in \mathcal{C}, 1 \leq k \leq g \\ \sum_{(i,j) \in \mathcal{H}^2} x_{ij}^c &= d^c & c \in \mathcal{C} \\ x_{ij}^c, x_{ji}^c &\in \{0, 1\} & (i, j) \in \mathcal{H}^2, c \in \mathcal{C} \end{aligned} \tag{5}$$

whose corresponding extended matrix form is equal to (4), except for the fact that $\mathbf{d} \in \mathbb{Z}^{|\mathcal{C}|(g+1)}$ is now the vector of within-color and within-group densities and $F \in \{0, 1\}^{g+1 \times m}$ is a matrix which can be reduced by elementary row operations to contain a unique unitary element of the same sign per column. As $F' = [F \ I]^T$ verifies the conditions of Proposition 1 the described system is TU.

Another case in which (4) is TU is obtained when the linking constraints are associated to the within-color degree of two nodes and within-color densities, as the family of network described in Proposition 4, so that $F \in \{0, 1\}^{3 \times n(n-1)/2}$.

2.3. Classes of real-valued networks

The described families of edge-colored networks are ways of modeling *non-binary connections* between nodes, i.e. edges associated to multidimensional properties (colors). The main concern with this class of models is the size of his mathematical programming representation (number of decision variables and constraints) when the number of colors grow large. A possible way of dimensionality reduction is to aggregate the information of edge colors in the corresponding real-valued edge.

Formally, a real-valued graph is a tuple $G_\nu = (V, E, \nu)$, where V and E are the sets of nodes and edges, respectively, and $\nu : E \rightarrow [0, 1] \subset \mathbb{R}$ a function assigning a real-value to each edge. By doing so the researcher is assuming a probabilistic model where any symmetric matrix in $[0, 1]^{n \times n}$ verifying a specified system of linear constraints $A\mathbf{x} = \mathbf{b}$ has a non-null probability density of being observed. Note that a full characterization of these classes of

networks does not require matrix A to be TU any more, as fractional solutions still belong to the specified families.

Classes of real-valued networks can be defined by relaxing the integrality condition of all families of networks presented in subsections 2.1 and 2.2. In particular relaxing the integrality conditions allows for a wider modeling possibility of families of networks by systems of linear constraints with arbitrary coefficient matrix A .

A simple example related to the previously defined model is the class of real-valued networks with fixed between-group valued-densities and valued-degrees: $CR(\chi) = \{\mathbf{x} \in [0, 1]^m : \sum_{j \in V} x_{ij} = f_i; \sum_{(i,j) \in H: \theta(i,j) = (\gamma_k, \gamma_h)} x_{ij} = d_{kh}, 1 \leq k \leq g, k \leq h \leq g\}$, where d_{kh} is the density between group γ_k and γ_h , $1 \leq k \leq g, k \leq h \leq g$, as described in Proposition 3, and f_i is the valued-degree of node $i \in V$, as described in Proposition 4.

3. Generating random networks by Linear Programming

The previous section provided an algebraic characterization of several families of networks by linear constraints. From a constructive point of view, this section proposes LP-based polynomial-time methods to generate instances of networks with given structural properties. Broadly speaking, the goal is to obtain random networks as the solution of a LP by randomly changing the cost vector. To validate these LP-based procedures, first we will derive the probability density function of the primal-dual LP solutions when the cost vector randomly changes. Although this probability density may not be uniform, we will see it can be used to obtain solutions of uniform distribution by considering the acceptance-rejection Metropolis-Hastings method [27], widely used in Statistics.

3.1. The probability of primal-dual solutions

Let $CR(\chi) = \{\mathbf{x} \in [0, 1]^{n'} : A\mathbf{x} = \mathbf{b}\}$ be a polytope whose set of extreme points is bijectively related to a given family of networks and consider the associated (feasible and bounded) LP

$$\min \mathbf{c}^T \mathbf{x} \text{ s. to } A\mathbf{x} = \mathbf{b}, 0 \leq \mathbf{x} \leq 1, \quad (6)$$

where $A \in \mathbb{R}^{m' \times n'}$ is a full row rank matrix, $\mathbf{b} \in \mathbb{R}^{m'}$ and $\mathbf{c} \in \mathbb{R}^{n'}$. By adding slacks, (6) can be written in standard form as

$$\min \hat{\mathbf{c}}^T \hat{\mathbf{x}} \text{ s. to } \hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}, \hat{\mathbf{x}} \geq 0, \quad (7)$$

where $\hat{A} \in \mathbb{R}^{\hat{m} \times \hat{n}}$ ($\hat{m} = m' + n'$ and $\hat{n} = 2n'$) is a full row rank matrix, $\hat{\mathbf{b}} = [\mathbf{b}^T \ \mathbf{e}^T]^T \in \mathbb{R}^{\hat{m}}$ and $\hat{\mathbf{c}} = [\mathbf{c}^T \ \mathbf{0}^T]^T \in \mathbb{R}^{\hat{n}}$. If the gradient of the objective function $\hat{\mathbf{c}}$ is a properly defined random vector of density function $f_G(\hat{\mathbf{c}})$, then the solution of (7) is also a random vector whose probability distribution can be computed as

$$P(\hat{\mathbf{x}}) = \int_{G^{-1}(\hat{\mathbf{c}}) = \hat{\mathbf{x}}} f_G(\hat{\mathbf{c}}) d\hat{\mathbf{c}}, \quad (8)$$

where $G(\hat{\mathbf{x}})$ is the set of gradients for which $\hat{\mathbf{x}}$ is an optimal solution and $G^{-1}(\hat{\mathbf{c}})$ is the set of optimal solutions of (7) for a fixed objective gradient $\hat{\mathbf{c}}$. Defined this way, G would be a set-to-set mapping. We thus consider an alternative definition, as shown by next Lemma, which guarantees a point-to-point function:

Lemma 1 (Dimensionality augmentation). *Consider the augmented sample space obtained by introducing the auxiliary random vectors $\hat{\mathbf{y}}$ and $\hat{\mathbf{z}}$ and the joint probability function*

$$P(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) = \int_{G^{-1}(\hat{\mathbf{c}}, \mathbf{0}, \mathbf{0}) = (\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}})} f_C(\hat{\mathbf{c}}) d\hat{\mathbf{c}}, \quad (9)$$

where $G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}})$ is defined through the KKT optimality conditions as

$$G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) \triangleq \begin{bmatrix} \hat{A}^T \hat{\mathbf{y}} + \hat{\mathbf{z}} \\ \hat{A} \hat{\mathbf{x}} - \hat{\mathbf{b}} \\ \hat{X} \hat{Z} \mathbf{e} \end{bmatrix} \quad \text{and} \quad (\hat{\mathbf{x}}, \hat{\mathbf{z}}) \geq 0, \quad (10)$$

$\hat{\mathbf{y}}$ and $\hat{\mathbf{z}}$ being respectively the Lagrangian multipliers of the equations and bounds of (7), and \hat{X} and \hat{Z} diagonal matrices made up with the components of $\hat{\mathbf{x}}$ and $\hat{\mathbf{z}}$.

We claim that G is a one-to-one continuously differentiable function from $\mathcal{I}_{D0} = \{(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) \in \mathbb{R}^{2n'+m'} : \hat{\mathbf{x}} > 0, \hat{\mathbf{z}} > 0, \hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}, \hat{A}^T \hat{\mathbf{y}} + \hat{\mathbf{z}} = \hat{\mathbf{c}}, \hat{X}\hat{Z}\mathbf{e} = \mathbf{0}\mathbf{e}\}$ to $\mathcal{I}_{C0} = \mathbb{R}^{n'} \times \{0\}^{m'+n'}$.

Proof. To proof that G is a one-to-one function we only need to note that the Lagrangian multipliers associated to an optimal solution of an LP are unique, as long as \hat{A} is full rank (see [35]). The continuous differentiability of G in \mathcal{I}_0 results from its Jacobian

$$J(G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}})) = \begin{bmatrix} \hat{A}^T & I \\ \hat{A} & \\ \hat{Z} & \hat{X} \end{bmatrix}. \quad (11)$$

□

Theorem 2 (Change of variables in integral calculus [12]). *Let $G : \mathcal{I}_D \rightarrow \mathcal{I}_C$ be a one-to-one and continuously differentiable map of the open set \mathcal{I}_D into \mathcal{I}_C , such that $J(G(\mathbf{t}))$ is nonsingular for all $\mathbf{t} \in \mathcal{I}_D$. If $f : \mathcal{I}_C \rightarrow \mathbb{R}$ is a non-negative locally integrable function, then*

$$\int_{\mathcal{I}_D} f(G(\mathbf{t})) |\det J(G(\mathbf{t}))| d\mathbf{t} = \int_{\mathcal{I}_C} f(\mathbf{s}) d\mathbf{s}, \quad (12)$$

where the symbol $|\det J(G(\mathbf{t}))|$ is used to denote the absolute value of the determinant of the Jacobian of G at point $\mathbf{t} \in \mathcal{I}_D$. By the inverse-function theorem, \mathcal{I}_C is open and the inverse point mapping $G^{-1}(\mathbf{s})$ is continuously differentiable.

The applicability of Theorem 2 to solve the integral in (9) is conditioned to the nonsingularity of $J(G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}))$, which cannot be claimed in the general case. (See Subsection 3.2 for special conditions of nonsingularity.)

However, it can be possible to guarantee the nonsingularity of $J(G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}))$ into a redefined domain $\mathcal{I}_{D\mu} = \{\mathbf{t} = (\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) : \hat{\mathbf{x}} > 0, \hat{\mathbf{z}} > 0, \hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}, \hat{A}^T \hat{\mathbf{y}} + \hat{\mathbf{z}} = \hat{\mathbf{c}}, \hat{X}\hat{Z}\mathbf{e} = \mu\mathbf{e}\}$ for some $\mu \in \mathbb{R}$, $\mu > 0$, which is defined based on the KKT- μ perturbed conditions

$$G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) = \begin{bmatrix} \hat{\mathbf{c}} \\ \mathbf{0} \\ \mu\mathbf{e} \end{bmatrix}. \quad (13)$$

The codomain of G is thus $\mathcal{I}_{C\mu} = \{\mathbf{s} = (\hat{\mathbf{c}}, \mathbf{0}, \mu\mathbf{e}) : \mu \in \mathbb{R}, \mu > 0, \hat{\mathbf{c}} \in \mathbb{R}^{\hat{n}}\}$. For a fixed $\hat{\mathbf{c}}$, the set of solutions $(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$ of (13) for $\mu > 0$ is an arc of feasible points known as the primal-dual central path [35, 36], which is widely used in interior-point methods. When $\mu \rightarrow 0$, the central path converges to an optimal solution of (7). If instead of a unique solution we have an optimal face, then the central path converges to the single analytic

center of this optimal face [36]. If the primal-dual strictly feasible set of (7) is nonempty (i.e., $\mathcal{F}^0 = \{(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) : \hat{\mathbf{x}} > 0, \hat{\mathbf{z}} > 0, \hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}, \hat{A}^T\hat{\mathbf{y}} + \hat{\mathbf{z}} = \hat{\mathbf{c}}\} \neq \emptyset$), then the central path exists and it is unique for each $\mu > 0$ (see [35, 36] for a proof of this result). This uniqueness guarantees that, given $\mathbf{s} = (\hat{\mathbf{c}}, \mathbf{0}, \mu\mathbf{e})$, there is a single point $\mathbf{t} = (\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$ satisfying (13). The adopted notation makes explicit the dependence of the primal-dual points with respect to the complementarity parameter μ .

Lemma 2 (Nonsingularity of the Jacobian). *The Jacobian $J(G(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu)))$ —which is the matrix of the Newton system to be solved at each iteration of primal-dual interior-point methods—is nonsingular if \hat{A} is full row rank and $(\hat{\mathbf{x}}(\mu), \hat{\mathbf{z}}(\mu)) > 0$.*

Proof. A simpler expression for the determinant can be derived if we are in an interior point, i.e., if $\mu > 0$, thus $\hat{\mathbf{x}}(\mu) > 0$ and $\hat{\mathbf{z}}(\mu) > 0$. Denoting by J the Jacobian, and adding to the first block-row of J the last block-row multiplied by the diagonal matrix $\hat{X}^{-1}(\mu)$ we have

$$\det(J) = \det(J_1) \prod_{i=1}^n \hat{\mathbf{x}}_i(\mu) \quad \text{where} \quad J_1 = \begin{bmatrix} -\hat{X}^{-1}(\mu)\hat{Z}(\mu) & \hat{A}^T \\ \hat{A} & \end{bmatrix}.$$

Since $J_1 = CDC^T$, where

$$C = \begin{bmatrix} I & \\ -\hat{A}\hat{X}(\mu)\hat{Z}^{-1}(\mu) & I \end{bmatrix} \quad \text{and} \quad D = \begin{bmatrix} -\hat{X}^{-1}\hat{Z} & \\ \hat{A}\hat{X}(\mu)\hat{Z}^{-1}(\mu)\hat{A}^T & \end{bmatrix},$$

using that $\det(C) = 1$ we finally obtain

$$\det(J) = \det(D) \prod_{i=1}^n \hat{\mathbf{x}}_i(\mu) = (-1)^n \det(\hat{A}\hat{X}(\mu)\hat{Z}^{-1}(\mu)\hat{A}^T) \prod_{i=1}^n \hat{\mathbf{z}}_i(\mu). \quad (14)$$

$\hat{A}\hat{X}(\mu)\hat{Z}^{-1}(\mu)\hat{A}^T$ is the symmetric and positive definite matrix—if \hat{A} has full row rank—of the normal equations system of interior-point methods and $(\hat{\mathbf{x}}(\mu), \hat{\mathbf{z}}(\mu)) > 0$ [35]. \square

The applicability of Theorem 2 when $G : \mathcal{I}_{D\mu} \rightarrow \mathcal{I}_{C\mu}$ allows computing the probability density function of $\mathbf{t} = G^{-1}(\mathbf{s})$ from that of the random variable \mathbf{s} . According to this result:

$$f_{\mathcal{I}_{D\mu}}(\mathbf{t}) = f_{\mathcal{I}_C}(\mathbf{s}) |\det(J(G(\mathbf{t})))| = f_{\mathcal{I}_{C\mu}}(\mathbf{s}) \left| \det \begin{bmatrix} \hat{A}^T & I \\ \hat{A} & \hat{X}(\mu) \\ \hat{Z}(\mu) & \end{bmatrix} \right|, \quad (15)$$

where $f_{\mathcal{I}_{D\mu}}$ and $f_{\mathcal{I}_{C\mu}}$ denote the probability density functions (PDF, from now on) of \mathbf{t} and \mathbf{s} , respectively.

3.2. The KKT- μ perturbed conditions for classes of real-valued networks

The discussion of Section 3 started by considering a polytope $CR(\chi) = \{\mathbf{x} \in [0, 1]^{n'} : \mathbf{A}\mathbf{x} = \mathbf{b}\}$, whose set of extreme points is bijectively related to a given family of networks, and the associated LP in standard form (7). Let $\text{Int}(CR(\chi))$ be the interior of $CR(\chi)$. The closed-form expression (15), associated to the augmented sample space of primal-dual solutions, suggests a proper definition of a PDF of $\text{Int}(CR(\chi))$ (primal strictly feasible solution of (7)), based on the KKT- μ perturbed conditions.

Let f_C and f_μ be the absolutely continuous PDFs (i.e., sets of zero Lebesgue measure have zero probability) of the cost vector $\hat{\mathbf{c}}$ and complementarity parameter μ . We assume $f_{\mathcal{I}_{C\mu}}$ to have the following form:

$$f_{\mathcal{I}_{C\mu}}\left(\mathbf{s} = \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \mathbf{s}_3 \end{bmatrix}\right) = \begin{cases} f_C(\mathbf{s}_1)f_\mu(\mathbf{s}_3) & \text{if } \mathbf{s} \in \mathcal{I}_{C\mu} \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

Therefore, under the transformation G we finally have, for $\mathbf{t} = (\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) \in \mathcal{I}_{D\mu}$:

$$f_{\mathcal{I}_{D\mu}}(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu)) = f_C(\hat{\mathbf{c}})f_\mu(\mu) \left| \det \begin{bmatrix} & \hat{A}^T & I \\ \hat{A} & & \\ \hat{Z}(\mu) & & \hat{X}(\mu) \end{bmatrix} \right|. \quad (17)$$

Note that the support of $f_{\mathcal{I}_{D\mu}}$ is $\mathcal{I}_{D\mu}$, so that it only provides the probability density of central path points for some particular μ .

Theorem 3. *If f_C and f_μ have non-null probability density over $\mathbb{R}^{n'}$ and \mathbb{R} respectively, then every $\hat{\mathbf{x}} \in \text{Int}(CR(\chi))$ has a non-null probability density.*

Proof. Let $\mathcal{C}_\mu(\hat{\mathbf{c}}) = \{\hat{\mathbf{x}}(\mu) \mid \exists \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu) > 0, G(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu)) = (\hat{\mathbf{c}}, \mathbf{0}, \mu\mathbf{e}), \mu > 0\}$ be the primal central path of (7), for a fixed gradient $\hat{\mathbf{c}} \in \mathbb{R}^{n'}$, and define

$$\mathcal{C}_\cup = \bigcup_{\hat{\mathbf{c}} \in \mathbb{R}^{n'}} \mathcal{C}_\mu(\hat{\mathbf{c}}). \quad (18)$$

and

$$f_{\mathcal{I}_{D\mu}}(\hat{\mathbf{x}}(\mu)) = \int_{\varphi} f_{\mathcal{I}_{D\mu}}(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu)) d\hat{\mathbf{y}}d\hat{\mathbf{z}} \quad (19)$$

where $\varphi = \{(\hat{\mathbf{y}}, \hat{\mathbf{z}}) \mid \hat{A}^T \hat{\mathbf{y}} + \hat{\mathbf{z}} = \hat{\mathbf{c}}, \hat{X} \hat{\mathbf{z}} = \mu \mathbf{e}\}$. If f_C and f_μ have non-null probability over $\mathbb{R}^{n'}$ and \mathbb{R} , then \mathcal{C}_\cup is the support of (17), so that a sufficient condition for Theorem 3 to hold is that $\text{Int}(CR(\chi)) \subseteq \mathcal{C}_\cup$ —as the opposite inclusion it trivially true.

Hence, the proof consists in showing that for any $\hat{\mathbf{x}} \in \text{Int}(CR(\chi))$, there exists $\hat{\mathbf{c}} \in \mathbb{R}^{n'}$ and $\mu > 0$, such that $G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}) = (\hat{\mathbf{c}}, \mathbf{0}, \mu\mathbf{e})$, for some Lagrangian multipliers $\hat{\mathbf{y}}$ and $\hat{\mathbf{z}} > 0$. Due to the μ -complementarity ($\hat{X} \hat{\mathbf{z}} = \mu \mathbf{e}$), the required sufficient condition is the existence of $\hat{\mathbf{c}} \in \mathbb{R}^{n'}$ and $\mu > 0$, such that for any $\hat{\mathbf{x}}$ verifying $\hat{A} \hat{\mathbf{x}} = \hat{\mathbf{b}}, \hat{\mathbf{x}} > 0$, the system $\hat{A}^T \hat{\mathbf{y}} + \mu \hat{X}^{-1} \mathbf{e} = \hat{\mathbf{c}}$ has a solution for some $\hat{\mathbf{y}} \in \mathbb{R}^{m'}$. And the statement immediately follows (for instance, setting $\hat{\mathbf{y}} = \mathbf{0}$ and $\hat{\mathbf{c}} = \mu \hat{X}^{-1} \mathbf{e} > 0$). \square

The ability to randomly generate primal strictly feasible points and the availability of a closed-form expression of their PDF (17) allow for a straightforward extension of the proposed methodology to simulate from families of real-valued networks with linear constraints. In fact, as it will be shown in Subsection 3.4, the applicability of well known statistical simulation methodologies (such as the Metropolis-Hastings algorithm) to generate strictly feasible points from arbitrary distribution is guaranteed by (17).

3.3. Asymptotic behaviour of $f_{\mathcal{I}_{D\mu}}$

The use of the central path in subsections 3.1 and 3.2 resulted in a general methodology to provide arbitrarily close approximations of the extreme points and to extend the range of

applicability of the described random graph simulation procedure to the case of real-valued networks.

In particular, as far as families of binary networks (characterized by the set of extreme points of algebraically defined polytopes) are concerned, the use of the central-path in the redefined sample space $\mathcal{I}_{D\mu}$ in Subsection 3.1 allowed bypassing the problem of the nonsingularity of the $J(G(\hat{\mathbf{x}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}))$ for a correct applicability of Theorem 2, by considering arbitrarily close μ -approximations of the extreme points. (It must be noted that in practice numerical solvers require such kind of numerical approximations of the optimal solution, which are governed by some optimality tolerances.)

In this subsection the convergence of $f_{\mathcal{I}_{D\mu}}(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$ to $f_{\mathcal{I}_{D0}}(\hat{\mathbf{x}}(0), \hat{\mathbf{y}}(0), \hat{\mathbf{z}}(0))$ is assessed, based on the continuity of the central path in $\mu = 0$, and the nullity of the set of critical values of $G(\hat{\mathbf{x}}(0), \hat{\mathbf{y}}(0), \hat{\mathbf{z}}(0))$. For the latter we will need Sard's Lemma, which is provided below.

Lemma 3 (Sard's Lemma). *Let $G : \mathcal{S} \subseteq \mathbb{R}^q \rightarrow \mathcal{T} \subseteq \mathbb{R}^p$ be a one-to-one and continuously differentiable map of the open set \mathcal{S} into \mathcal{T} . Let \mathcal{L}^q denote the q -dimensional Lebesgue-measure and Ξ the set of critical values of G . Then Ξ has \mathcal{L}^q -measure zero.*

Proof. See [1, Paragraph 15] and [32, Theorem II.3.1]. □

As just mentioned in Subsection 3.1, the support of $f_{\mathcal{I}_{D\mu}}$ is $\mathcal{I}_{D\mu}$, so that it only provides the probability of central path points for some particular μ . As $\mu \rightarrow 0$ the central path points converge to the solution of (7), and then (15) converges to the probability distribution of the primal-dual solutions of the LP problem in terms of the probability distribution of the cost vector, as shown by the next theorem:

Theorem 4. *The probability density function of the μ -parameterized primal-dual solutions converge to the one of the optimal primal-dual solutions, that is,*

$$\lim_{\mu \rightarrow 0} f_{\mathcal{I}_{D\mu}}(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu)) = f_{\mathcal{I}_{D0}}(\hat{\mathbf{x}}(0), \hat{\mathbf{y}}(0), \hat{\mathbf{z}}(0)). \quad (20)$$

as long as $f_\mu(\mu) > 0$, for all $\mu \geq 0$ (mild condition).

Proof. The first part of the proof consists in showing that the limit exists. After taking for granted the existence of the limit, we show that this limit is the desired PDF.

- *Existence of the limit*

By [36, Theorem 2.17], the points $(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$ on the central path are bounded, the central path converges to $(\hat{\mathbf{x}}(0), \hat{\mathbf{y}}(0), \hat{\mathbf{z}}(0))$, and $\hat{\mathbf{x}}(0)$ and $\hat{\mathbf{z}}(0)$ are, respectively, the analytic centers of the primal and dual optimal faces (i.e., for any $\hat{\mathbf{c}}$ the central path converges to a unique point—the analytic center of the optimal face—even if there are multiple solutions for this $\hat{\mathbf{c}}$). Therefore, (15) exists in \mathcal{I}_{D0} , as the determinant of (15) computed at $\hat{\mathbf{Z}}(0)$ and $\hat{\mathbf{X}}(0)$ is bounded, and $\lim_{\mu \rightarrow 0} f_{\mathcal{I}_{D\mu}}(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$ exists by continuity of the determinant.

- *Convergence in distribution*

Let $\Psi_\mu = \mathcal{I}_{D\mu} \setminus \Xi$, i.e. the subset of $\mathcal{I}_{D\mu}$ obtained by removing the set of critical points Ξ of $G(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$. For $\mu > 0$, Lemma 2 ensures that $\Xi = \emptyset$. For $\mu = 0$, by Sard's Lemma, Ξ has \mathcal{L}^q -measure zero, since G is a bijection from \mathcal{I}_{D0} to \mathcal{I}_{C0} . This fact allows the applicability of Theorem 2 in Ψ_μ . Thus, as long as $f_\mu(\mu) > 0$, for all

$\mu \geq 0$, we claim that for any (Lebesgue-integrable) real-valued function $g : \mathcal{I}_{D\mu} \rightarrow \mathbb{R}$, the following equality must be true:

$$\mathbb{E}_\mu[g(\mathbf{t})] = \int_{\Psi_\mu} g(\mathbf{t}) f_{\mathcal{I}_{D\mu}}(\mathbf{t}) d\mathbf{t} = \int_{\mathcal{I}_{D\mu}} g(\mathbf{t}) f_{\mathcal{I}_{D\mu}}(\mathbf{t}) d\mathbf{t} \quad (21)$$

and $\lim_{\mu \rightarrow 0} \mathbb{E}_\mu[g(\mathbf{t})] = \mathbb{E}_0[g(\mathbf{t})]$.

□

The above results are illustrated by this small example.

Example 1. Consider the small two-dimensional problem $\min c_1 x_1 + c_2 x_2$ s. to $x_1 + M x_2 = 1, (x_1, x_2) \geq 0$ ($M > 0$ being a given parameter), which matches (7) for $\hat{A} = [1 \ M]$ and $\hat{b} = 1$. The feasible region of this problem is the segment between $V_A = (1, 0)$ and $V_B = (0, 1/M)$. When $c_2 > M c_1$, V_A is the optimal extreme point; when $c_2 < M c_1$, the solution is V_B ; when $c_2 = M c_1$ (which is unlikely if the cost vector is randomly generated) the whole segment is optimal. Unless $M = 1$, the probability of V_A and V_B being optimal is not uniform, even if the cost vector is randomly generated (in particular we have $P(V_A) = 1 - M/2$ and $P(V_B) = M/2$).

The associated KKT- μ perturbed conditions are

$$\begin{aligned} x_1 + M x_2 &= 1 \\ y + z_1 &= c_1 \\ M y + z_2 &= c_2 \\ x_1 z_1 &= \mu \\ x_2 z_2 &= \mu \\ (x_1, x_2, z_1, z_2) &> 0 \end{aligned} \quad (22)$$

and, after a few manipulations from the dual feasibility and complementarity conditions, we obtain:

$$M c_1 + \frac{M \mu}{1 - x_1} = c_2 + \frac{M \mu}{x_1} \quad \text{and} \quad M c_1 + \frac{\mu}{x_2} = c_2 + \frac{M \mu}{1 - M x_2}. \quad (23)$$

When $c_2 \neq M c_1$, the central path is obtained by solving the two quadratic equations (23) with respect to $x_1(\mu)$ and $x_2(\mu)$:

$$\begin{aligned} x_1(\mu) &= \frac{c_2 - M c_1 - 2M \mu + \sqrt{\Delta}}{2(c_2 - M c_1)}, \quad x_2(\mu) = \frac{c_2 - M c_1 + 2M \mu - \sqrt{\Delta}}{2M(c_2 - M c_1)}, \\ \Delta &= 4M^2 \mu^2 + (c_2 - M c_1)^2, \quad z_1(\mu) = \frac{\mu}{x_1(\mu)}, \quad z_2(\mu) = \frac{\mu}{x_2(\mu)}, \quad y(\mu) = c_1 - z_1(\mu). \end{aligned} \quad (24)$$

When $c_2 = M c_1$, we directly have

$$x_1(\mu) = \frac{1}{2}, \quad x_2(\mu) = \frac{1}{2M}, \quad z_1(\mu) = 2\mu, \quad z_2(\mu) = 2M\mu, \quad y(\mu) = c_1 - 2\mu. \quad (25)$$

For $c_2 \neq M c_1$, the limit point of the central path can be obtained either by direct substitution (as for $x_1(\mu)$, $x_2(\mu)$) or by applying L'Hôpital's rule (as for $z_1(\mu)$, $z_2(\mu)$):

$$\begin{aligned} \lim_{\mu \rightarrow 0} x_1(\mu) &= 1 & \lim_{\mu \rightarrow 0} x_1(\mu) &= 0 \\ \lim_{\mu \rightarrow 0} x_2(\mu) &= 0 & \lim_{\mu \rightarrow 0} x_2(\mu) &= \frac{1}{M} \\ \lim_{\mu \rightarrow 0} z_1(\mu) &= 0 & \lim_{\mu \rightarrow 0} z_1(\mu) &= \frac{M c_1 - c_2}{M} \text{ for } c_2 < M c_1. \\ \lim_{\mu \rightarrow 0} z_2(\mu) &= c_2 - M c_1 & \lim_{\mu \rightarrow 0} z_2(\mu) &= 0 \\ \lim_{\mu \rightarrow 0} y(\mu) &= c_1 & \lim_{\mu \rightarrow 0} y(\mu) &= \frac{c_2}{M} \end{aligned} \quad \text{for } c_2 > M c_1, \quad \text{and}$$

For $c_2 \neq M c_1$, the primal solution $(x_1(\mu), x_2(\mu))$ does not depend on μ and it provides the analytic center of the feasible primal segment, as expected. The dual limit point is $z_1(0) = z_2(0) = 0$, $y(0) = c_1$.

By (17), the asymptotic value of the density function of primal-dual solutions is

$$f_{\mathcal{I}_{D\mu}}(x_1(\mu), x_2(\mu), y(\mu), z_1(\mu), z_2(\mu)) = f_{I_c}(c_1, c_2) |\det(J)|, \quad (26)$$

where in this simple problem $\det(J) = M^2 x_2(\mu) z_1(\mu) + x_1(\mu) z_2(\mu)$. Therefore, when $\mu \rightarrow 0$ we have that

$$\det(J(x_1(0), x_2(0), z_1(0), z_2(0))) = \begin{cases} (c_2 - Mc_1) & \text{if } c_2 > Mc_1 \\ (Mc_1 - c_2) & \text{if } c_2 < Mc_1 \\ 0 & \text{if } c_2 = Mc_1. \end{cases} \quad (27)$$

First, we note by (27) that the set of points with 0 determinant (i.e., the critical values $\{(c_1, c_2) \in \mathbb{R}^2 : c_2 = Mc_1\}$) has 0 measure in \mathbb{R}^2 , as expected by Sard's Lemma 3. The density of those points is 0, according to our development. We also see from (26) and (27) that the probability density of the primal solutions V_A and V_B increase, respectively, with $c_2 - Mc_1$ and $c_1 - Mc_2$, which is consistent with the solution of the primal problem.

It is worth to make some observations to (17) and Theorem 4:

- From (17), the probability density function of the primal-dual solutions depends on the randomness of $\hat{\mathbf{c}}$ and μ , but also on the feasible polyhedron defined by the constraints matrix \hat{A} , which appears in the determinant of the Jacobian of G . This is coherent with the intuition.
- Given a primal-dual central path point $(\hat{\mathbf{x}}(\mu), \hat{\mathbf{y}}(\mu), \hat{\mathbf{z}}(\mu))$ for some cost vector $\hat{\mathbf{c}}$, equation (17), which is easily computed, provides the probability density $f_{\mathcal{I}_{D\mu}}$ of this primal-dual point. However, the probability distribution of the primal points is obtained as the marginal distribution with respect of the Lagrangian multipliers, as in (19). Although (19) is a difficult integral, the expression of the density (17) is enough to compute primal-dual solutions with any desired distribution (as seen in below Subsection 3.4).
- A closed form expression for (17) can not be computed, in general, because of the determinant of the Jacobian of G , which involves the constraints matrix \hat{A} and the values of $(\hat{\mathbf{x}}(\mu), \hat{\mathbf{z}}(\mu))$ in the central path (which do not admit a closed form expression, but for toy problems).
- Consistently with Sard's Lemma, Example 1 shows that the set of objective gradients such that the optimal solution to the corresponding LP is not a unique vertex has measure zero. As a corollary of Theorem 4, eliminating such a set of points from the support of f_C does not affect the moments of the primal-dual distribution.

3.4. Computing solutions with uniform distribution

Due to the availability of a computable probability density function (17), we can guarantee the uniform distribution (indeed, any distribution) of the primal-dual solutions, by a proper application of *Markov Chain Monte Carlo* methods. One of these approaches is the *Metropolis-Hastings* algorithm, which is a well-known method to sample from probability distributions for which direct sampling is not possible, though a well defined probability density function is available. A detailed derivation of this algorithm, which is out of the scope of this work, can be found in [27]. We just outline the main steps of this procedure.

When applied to our problem, this method consists in generating a stochastic sequence of primal-dual solutions, from an arbitrary starting point $\mathbf{t}^0(\mu) = (\hat{\mathbf{x}}^0(\mu), \hat{\mathbf{y}}^0(\mu), \hat{\mathbf{z}}^0(\mu))$, and the following rule to go from a current state $\mathbf{t}^k(\mu) = (\hat{\mathbf{x}}^k(\mu), \hat{\mathbf{y}}^k(\mu), \hat{\mathbf{z}}^k(\mu))$ to a new state $\mathbf{t}^{k+1}(\mu) = (\hat{\mathbf{x}}^{k+1}(\mu), \hat{\mathbf{y}}^{k+1}(\mu), \hat{\mathbf{z}}^{k+1}(\mu))$:

1. Initialization: choose an arbitrary point $\mathbf{t}^0(\mu)$ to be the first sample and let (17) be the proposal probability distribution, which suggests a candidate for the next sample value $\mathbf{t}^{k+1}(\mu)$, given the previous sample value $\mathbf{t}^k(\mu)$. (In the case the vectors $\widehat{\mathbf{c}}$ is independently generated, that is $f_{I_C}(\widehat{\mathbf{c}} \mid \mathbf{t}^k(\mu)) = f_{I_C}(\widehat{\mathbf{c}})$, the probability (17) of the candidate point is independent from the previous point.)
2. Candidate state: propose a candidate point $\mathbf{t}^{k+1}(\mu)$ from a proposal distribution (17) that may depend on the current state $\mathbf{t}^k(\mu)$. (In our case the proposal entails the solution of a LP, as the candidate primal-dual solution is obtained by solving the KKT- μ perturbed conditions with a fixed and very close to 0 value of μ and a random value of $\widehat{\mathbf{c}}$.)
3. Acceptance criterion: accept the candidate state with probability

$$\alpha(\mathbf{t}^k(\mu), \mathbf{t}^{k+1}(\mu)) = \min \left(1, \frac{f_{\mathcal{I}_{D\mu}}(\mathbf{t}^k(\mu))}{f_{\mathcal{I}_{D\mu}}(\mathbf{t}^{k+1}(\mu))} \right). \quad (28)$$

The underlying state-space of this Markov chain is the set of primal-dual solutions, associated to a fixed value μ , which are obtained by solving the KKT- μ perturbed conditions.

The correctness of the Metropolis-Hastings algorithm to simulate from the target distribution (i.e. to generate uniform primal-dual solutions) occurs under mild regularity conditions: *irreducibility* and *aperiodicity*. Broadly speaking, irreducibility means that the proposal distribution should allow to move from $\mathbf{t}(\mu)$ to $\mathbf{t}'(\mu)$ in a finite number of iterations with nonzero probability. Aperiodicity occurs when the number of steps needed to move from $\mathbf{t}(\mu)$ to $\mathbf{t}'(\mu)$ is not required to be a multiple of some integer greater than one. Consider the transition distribution of the defined Metropolis-Hastings chain

$$Q(\mathbf{t}'(\mu), \mathbf{t}(\mu)) = f_{\mathcal{I}_{D\mu}}(\mathbf{t}'(\mu)) \min \left(1, \frac{f_{\mathcal{I}_{D\mu}}(\mathbf{t}(\mu))}{f_{\mathcal{I}_{D\mu}}(\mathbf{t}'(\mu))} \right). \quad (29)$$

- Based on (29), the described properties of the proposal distribution (17) ensures that every state can be generated in one step with non-null probability (due to the positivity of the determinant in Ψ_μ).
- Since under mild conditions (29) is non-null in Ψ_μ , the chain is able to consecutively generate the same primal-dual solutions, so that all the states communicate with each other and the aperiodicity is guaranteed.

Thus, the above procedure provides primal-dual feasible points $(\widehat{\mathbf{x}}(\mu), \widehat{\mathbf{y}}(\mu), \widehat{\mathbf{z}}(\mu))$ with uniform distribution. It establishes a probabilistic framework to sample random graphs belonging to specified families (both binary and real-valued). In fact, the uniform distribution of $(\widehat{\mathbf{x}}(\mu), \widehat{\mathbf{y}}(\mu), \widehat{\mathbf{z}}(\mu))$ implies that the distribution of $\widehat{\mathbf{x}}(\mu)$ (the primal solution associated to a random graph) is also uniformly distributed, as requested.

An efficient implementation of this method is provided in the next subsection, where the LPs are solved in each step for a few randomly chosen variables, allowing a consistent reduction of the computational time, while keeping the described probabilistic properties. Such method, which we call *sequential r-blocks algorithm*, will be numerically analyzed in Section 4 and compared with a simplex-based approach (described in Subsection 3.6), which we call *sequential s-pivots algorithm*. The analyzed probabilistic properties do not hold for the *sequential s-pivots algorithm*, so that no guarantee of uniformity of the generated sample is available when network are generated by random pivots.

3.5. Sequential r -blocks algorithm

Let χ be one of the families of networks associated to extreme points of polytopes of the form $CR(\chi) = \{\mathbf{x} \in [0, 1]^{n'} : A\mathbf{x} = \mathbf{b}\}$, $A \in \mathbb{R}^{m' \times n'}$. As we noted in Subsection 3.1, given a $\mathbf{c} \in \mathbb{R}^{n'}$, we can compute a network by solving $\min_{\mathbf{y}} \mathbf{c}^T \mathbf{y}$, s.to $\mathbf{y} \in CR(\chi)$. Similarly, if we have a given extreme point \mathbf{x}^k of $CR(\chi)$, we might obtain another extreme point \mathbf{x}^{k+1} by fixing $n' - t$ variables and optimizing, with a given objective cost vector $\mathbf{c} \in \mathbb{R}^t$, the remaining t variables.

Formally, if we partition the set of variables in r blocks of dimensions t_i , $i = 1, \dots, r$, $\sum_{i=1}^r t_i = n'$, and denote by $\mathbf{x}_{F_i} \in \mathbb{R}^{n'-t_i}$ and $\mathbf{x}_{C_i} \in \mathbb{R}^{t_i}$ the fixed and *changing* components of \mathbf{x} associated to block i , and by A_{F_i} and A_{C_i} the submatrices of A associated to \mathbf{x}_{F_i} and \mathbf{x}_{C_i} , the new extreme point is obtained by solving

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{y} \\ \text{s.to} \quad & \\ & A_{C_i} \mathbf{y} = \mathbf{b} - A_{F_i} \mathbf{x}_{F_i} \\ & \mathbf{0} \leq \mathbf{y} \leq \mathbf{1} \end{aligned} \tag{30}$$

for some random vector $\mathbf{c} \in \mathbb{R}^{n'-t_i}$ and $i \in \{1, \dots, r\}$. Algorithm 1 shows how to obtain \bar{k} random networks by iteratively applying this procedure. For small r values, the r -blocks algorithm generates less dependent networks at the expense of solving from scratch many linear optimization problems. In the extreme case, for $r = 1$, the cost vectors generated are non-correlated.

Algorithm 1 r -blocks

- 1: Let $k = 0$, \mathbf{x}^0 be an initial extreme point;
 - 2: **repeat**
 - 3: Randomly select $i \in \{1, \dots, r\}$ and $\mathbf{c} \in \mathbb{R}^{t_i}$;
 - 4: Let $\mathbf{x}_{F_i}^k$ and $\mathbf{x}_{C_i}^k$ be the vectors of fixed and changing components respectively;
 - 5: Let $A_{F_i}^k$ and $A_{C_i}^k$ be the associated coefficient matrices;
 - 6: Solve (30) and let \mathbf{y}^* be its optimal solution;
 - 7: $\mathbf{x}_{C_i}^{k+1} = \mathbf{y}^*$, $\mathbf{x}_{F_i}^{k+1} = \mathbf{x}_{F_i}^k$;
 - 8: $k := k + 1$;
 - 9: **until** $k \geq \bar{k}$
-

The *sequential r -blocks algorithm* can be used as a specific implementation of the Metropolis-Hastings algorithm (as numerically done in Section 4), which accept and reject candidate solutions in accordance with (28), although the acceptance step is not explicitly reported in Algorithm 1.

3.6. Sequential s -pivots algorithm

Considering again the polytope $CR(\chi) = \{\mathbf{x} \in [0, 1]^{n'} : A\mathbf{x} = \mathbf{b}\}$, where $A \in \mathbb{R}^{m' \times n'}$, $m' < n'$, we know from LP that there is an equivalence between extreme points and basic solutions, which can be written as $\mathbf{x}^T = [\mathbf{x}_B^T, \mathbf{x}_N^T]$, $\mathbf{x}_B \in \mathbb{R}^{m'}$, $\mathbf{x}_N \in \mathbb{R}^{n'-m'}$, $A = [B \ N]$, $B \in \mathbb{R}^{m' \times m'}$, $N \in \mathbb{R}^{m' \times (n'-m')}$, for a suitable permutation of the variables. Since the extreme points of $CR(\chi)$ have been proved to be all integer, it turns out that all basic variables must be at their limits (either at 0 or 1), as well as the non-basic variables. The basic solutions are thus fully degenerate.

Denoting by e_q the q -th column vector of the identity matrix, and by B_k and N_k the basic and nonbasic submatrices of A , given a basic solution \mathbf{x}^k we can obtain another one by

moving along the simplex-like direction

$$\Delta_k(q) = \begin{bmatrix} -B_k^{-1}N_k e_q \\ e_q \end{bmatrix}. \quad (31)$$

If the nonbasic variable q is 0, then the iteration performed is $\mathbf{x}^{k+1} = \mathbf{x}^k + \lambda\Delta_k(q)$, for some non-negative step-length λ . On the other hand if $\mathbf{x}_{N_q}^k = 1$ then we apply $\mathbf{x}^{k+1} = \mathbf{x}^k - \lambda\Delta_k(q)$. It can be easily verified that in both cases $A\mathbf{x}^{k+1} = A\mathbf{x}^k = b$, i.e., the new point satisfies the linear constraints. In addition, since the constraints matrices of Section 2 are TU, the step-lengths λ —computed by a ratio test—are always either 0 or 1. It is thus possible to generate a new basic solution (i.e., a new random graph) by randomly selecting $q \in \{1, \dots, n' - m'\}$ and computing $\mathbf{x}^{k+1} = \mathbf{x}^k \pm \lambda\Delta_k(q)$. A sample of \bar{k} networks can be obtained by iteratively applying this procedure. Since the resulting sample may be claimed to be quite local, every s iterations we can jump to an independent extreme point of the polytope by generating some random cost vector, and solving the associated LP. Two drawbacks of this procedure are: (1) many iterations may be degenerate, i.e., $\lambda = 0$, so no new point is obtained; (2) the sample of networks obtained may be highly correlated if s is large because of its proximity in the feasible polytope. On the other hand, this procedure may be very efficient, since it only requires simplex pivots to obtain a new network. Algorithm 2 summarizes this procedure. Note that the s -pivots and r -blocks methods are equivalent for $s = r = 1$. They will be computationally evaluated in Section 4.

Algorithm 2 s -pivots

- 1: Let \mathbf{x}^0 be an extreme point computed with some initial cost vector \mathbf{c} ; $k = 0$;
 - 2: **repeat**
 - 3: **if** $((k + 1) \bmod s) > 0$ **then**
 - 4: **repeat**
 - 5: Randomly select $q \in \{1, \dots, n' - m'\}$;
 - 6: Compute step-length $\lambda \in \{0, 1\}$ associated with $\pm\Delta_k(q)$;
 - 7: **until** $\lambda \neq 0$
 - 8: Compute $\mathbf{x}^{k+1} = \mathbf{x}^k \pm \lambda\Delta_k(q)$ and update B_k and N_k ;
 - 9: $k = k + 1$;
 - 10: **else**
 - 11: Generate cost vector $\mathbf{c} \in \mathbb{R}^{n'}$ and compute a new extreme point \mathbf{x}^k ;
 - 12: $k = k + 1$;
 - 13: **end if**
 - 14: **until** $k \geq \bar{k}$
-

The *sequential s -pivots algorithm* is not a specific case of the described Metropolis-Hastings procedures, as a full characterization of the proposal distribution (associated to the random pivots) is not available. It turns out that the probabilistic properties, analyzed in subsections 3.1 and 3.4 do not hold, so that no guarantee of uniformity of the generated sample is available when network are generated by random pivots.

3.7. Efficient solution of the resulting LPs

The r -blocks and s -pivots procedures of subsections 3.5 and 3.6 require the repeated solution of several LP problems, at each iteration and every s iterations, respectively. Moreover, each one of these LPs may be computationally expensive for large networks. Therefore, although state-of-the-art implementations of the simplex method and polynomial time interior-point algorithms can be used, it is worth to exploit the problem structure whenever possible [9].

n	C	n. var.	n. constr.	Dual Simplex Cplex 12.5	Barrier Cplex 12.5	BlockIP
50	50	61250	100	0.7 (733)	0.2 (13)	0.2 (34)
150	50	558750	200	6.4 (934)	5.3 (26)	3.4 (43)
450	50	5051250	500	126.3 (2462)	90.8 (46)	45.7 (56)
50	150	183750	200	5.0 (1435)	0.9 (14)	0.9 (40)
150	150	1676250	300	58.3 (2503)	18.7 (30)	12.6 (52)
450	150	15153750	600	1175.5 (5243)	384.6 (53)	147.2 (58)
50	450	551250	500	5.2 (1956)	2.9 (15)	3.3 (45)
150	450	5028750	600	378.8 (4822)	64.9 (35)	37.7 (51)
450	450	45461250	900	10926.3 (13047)	1287.4 (52)	409.6 (54)

Table 1: CPU time and iterations (in parentheses) of three LP algorithms.

It can be shown that, under a proper row and column permutation, most of the constraints matrices of Section 2 exhibit a primal block-angular structure such as

$$\begin{bmatrix} N_1 & & & & \\ & N_2 & & & \\ & & \ddots & & \\ & & & N_k & \\ L_1 & L_2 & \dots & L_k & I \end{bmatrix} \begin{bmatrix} x^1 \\ x^2 \\ \vdots \\ x^k \\ x^0 \end{bmatrix} = \begin{bmatrix} b^1 \\ b^2 \\ \vdots \\ b^k \\ b^0 \end{bmatrix}. \quad (32)$$

Matrices N_i and L_i $i = 1, \dots, k$, respectively define the block-diagonal and linking constraints, k being the number of blocks. Vectors x^i , $i = 1, \dots, k$, are the variables for each block. x^0 are the slacks of the linking constraints $\sum_{i=1}^k L_i x^i \leq b^0$ ($x^0 = 0$ if linking constraints are equalities). b^i , $i = 1, \dots, k$, is the right-hand side vector for each block of constraints, whereas b^0 is for the linking constraints.

Consider, for instance, the constraints matrix of the edge-colored network (4). The number of blocks is $k = C$, the decision variables are the $n(n-1)/2$ edges of each one of the C AMs, and $N_i = \mathbf{e}^T$, $i = 1, \dots, C$. Matrices N_i are also row vectors in most of the other families of networks introduced in Section 2.

Problems with the constraints structure of (32) can be efficiently solved by specialized approaches, such as the interior-point method of [5, 6, 7] (BlockIP from now on). A description of this method is out of the scope of this work; details can be found in the above references. For instance, Table 1 reports computational results for the solution of several edge-colored network instances. Columns n and C show the number of nodes and colors of the networks. Columns “n. var.” and “n. constr.” give the number of variables and constraints of the resulting LP problems. Note that the largest case has more than 45 million variables, and 900 constraints. The remaining columns of Table 1 give the CPU time and number of iterations (in parentheses) for the three algorithms tested: Cplex 12.5 dual simplex, Cplex 12.5 barrier (interior-point), and the specialized interior-point method of BlockIP. The runs were carried out on a Fujitsu Primergy RX300 server with 3.33 GHz Intel Xeon X5680 CPUs (24 cores) and 144 GB of RAM, under a GNU/Linux operating system (Suse 11.4), without exploitation of multithreading capabilities.

From Table 1, the simplex method is clearly outperformed by the barrier algorithm, and the gap increases with the size of the instance. BlockIP, the specialized interior-point algorithm, was two to three times faster than the Cplex barrier in the largest instances, resulting to be the most efficient approach for this kind of problems.

Graph conditioned to ...	bases	CPU time
total density	1166	1.7
within group densities	135	1.6
lower bound of the degrees range and density	1267	1.7

Table 2: Number of bases visited to compute 100 extreme points with the s -pivots method for the three specified polytopes

4. Computational analysis

This section provides numerical results comparing the efficiency and correctness of the r -blocks and s -pivots procedures, both in terms of CPU time and analysis of specified network features. Three network data sets are used: a 62 node undirected graph, representing the social network of frequent associations between dolphins in a community living off Doubtful Sound, New Zealand [18, 19]; and two 39 node undirected graphs, representing alliances among workers during extended negotiations for higher wages in a tailor shop in Zambia at two different times (seven months apart) over a period of one month [17].

The runs were carried out on the same hardware used for the results of Subsection 3.7. According to these results, the LPs were solved with BlockIP, the most efficient approach for these problems. The simplex pivots required by the s -pivots procedure were implemented in Matlab.

4.1. Degeneracy in the s -pivots procedure

Consider the previously introduced data set of 62 dolphins and the following families of networks: (1) undirected networks conditioned to the density; (2) undirected networks conditioned to the within and between group densities; (3) undirected networks conditioned to the lower bound of the degree sequence range and density. These three models are specified by the observed parameters of the dolphin’s social network. The within group densities are obtained from the community structure of the observed network, computed by the *walk trap* community search algorithm of [23]. Table 2 shows the number of basic solutions explored by the s -pivots procedure to generate 100 different networks, and the required CPU time. The results confirm the high degeneracy of the s -pivots procedure.

4.2. Correctness and efficiency of r -blocks and s -pivots procedures

For the data sets of dolphins and the two of workers previously introduced, we consider the simple uniform random network conditioned to the density model. Our interest is in two network features: clustering coefficient (CC) and assortativity coefficient (AC). The CC of a network is the average CC of all the nodes, as a measure of how dense is the neighborhood of each node. The AC is the Pearson correlation coefficient of degree between pairs of linked nodes. Positive values of AC indicate a correlation between nodes of similar degree, while negative values indicate relationships between nodes of different degree.

For each of the three data sets, eight samples of 1000 networks have been obtained using the r -blocks and the s -pivots methods for different r and s values. Tables 3, 5 and 7 show, respectively for each data set, the mean and standard deviation of the CC and AC over the samples generated by the r -blocks method. Last column reports the CPU time in seconds. The Metropolis-Hastings algorithm has been used to guarantee a uniform distribution of the generated sample; the sixth column informs about the number of rejections.

r	mean CC	std. CC	mean AC	std. AC	rejected	CPU time
12	0.0819	0.0185	-0.0353	0.0774	1226	58.4
24	0.0806	0.0109	-0.0313	0.0740	1659	47.4
48	0.0796	0.0185	-0.0195	0.0618	1779	61.5
96	0.0869	0.0181	-0.0562	0.0860	1865	55.3

Table 3: Numerical results using the r -blocks method for the dolphin data set.

s	mean CC	std. CC	mean AC	std. AC	CPU time
20	0.0846	0.0207	-0.0301	0.0751	20.4
40	0.0836	0.0192	-0.0301	0.0823	18.8
80	0.0801	0.0198	-0.0468	0.0816	17.9
160	0.0759	0.0121	-0.0265	0.0659	17.4

Table 4: Numerical results using the s -pivots method for the dolphin data set.

Likewise, tables 4, 6 and 8 summarize the analogous information for networks generated with the s -pivots methods.

Theoretical results [4] state that the expected clustering coefficient of a uniform random network with n nodes conditioned to d edges is $2d/n^2$, which in the case of the dolphin social network is $2 \cdot 159/62^2 = 0.08272633$. This is approximately what we obtained in tables 3 and 4. The expected assortativity coefficient of a uniform random network with n nodes conditioned to d is approximately $-\frac{1}{n/2-1}$ (this approximation is based on the multivariate hypergeometric distribution of the degree vector), which in the case of the dolphin social network is $-\frac{1}{62/2-1} = -0.03333$, fitting reasonably well the simulated networks.

The numerical values of CC and AC obtained for the first data set of workers in tables 5 and 6 also resemble the theoretical values under the considered probabilistic model, which are $2 \cdot 158/39^2 = 0.2077581$ and $-\frac{1}{39/2-1} = -0.05405405$ respectively. Beside, the corresponding theoretical values of CC and AC for the second observed network of workers are $2 \cdot 223/39^2 = 0.2932281$ and $-\frac{1}{39/2-1} = -0.05405405$, which are consistent with the numerical results in tables 7 and 8. The observed numerical correctness of the r -blocks method supports the results of Subsection 3.1.

It is worth remarking that this kind of simulations are normally divided into pre- and post-convergence periods, where the pre-convergence part, known as *burn-in*, is discarded and the post-convergence part is used for inference. The sample means in tables 3–8 have been calculated discarding the first 200 networks, so they only include a relatively small sample size of 800 instances.

The length of the burn-in period depends on the autocorrelation of the resulting sample. The plots in figures 1–6 show the autocorrelations function of CC and AC for the 1000 networks obtained in each of the runs of tables 3–8 with $r = 12$ and $s = 20$. It can be observed from the autocorrelation function associated to different values of r and s (not plotted to

r	mean CC	std. CC	mean AC	std. AC	rejected	CPU time
12	0.2145	0.0237	-0.0481	0.0629	1006	45.6
24	0.2160	0.0216	-0.0491	0.0715	1216	40.2
48	0.2170	0.0215	-0.0511	0.0601	1452	37.4
96	0.2154	0.0207	-0.0574	0.0639	1673	37.7

Table 5: Numerical results using the r -blocks method for the first workers data set.

s	mean CC	std. CC	mean AC	std. AC	CPU time
20	0.2099	0.0185	-0.0353	0.0635	8.1
40	0.2070	0.0201	-0.0707	0.0700	7.9
80	0.2111	0.0159	-0.1054	0.0714	7.8
160	0.2211	0.0212	-0.0006	0.0474	7.8

Table 6: Numerical results using the s -pivots method for the first workers data set.

r	mean CC	std. CC	mean AC	std. AC	rejection	CPU time
12	0.3002	0.0154	-0.0531	0.0540	910	71.4
24	0.3022	0.0168	-0.0527	0.0526	1082	48.4
48	0.3006	0.0164	-0.0441	0.0572	1265	42.1
96	0.2972	0.0134	-0.0334	0.0624	1305	41.5

Table 7: Numerical results using the r -blocks method for the second workers data set.

save space) that the autocorrelations tend to zero faster when r and s are small. Nonetheless, the s -pivots methods maintains a strongly autocorrelated behavior even when $s = 20$. On the other hand, the autocorrelations of the sample obtained with the r -blocks procedure quickly tend to zero for small lags. Thus, despite the s -pivots outperforms the r -blocks method in terms of efficiency, the generation of less autocorrelated networks by the r -blocks method allows for a much smaller sample size when simulating large networks. Moreover, the availability of (15) allows a more rigorous evaluation of the probabilistic properties of the networks obtained with the r -blocks method and the application of the Metropolis-Hastings algorithm. As it can be seen from tables 3, 5 and 7, the number of Metropolis-Hastings rejections is approximately one half of the generated sample. One possibility to increase the acceptance rate is to update the probability density function of the objective costs along the iterations, increasing the internal dependencies of the described Markov Chain.

As already mentioned, the study of uniform random networks conditioned to the density by the s -pivots or r -blocks methods are quite unreasonable, since plenty of theoretical results for this model are available [4]. However this simple model allowed us to validate our procedures. For more complicated models, as the one of next subsection, these LP-based network generation procedures are instrumental.

4.3. Application to models with non-TU constraints matrices

In [18, 19] it was shown that the social network of dolphins exhibits a remarkable level of community structure, i.e., dolphins can be easily grouped into (potentially overlapping) sets such that each set of nodes is densely connected internally. Characterizing those communities is in general a difficult task. We used two of the several algorithms for community finding, the walk trap community and the fast greedy community, as implemented in the `igraph` library of the R package. The first algorithm finds densely connected subgraphs by simulating

s	mean CC	std. CC	mean AC	std. AC	CPU time
20	0.3001	0.0125	-0.0391	0.0521	8.5
40	0.3055	0.0135	-0.0578	0.0614	8.2
80	0.2993	0.0171	-0.0667	0.0518	8.1
160	0.3093	0.0101	-0.0118	0.0330	8.0

Table 8: Numerical results using the s -pivots method for the second workers data set.

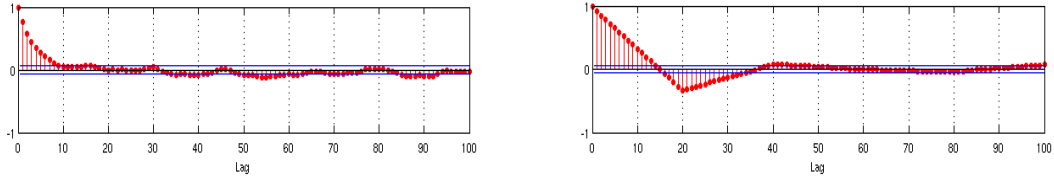


Figure 1: Autocorrelation function of CC for the r -blocks algorithm (left plot), with $r = 12$, and s -pivots algorithm (right plot), with $s = 20$, corresponding to the samples in tables 3 and 4.

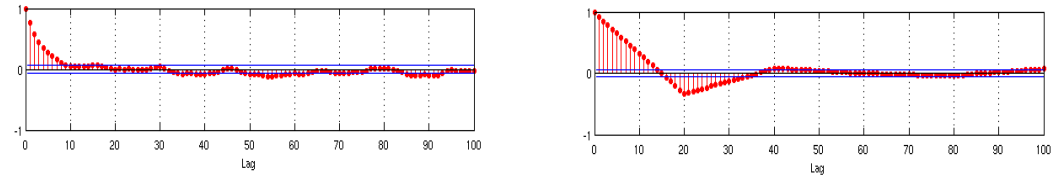


Figure 2: Autocorrelation function of AC for the r -blocks algorithm (left plot), with $r = 12$, and s -pivots algorithm (right plot), with $s = 20$, corresponding to the samples in tables 3 and 4.

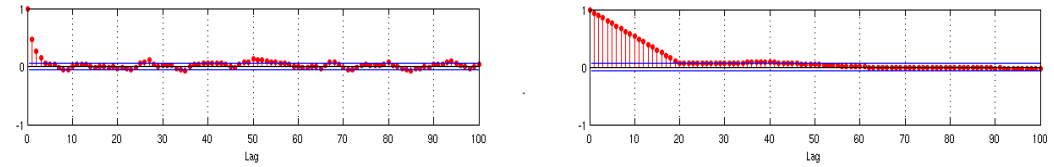


Figure 3: Autocorrelation function of CC for the r -blocks algorithm (left plot), with $r = 12$, and s -pivots algorithm (right plot), with $s = 20$, corresponding to the samples in tables 5 and 6.

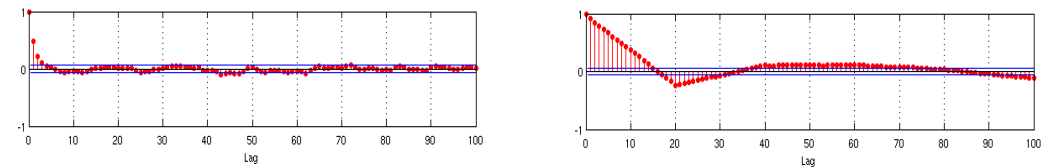


Figure 4: Autocorrelation function of AC for the r -blocks algorithm (left plot), with $r = 12$, and s -pivots algorithm (right plot), with $s = 20$, corresponding to the samples in tables 5 and 6.

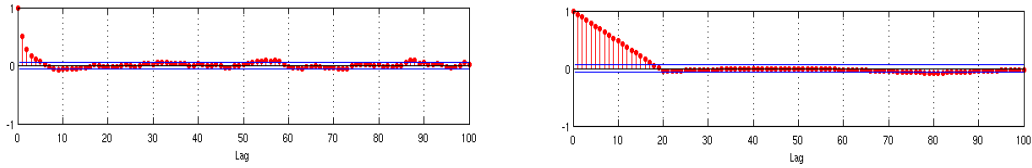


Figure 5: Autocorrelation function of CC for the r -blocks algorithm (left plot), with $r = 12$, and s -pivots algorithm (right plot), with $s = 20$, corresponding to the samples in tables 7 and 8.

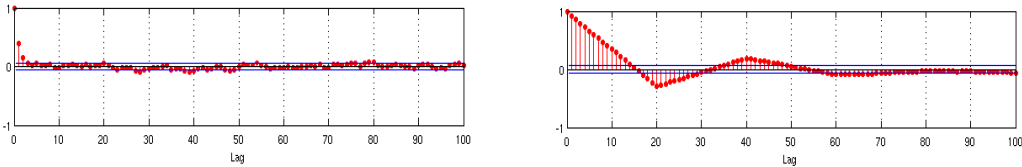


Figure 6: Autocorrelation function of AC for the r -blocks algorithm (left plot), with $r = 12$, and s -pivots algorithm (right plot), with $s = 20$, corresponding to the samples in tables 7 and 8.

random walks on the graph, which tend to stay inside communities. The second algorithm identifies (using the betweenness measure) edges in a network that lie between communities and then removes them, leaving behind just the communities themselves. Both algorithms found almost the same four communities.

Using the above four groups, the goal is to decide whether the CC and AC of our social network of dolphins is likely to have been randomly obtained from the distribution of undirected networks conditioned to the observed within-community-densities and degree nodes. As noted in Section 2 the constraints matrix of this model is not TU. However, we can still use the r -blocks procedure, discarding the fractional solutions found.

A sample of 100 networks was generated using the observed densities and degree of the dolphin’s social network. Table 9 provides the numerical results for $r \in \{4, 5, 6, 8, 10, 13\}$. Column “Fractional networks” shows the number of rejected fractional solutions. If two consecutive LPs with different objectives provided the same solution (named “loops”), the repeated network is also rejected; they are reported in column “Loops” of Table 9. Last column provide the CPU time in seconds, using the same computational environment than in previous sections.

In accordance with the results in Table 9, the number of fractional networks seems to decrease with the number of blocks r . Nonetheless, when r is large, consecutive networks are

r	Fractional networks	Loops	CPU time
4	44	0	112.0
5	38	1	103.2
6	26	5	89.1
8	10	16	61.4
10	8	110	68.7
13	3	372	74.6

Table 9: Numerical results of the sampling procedure using the sequential r -blocks optimization method.

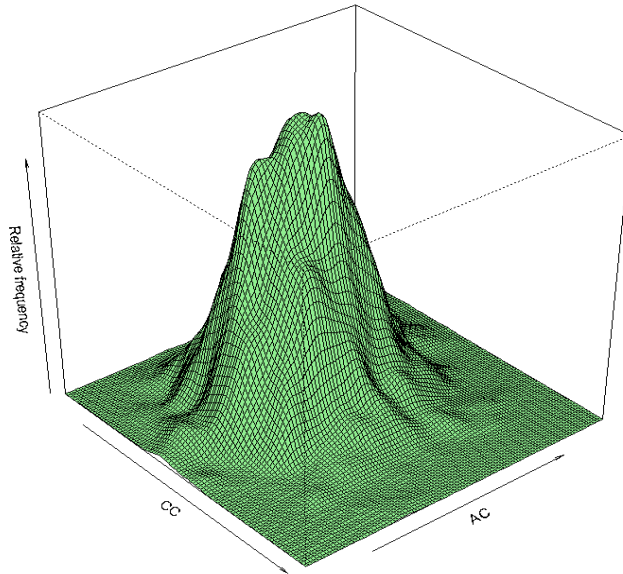


Figure 7: Empirical distribution of CC and AC obtained with the 8-blocks procedure

	sample mean	sample std.	observed value	one tail p-value
CC	0.199145	0.01949775	0.2901595	0.0112000
AC	-0.065980	0.07585131	-0.0436017	0.6941203

Table 10: Numerical results from the sample obtained with the 8-blocks method

more likely to be the same, so that we face a trade-off between minimizing the number of loops and minimizing the number of fractional solutions. In any case, the r -blocks procedure was still efficient for this non-TU model.

Based on the results in Table 9, we set $r = 8$ (which guarantees an uncorrelated sample with low number of loops and fractional solutions) and generated 10100 networks, removing the first 100 ones as burn-in period.

The resulting empirical distribution of CC and AC is shown in the density plot of Figure 7. Table 10 shows the sample mean and standard deviation of the CC and AC over the 10000 generated networks, the observed values in the dolphins social network, and the associated p-values. From these p-values we conclude that the CC cannot be explained by only using the information concerning the within-community-density and the degree nodes. On the other hand, the AC seems instead to be likely induced by the fixed structural properties we considered. The LP-based sampling procedures developed in this work allowed to easily perform this kind of inference analysis, which otherwise could not have been done with the available theoretical results for these complex network models.

5. Conclusions

The constraints matrices associated to several classes of random graph problems have shown to be TU. The resulting MILP problems are thus efficiently solved as LPs. It was observed that specialized interior-point algorithms outperform state-of-the-art simplex solvers for this kind of problems. Two particular procedures, r -blocks and s -pivots, were introduced to generate large samples of random graphs. These two procedures were validated both empirically, using three real-world data sets, and theoretically, obtaining expressions for

the density function and entropy of the LP solutions as a function of the distribution and entropy of the cost vectors. Extending this approach to other classes of random graph problems, and showing whether their constraints matrices are TU, is part of the further work to be done. The application of the acceptance-rejection Metropolis-Hastings method [27] to obtain LP solutions guaranteeing other than the uniform particular distributions should also be explored in the future. Indeed, the observed numerical correctness of the r -blocks method suggests a particularly low rejection ratio in the Metropolis-Hastings algorithm.

This work showed how mathematical programming tools can be efficiently used for the analysis of complex networks, and it opens the possibility for other applications in this field.

Acknowledgments

This work has been supported by grants MTM2012-31440 of the Spanish research program. Stefano Nasini has been supported by a FPI-UPC grant. The authors also want to thank the anonymous referees whose comments greatly improved the presentation of the paper.

References

- [1] Abraham, R., Robbin, J. (1967), *Transversal mappings and flows*, W. A. Benjamin, Inc., New York-Amsterdam.
- [2] Ahuja, R.K., Magnanti, T.L., Orlin, J.B., (1991), *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, Inc., Englewood Cliffs.
- [3] Berghammer, R., Rusinowska A., de Swart, H., (2010), Applying relation algebra and RelView to measures in a social network, *European Journal of Operational Research*, 202, 182–195.
- [4] Bollobas, B., (1985), *Random Graphs*, Cambridge University Press.
- [5] Castro, J., (2000), A specialized interior-point algorithm for multicommodity network flows, *SIAM Journal on Optimization*, 10, 852–877.
- [6] Castro, J., (2007), An interior-point approach for primal block-angular problems, *Computational Optimization and Applications*, 36, 195–219.
- [7] Castro, J., Cuesta, J., (2011), Quadratic regularizations in an interior-point method for primal block-angular problems, *Mathematical Programming*, 130, 415–445.
- [8] Charon, I., Germa, A., Hudry, O., (1996), Random generation of tournaments and asymmetric graphs with given out-degrees, *European Journal of Operational Research*, 95, 411–419.
- [9] Conejo, A.J., Castillo, E., Minguez, R., Garcia-Bertrand, R., (2006), *Decomposition Techniques in Mathematical Programming. Engineering and Science Applications*, Springer.
- [10] Cover, T.M., Thomas, J.A., (1991), *Elements of Information Theory*, Wiley.
- [11] Erdős, P., Rainyi, A., (1959), On random graphs, *Publicationes Mathematicae*, 6, 290–297.
- [12] Fremlin, D.H. (2010), *Measure Theory, Volume 2, 2nd. Ed.*, Torres Fremlin.

- [13] Gómez, D., Figueira, J.R., Eusébio, A., (2013), Modeling centrality measures in social network analysis using bi-criteria network flow optimization problems, *European Journal of Operational Research*, 95, 354–365.
- [14] Granovetter, M., (1985), Economic action and social structure: The problem of embeddedness, *American Journal of Sociology*, 91, 481–510.
- [15] 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, 247–252.
- [16] Holland, P.W., Leinhardt, S., (1976), Local structure in social networks, *In D. Heise (Ed.), Sociological Methodology*, 1–45.
- [17] Kapferer, B., (1972), *Strategy and transaction in an African factory*, Manchester University Press.
- [18] Lusseau D., (2003), The emergent properties of a dolphin social network, *Proceedings of the Royal Society of London Series B-Biological Sciences*, 270, 186–188.
- [19] Lusseau D., (2004), Identifying the role that individual animals play in their social network, *Proceedings of the Royal Society of London Series B-Biological Sciences*, 271, 477–481.
- [20] Newman, M.E.J., Strogatz, S.H., Watts, D.J., (2001). Random graphs with arbitrary degree distributions and their applications, *Phys. Rev. E*, 64, 026118.
- [21] Newman, M.E.J., (2002). Random graphs as models of networks, *Working Papers 02-04-020, Santa Fe Institute*.
- [22] Newman, M.E.J., Park, J., (2003). Why social networks are different from other types of networks, *Phys. Rev. E*, 68, 036122.
- [23] Newman, M.E.J., (2004), Detecting community structure in networks, *The European Physical Journal B*, 38, 321–330.
- [24] Padberg, M., *Linear optimization and extensions. 2nd edition*, Springer, New York, 1999.
- [25] Pevzner, A., (1995), DNA physical mapping and alternating Eulerian cycles in colored graphs, *Algorithmica*, 13, 77–105.
- [26] Rao, A., Jana, R., Bandyopadhyay, S., (1996), A Markov chain Monte Carlo method for generating random (0,1) matrices with given marginals, *Sankhya, ser. A*, 58, 225–242.
- [27] Robert, C., Casella, G., (2004), *Monte Carlo Statistical Methods*. Springer.
- [28] Roberts, J.M., (2000), Simple methods for simulating sociomatrices with given marginal totals, *Social Networks*, 22, 273–283.
- [29] Ryser, H.J., (1957), Combinatorial properties of matrices of zeros and ones, *Canadian Journal of Mathematics*, 9, 371–377.
- [30] Schrijver, A., (1998), *Theory of linear and integer programming*, Wiley.
- [31] Snijders, T.A.B., (1991), Enumeration and simulation methods for 0-1 matrices with given marginals, *Psychometrika*, 56, 397–417.

- [32] Sternberg, S., (1964), *Lectures on differential geometry*, Englewood Cliffs, NJ, Prentice-Hall.
- [33] Verhelst, N.D., (2008), An efficient MCMC algorithm to sample binary matrices with fixed marginals, *Psychometrika*, 73(4), 705–728.
- [34] Wasserman, S., Faust, K., *Social Network Analysis: Methods and Applications*, Cambridge University Press, 1994.
- [35] [Wright 1996] Wright, S.J., (1996), *Primal-dual interior-point methods*, SIAM.
- [36] Ye, Y. (1997), *Interior Point Algorithms. Theory and Analysis*, Wiley.