

PAPER • OPEN ACCESS

## A faster horse on a safer trail: generalized inference for the efficient reconstruction of weighted networks

To cite this article: Federica Parisi *et al* 2020 *New J. Phys.* **22** 053053

View the [article online](#) for updates and enhancements.

**PAPER**

# A faster horse on a safer trail: generalized inference for the efficient reconstruction of weighted networks

**OPEN ACCESS****RECEIVED**

18 October 2019

**REVISED**

27 January 2020

**ACCEPTED FOR PUBLICATION**



10 February 2020

**PUBLISHED**

27 May 2020

Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](#).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

Federica Parisi<sup>1</sup>, Tiziano Squartini<sup>1</sup>  and Diego Garlaschelli<sup>1,2</sup> <sup>1</sup> IMT School for Advanced Studies Lucca, Italy<sup>2</sup> Lorentz Institute for Theoretical Physics, Leiden University, The NetherlandsE-mail: [tiziano.squartini@imtlucca.it](mailto:tiziano.squartini@imtlucca.it)**Keywords:** complex networks, network reconstruction, entropy maximization**Abstract**

Due to the interconnectedness of financial entities, estimating certain key properties of a complex financial system, including the implied level of systemic risk, requires detailed information about the structure of the underlying network of dependencies. However, since data about financial linkages are typically subject to confidentiality, network reconstruction techniques become necessary to infer both the presence of connections and their intensity. Recently, several ‘horse races’ have been conducted to compare the performance of the available financial network reconstruction methods. These comparisons were based on arbitrarily chosen metrics of similarity between the real network and its reconstructed versions. Here we establish a generalized maximum-likelihood approach to rigorously define and compare weighted reconstruction methods. Our generalization uses the maximization of a certain conditional entropy to solve the problem represented by the fact that the density-dependent constraints required to reliably reconstruct the network are typically unobserved and, therefore, cannot enter directly, as sufficient statistics, in the likelihood function. The resulting approach admits as input any reconstruction method for the purely binary topology and, conditionally on the latter, exploits the available partial information to infer link weights. We find that the most reliable method is obtained by ‘dressing’ the best-performing binary method with an exponential distribution of link weights having a properly density-corrected and link-specific mean value and propose two safe (i.e. unbiased in the sense of maximum conditional entropy) variants of it. While the one named CREM<sub>A</sub> is perfectly general (as a particular case, it can place optimal weights on a network if the bare topology is known), the one named CREM<sub>B</sub> is recommended both in case of full uncertainty about the network topology and if the existence of some links is certain. In these cases, the CREM<sub>B</sub> is faster and reproduces empirical networks with highest generalized likelihood among the considered competing models.

**1. Introduction**

Network reconstruction is an active field of research within the broader field of complex networks. In general, network reconstruction consists in facing the double challenge of inferring both the bare topology (i.e. the existence or absence of links) and the magnitude (i.e. the weight) of the existing links of a network for which only aggregate or partial structural information is known. These two pieces of the puzzle (i.e. the ‘topology’ and the ‘weights’) represent equally important targets of the reconstruction problem, although reaching those targets may require very different strategies. In general, the available pieces of information represent the constraints guiding the entire reconstruction procedure. Depending on the nature of the available constraints, different reconstruction scenarios materialize. The scenario considered in this paper is the one that is recurrently encountered in the study of financial and economic networks [1, 2].

Indeed, financial networks are a class of networks for which the reconstruction challenge is particularly important. The estimation of systemic risk, the simulation of financial contagion and the ‘stress testing’ of a financial network in principle require the complete knowledge of the underlying network structure. If the

description of this structure is naively simplified or reduced, then the outcome of those stress tests becomes unreliable when taken as a proxy of what would happen on the real network in the same situation. This may imply a severe underestimation of the level of systemic risk, as research conducted in the aftermath of the 2007–2008 crisis showed. In the typical situation for financial networks, the total number  $N$  of nodes (e.g. the number of banks in a given network of interbank lending) is known, but the number, intensity and position of the links among those nodes is unknown because of confidentiality issues. Generally, one has access only to node-specific information that is publicly available. For instance, from publicly reported balance sheets one knows the so-called ‘total assets’ (total value of what a bank owns, including what it lent out to other banks in the network) and ‘total liabilities’ (total value of what a bank owes to the external world, including what it borrowed from other banks in the network) of each bank in an interbank system. Similar considerations apply to inter-firm networks, where links represent typically unobservable individual transactions while the total purchases and total sales of each of the firms in the system considered are more easily accessible. One more example, which is relevant not strictly for the reconstruction problem but rather from a modeling point of view, is that of the international trade network, where one would like to obtain a good model of international trade flows from country-specific aggregate quantities such as the total imports and total exports of each country.

In all the examples mentioned above, the pieces of node-specific information typically represent a good proxy of the *margins*, i.e. the sums along columns and rows, of the weighted adjacency matrix  $\mathbf{W}^*$  of the (in general directed) network, whose entry  $w_{ij}^*$  quantifies the magnitude of the link existing from node  $i$  to node  $j$  (including  $w_{ij}^* = 0$  if no link is there). In the language of network science, these two margins are called the *out-strength*  $s_i^{\text{out}*} \equiv \sum_{j \neq i} w_{ij}^*$  and the *in-strength*  $s_i^{\text{in}*} \equiv \sum_{j \neq i} w_{ji}^*$  of node  $i$ , where the asterisk indicates the ‘true’ value, i.e. the value measured on the true network  $\mathbf{W}^*$ , of those quantities. In general, one assumes that the full matrix  $\mathbf{W}^*$  itself is unobservable, while  $s_i^{\text{out}*}$  and  $s_i^{\text{in}*}$  are (directly or indirectly) measurable for each node ( $i = 1 \dots N$ ). The  $N$ -dimensional vectors  $\vec{s}^{\text{out}*}$  and  $\vec{s}^{\text{in}*}$  constructed from all node strengths are called the *out-strength sequence* and *in-strength sequence*, respectively. It is worth stressing here that the in- and out-strength sequences represent a form of *weighted constraints* that can be imposed in the reconstruction procedure, because they depend explicitly on the magnitude of the links in the network. As such, they do not contain direct information about the *binary topology* of the network, such as the overall density of links, the number of links (i.e. the *degree*) of each node, etc. This makes the simultaneous inference of both the link weights *and* the bare topology of the network particularly challenging in this setting.

Irrespective of how the strength sequences are used in the reconstruction method, it is clear that there are multiple (in fact, hugely many) possible networks that are consistent with such margins. The essence of each method lies in how this set of compatible networks is further restricted to produce the output networks. At one extreme, there are greedy methods based on certain heuristics or ansatz that in the end produce a single possible instance of the network. The problem with these ‘deterministic’ methods is that, by producing a single outcome, they give zero probability to any other network, including (apart from sheer luck) the true unobserved network  $\mathbf{W}^*$ . This implies that the likelihood of producing the real network given the model is always zero. The success of such deterministic methods, as well as their comparison with competing methods, has therefore to be assessed via some arbitrarily chosen metric of network similarity. At the other extreme, there are maximally agnostic methods designed to impose absolutely no other ansatz or heuristic besides the knowledge of the strengths sequences, so that all the compatible networks are accepted with equal probability. This is the class of (unconditional) *maximum-entropy methods*, that look for the probability distribution (in the space of weighted networks) maximizing the entropy, subject to the imposed constraints. Research has shown that typical networks sampled from maximum-entropy ensembles of networks with given strength sequence are fully or almost fully connected [3]. In light of the sparsity of most real-world networks, this is a major limitation.

All the state-of-the-art reconstruction methods are found somewhere in between the two extreme cases described above. Among the methods proposed so far, some assume that the constraints concerning the binary and the weighted network structure jointly determine the reconstruction output. An example providing an excellent reconstruction of several real-world weighted networks is the enhanced configuration model (ECM) [3], defined by simultaneously constraining the nodes degrees and strengths. However the inaccessibility of empirical degrees makes this method inapplicable in the setting considered here. This has led to the introduction of two-step algorithms [4, 5] that perform a preliminary estimation of node degrees to overcome the lack of binary information. Other methods consider the weights estimation step as completely unrelated to the binary one [6, 7]. Examples include methods that adjust the link weights iteratively on top of some previously determined topological structure (e.g. via the recipe firstly proposed in [8]), in such a way to satisfy the strengths constraints *a posteriori*. This kind of procedure, however, assigns weights deterministically, thus being unable to provide confidence bounds accompanying the weight estimates [9] and basically giving zero probability to any real-world network.

In this paper, after reviewing the state-of-the-art methods and discussing their performance, we develop a theoretical framework that provides an analytical, unbiased<sup>3</sup> (i.e. based on the maximization of a certain conditional entropy) procedure to estimate the weighted structure of a network. The maximization of the conditional entropy generalizes the exponential random graph (ERG) formalism to situations where the aggregate topological properties that effectively summarize the network topology are not directly observable and cannot therefore enter as sufficient statistics into the model (and in the ordinary likelihood function). Information about the topological structure (either available *ab initio* or obtained by using any of the existing algorithms for the purely binary reconstruction) is treated as *prior* information. Together with the available weighted constraints, this prior information represents the input of our generalized reconstruction procedure. The probability distribution describing link weights is, then, determined by maximizing a suitably defined conditional entropy. This construction allows us to achieve an optimal compromise between the deterministic and fully agnostic extremes mentioned above: while we allow the method to incorporate a certain ansatz (both for the purely binary structure and for the weights) that effectively restricts the set of compatible networks, we still maximize a certain entropy in order to preserve the necessary indifference among configurations that have the same ‘good’ properties, induced by the ansatz itself. Finally, the parameters of the conditionally maximum-entropy distribution are found by maximizing a generalized likelihood function that depends on the probability distribution over binary graphs implied by the binary reconstruction method. This last step makes the weight distribution dependent, as it should, on the purely binary expected network properties.

As it turns out, when link weights are treated as continuous random variables, their distribution—conditional on the existence of the links themselves—is exponential, a result that can be used to further enhance the performance of the best-performing methods available to date, providing them with a recipe to determine confidence intervals for the weight estimates. While it is a well know result that the exponential distribution follows from the maximization of the entropy subject to a constraint on the mean value, what is nontrivial here is the determination of how the mean value itself should depend on a combination of certain empirically observed regularities and, crucially, on the prior probability distribution of the bare topological projection of the network implied by the binary reconstruction method chosen as input. As a byproduct, our generalized reconstruction framework leads to a computationally simpler variant of our method, based on the solution of a single nonlinear equation in place of several coupled, nonlinear equations as in some of the previous methods<sup>4</sup>.

The rest of the paper is organized as follows. In section 2 we review the state of the art of network reconstruction methods and discuss their performance. We then describe our generalized ‘conditional reconstruction method’ in detail, providing two different specifications of it. In section 3 we test the performance of the method on real-world networks. In section 4 we discuss the results.

## 2. Methods

In what follows, we indicate a weighted adjacency matrix as  $\mathbf{W}$  and its generic entry as  $w_{ij}$ . Analogously, we indicate the corresponding adjacency matrix as  $\mathbf{A}$  and its entry as  $a_{ij} = \Theta(w_{ij})$ , with  $\Theta(x)$  representing the Heaviside step function, defined as  $\Theta(x) = 1$  if  $x > 0$  and  $\Theta(x) = 0$  if  $x \leq 0$ .

### 2.1. Network reconstruction methods: an overview of the state-of-the-art

*The MaxEnt method: deterministic link weights on a complete graph.* A traditional approach to network reconstruction is the so-called MaxEnt method [10–12], defined by the maximization of the ‘entropy’  $S(\mathbf{W}) = -\sum_{i,j} w_{ij} \ln w_{ij}$  under the constraints represented by the network weighted marginals, i.e.

$s_i^{\text{out}*} = \sum_{j \neq i} w_{ij}^*$ ,  $\forall i$  and  $s_i^{\text{in}*} = \sum_{j \neq i} w_{ji}^*$ ,  $\forall i$ . The resulting ‘maximum-entropy’ expression for  $w_{ij}$  is easily found to be

$$\hat{w}_{ij}^{\text{ME}} = \frac{s_i^{\text{out}*} s_j^{\text{in}*}}{W^*}, \quad \forall i, j \quad (2.1)$$

with  $W^* = \sum_i s_i^{\text{out}*} = \sum_i s_i^{\text{in}*}$ . The major drawback of the above model is the prediction of a fully connected network with all positive link weights given by equation (2.1). Yet, the above expression often provides an accurate estimation of the subset of realized (i.e. positive) real-world link weights. This fact will turn out useful

<sup>3</sup> Throughout the paper, we use the term ‘unbiased’ as intended in the application of the maximum-entropy principle, i.e. when we refer to outcomes that maximize the (conditional) entropy, so that the resulting probability distribution does not make arbitrary preferences (corresponding to hidden or unjustified assumptions) among configurations that share the same values of certain target quantity. Constrained maximum-entropy distributions produce maximally random outcomes given what is supported by the data used as constraints, thereby ensuring unbiasedness. To avoid confusion with the meaning of the term ‘bias’ in statistics, we do not use the term in the sense of ‘biased parameter estimation’.

<sup>4</sup> Where appropriate, the interested reader will be redirected towards freely available codes to run all variants of our framework.

**Table 1.** Overview of the reconstruction methods reviewed in [1]. The letter ‘P’ indicates that the considered estimation step is probabilistic in nature while the letter ‘D’ indicates that it is deterministic. The log-likelihood is defined as in equation (2.2), i.e.  $\ln Q(\mathbf{W}^*)$ .

Method	Topology	Weights	Log-likelihood
MaxEnt (ME) [10, 11]	D	D	$-\infty$
Minimum-density (MD) [13]	D	D	$-\infty$
Copula approach [14]	P	D (IPF)	$-\infty$
Drehmann and Tarashev [15]	P	D (IPF)	$-\infty$
Montagna and Lux [16]	P	D (IPF)	$-\infty$
Mastromatteo et al [17]	P	D (IPF)	$-\infty$
Gandy and Veraart [9]	P	D (MF)	$-\infty$
dcGM [5]	P	D (ME)	$-\infty$
MECAPM [18]	P	$P(w_{ij} \in \mathbb{N})$	$-\infty$
Fitness-induced DECM [4]	P	$P(w_{ij} \in \mathbb{N})$	$-\infty$
Halaj and Kok [7]	P	P	$\in \mathbb{R}$
Moussa [19]	P	P	$\in \mathbb{R}$

later in our analysis. At a fundamental level, the ultimate issue with this method is that, although the quantity  $S(\mathbf{W})$  is referred to as ‘entropy’, actually the link weight  $w_{ij}$  admits no natural interpretation as a probability distribution over the entries of the matrix, contrary to what the definition of entropy would instead require. In particular,  $S(\mathbf{W})$  is a function of a *single* matrix, rather than a function  $S(\mathcal{W})$  of a *probability distribution* over an ensemble of realizations of the matrix, treated as a random variable  $\mathcal{W}$  that can take  $\mathbf{W}$  as one of its possible values with a certain probability (the approach that we introduce later will be based precisely on a proper entropy  $S(\mathcal{W})$  of this type, and particularly on a certain conditional version of it). This consideration immediately questions the interpretation of equation (2.1) as a truly ‘maximum-entropy’ result. In fact, by producing a single matrix as output, the method is actually a deterministic (i.e. a zero-entropy) one, rather than a probabilistic one as proper maximum-entropy methods necessarily are.

*Iterative proportional fitting: deterministic link weights on any graph.* The search for nontrivial (i.e. sparser) topological configurations still guaranteeing that the weighted marginals are satisfied has led to a plethora of reconstruction methods. These models are described below; here we mention an aspect common to many of them. Irrespective of the method used for the reconstruction of the binary topology, a popular way to assign link weights on a non-complete (not fully connected) graph, while still matching the constraints given by the in- and out-strength sequences, is the iterative proportional fitting (IPF) algorithm [8]. The IPF recipe assumes that the network topology is given and iteratively ‘adjusts’ link weights until the constraints are satisfied [1, 8]. In the special case when the network is fully connected, the IPF algorithm reduces to MaxEnt. Since the IPF algorithm always yields a (unique) matrix satisfying the weighted marginals irrespective of the topological details<sup>5</sup> of the given underlying binary structure  $\mathbf{A}$ , many researchers have focused on methods for improving the reconstruction of the bare network topology, while considering the ‘link weight’ problem virtually solved and, more importantly, decoupled from the ‘topology’ problem. As we argue later on, this consideration is incorrect. Moreover, the IPF recipe suffers from two serious drawbacks, both imputable to the deterministic rule used to assign weights to a given binary configuration. First, it cannot provide confidence bounds accompanying the weight estimates. Second, the probability of reproducing any real-world weighted network is virtually zero, even if the bare topology were known exactly.

*Many horses and many races.* Here we succinctly describe the state-of-the-art reconstruction methods (the ‘horses’) that have been recently considered in various ‘horse races’ comparing the performance of different methods over a number of real-world networks. These methods have been recently reviewed in [1] and are here compactly collected in table 1. In order to unambiguously assess the performance of a given method, we consider the probability (density)  $Q(\mathbf{W})$  of generating a given weighted graph  $\mathbf{W}$  according to the method, and use the corresponding log-likelihood

$$\begin{aligned} \ln Q(\mathbf{W}^*) &= \ln[P(\mathbf{A}^*)Q(\mathbf{W}^*|\mathbf{A}^*)] \\ &= \ln P(\mathbf{A}^*) + \ln Q(\mathbf{W}^*|\mathbf{A}^*) \end{aligned} \quad (2.2)$$

as a score function quantifying how likely the structure of the specific real-world network  $\mathbf{W}^*$  is reproduced by a given algorithm. Notice that we have written  $Q(\mathbf{W}) = P(\mathbf{A})Q(\mathbf{W}|\mathbf{A})$  where  $P(\mathbf{A})$  is the probability of generating

<sup>5</sup> The only request about  $\mathbf{A}$  is its irreducibility [1, 8].

the bare topology  $\mathbf{A}$  of  $\mathbf{W}$  and  $Q(\mathbf{W}|\mathbf{A})$  is the conditional probability of generating the weights of the network, given its topology. Therefore  $P(\mathbf{A}^*)$  is a sort of purely binary likelihood.

Upon looking at table 1, several classes of algorithms can be distinguished. A first group gathers the algorithms whose estimation steps are both deterministic (notice that the purely deterministic version of the minimum-density algorithm is considered here, i.e. the one used in the ‘horse race’ [20]). Since the deterministic nature of an algorithm implies that the probability  $f_{ij}$  that nodes  $i$  and  $j$  are connected is  $f_{ij} \in \{0, 1\}$ , the only way to reproduce the actual topological structure  $\mathbf{A}^*$  is implementing the rule  $f_{ij} = 1 \iff a_{ij}^* = 1$  and  $f_{ij} = 0 \iff a_{ij}^* = 0$ . However, this prescription is viable only if the actual configuration is given, otherwise the probability of reproducing its structure is  $P(\mathbf{A}^*) = 0$ , further implying that  $\ln Q(\mathbf{W}^*) = -\infty$ .

A second group gathers algorithms where the topological estimation step is indeed probabilistic while the recipe for assigning link weights is deterministic: while the vast majority of such methods rests upon the IPF algorithm, the *density-corrected Gravity Model (dcGM)* [5] method rests upon the MaxEnt prescription. The method proposed in [9], instead, employs the *maximum-flow (MF)* algorithm to adjust weights. Even if these algorithms indeed allow for the observed topological structure to be replicable (i.e.  $P(\mathbf{A}^*) > 0$ ), they still assign weights in a deterministic fashion: this implies that the actual configuration  $\mathbf{W}^*$  can be reproduced if and only if  $\hat{w}_{ij} = w_{ij}^*$ , i.e. only in case the actual configuration is given, otherwise  $Q(\mathbf{W}^*|\mathbf{A}^*) = 0$ , again implying  $\ln Q(\mathbf{W}^*) = -\infty$ .

A third group gathers algorithms whose steps are both probabilistic. However, weights are assumed to be natural numbers: hence, configurations with *real* weights—i.e. typical real-world networks—cannot, by definition, be reproduced by such recipes.

The last two methods may, in principle, lead to recover the structure of a network with real-valued weights. However, the method by Hałaj and Kok induces a completely random topological structure, leading to a probability of reproducing any observed  $\mathbf{A}^*$  that reads  $P(\mathbf{A}^*) = 2^{-N(N-1)}$ , rapidly vanishing as  $N$  grows. On the other hand, the method proposed by Moussa aims at reproducing a specific feature of several real-world networks, i.e. a power-law degree distribution: as a consequence, it is optimized to reconstruct such a peculiar topological feature and does not allow for generic degree distributions. Moreover, the method does not come with a recipe for assigning the generated degrees to the different nodes.

*A good horse on the binary trail: the dcGM.* The above considerations imply that, as far as the simultaneous reconstruction of both the topology and the weights is concerned, none of the current methods is satisfactory. The remainder of the paper aims at introducing a viable and efficient method. The method will be designed in such a way that any purely binary reconstruction method, i.e. any  $P(\mathbf{A})$ , can be taken as input, while aiming at placing link weights optimally. This will allow us to freely choose the binary method at the end. It is therefore worthwhile to describe in some detail here the specific binary method that we will eventually select for our analyses when putting the full method at work. Our choice is guided by the results of four independent tests (‘horse races’) [20–23], that have found that, as far as the imputation of the overall binary topology of the network is concerned, the dcGM [5] systematically outperforms competing methods. Quoting from the source references:

- ‘in presenting our results we face the challenge that some algorithms produce an ensemble of networks while others produce a single matrix. This makes a straightforward comparison difficult. Fortunately, the Cimi method is the clear winner between the ensemble methods’ [20] (note: ‘Cimi’ is the name given in [20] to the dcGM);
- ‘according to our analysis, reconstructing via fitness model outperforms the other methods when the same input information is used’ [21] (note: ‘fitness model’ is the name given in [21] to the dcGM);
- ‘second, concerning the individual performance of each null model, we find that CM1, followed by CM2 and MaxEntropy, has the closest behavior to the actual network overall. Since CM2 requires much less information than CM1, we find that this makes CM2 more appealing for practical purposes’ [22] (note: ‘CM2’ is the name given in [22] to the dcGM);
- ‘as an ‘off the shelf’ model in situations without exogenous information available, the density-corrected gravity model (DC-GRAVITY) can be recommended because it is found to work well on the big sparse network as well as on the small dense network with respect to the edge probabilities and the edge values [...] Similarly, Gandy and Veraart (2019) report that this model is performing very well in binary and valued reconstruction. Further, the model can be extended towards the inclusion of exogenous information in a simple way’ [23] (note: ‘DC-GRAVITY’ is the name given in [22] to the dcGM).

The above results prompt us to select the binary part of the dcGM as the best candidate to be given as input to the full method to be developed. The dcGM is defined by the simple Bernoulli prescription



$$a_{ij} = \begin{cases} 1 & \text{with } p_{ij}^{\text{dcGM}} = \frac{z s_i^{\text{out}} s_j^{\text{in}}}{1 + z s_i^{\text{out}} s_j^{\text{in}}} \\ 0 & \text{with } 1 - p_{ij}^{\text{dcGM}} \end{cases} \quad (2.3)$$

(for  $i \neq j$ ), where the only free parameter  $z$  is tuned to reproduce the actual link density of the network [5]. It is worth mentioning here that the dcGM takes the functional form of the connection probability from the binary configuration model (BCM), i.e. the maximum-entropy model of binary graphs with given in- and out-degrees for all nodes (see [appendix](#)). In the BCM, the parameters are Lagrange multipliers that control the in- and out-degrees. In the dcGM, these parameters (that are unidentifiable, given the inaccessibility of the degrees) are replaced by the observed values of the in- and out-strengths, respectively, up to the global proportionality constant  $z$ . This is the so-called ‘fitness ansatz’, motivated by an empirical regularity showing the systematic approximate proportionality between empirical strengths and Lagrange multipliers coupled to the degrees [5]. More details are provided in the [appendix](#).

## 2.2. A framework for conditional reconstruction

In the following, we aim at introducing a method overcoming the drawbacks affecting current weight imputation recipes. Ideally, our recipe should satisfy the following requirements:

- allowing for any probability distribution (over purely binary graphs) to be acceptable as input for the preliminary topology reconstruction step (this requirement allows us to take any binary reconstruction method as input—and clearly, to select a good one for practical purposes);
- allowing for the generation of continuous weights (this requirement implies that the real unobserved network will be generated with positive likelihood);
- satisfying the constraints that are usually imposed by the availability of limited information (i.e. the out- and in-strength sequences  $\{s_i^{\text{out}}\}_{i=1}^N$  and  $\{s_i^{\text{in}}\}_{i=1}^N$ ).

As anticipated in the Introduction, these three postulates will be addressed by proposing a probabilistic reconstruction method *conditional on some prior binary information* and constrained to reproduce the aforementioned, weighted observables. In order to do so, we build upon the formalism proposed by the authors of [24] who define a fully probabilistic procedure to *separately* constrain binary and weighted network properties. In short, they introduced the continuous version of the ECM [3] and replaced the resulting probability of the binary projection of the network with the one coming from the undirected binary configuration model [25]. In such a way, the estimation of the probability coefficients controlling for the presence of links is disentangled from the estimation step concerning link weights. Unfortunately, the framework proposed in [24] cannot be directly used for network reconstruction as the information about the degrees of nodes is practically never accessible.

## 3. Results

Before entering into the details of our method, let us briefly describe the formalism we adopt. In what follows, we assume that  $\mathbf{A} \in \mathbb{A}$  is a realization of the random variable  $\mathcal{A}$ ; analogously, the weighted adjacency matrix  $\mathbf{W} \in \mathbb{W}$  instantiates the random variable  $\mathcal{W}$ . The probability mass function of the event  $\mathcal{A} = \mathbf{A}$  is denoted with  $P(\mathbf{A})$ , while  $Q(\mathbf{W}|\mathbf{A})$  is a conditional probability density function, for the variable  $\mathcal{W}$  taking the value  $\mathbf{W}$ , given the event  $\mathcal{A} = \mathbf{A}$ . Notice that we are considering continuous (non-negative and real-valued) weights and that  $Q(\mathbf{W}|\mathbf{A})$  is non-zero only over the continuous set  $\mathbb{W}_{\mathbf{A}} = \{\mathbf{W}: \Theta(\mathbf{W}) = \mathbf{A}\}$  of weighted matrices with binary projection equal to  $\mathbf{A}$ .

*Input.* Our Conditional Reconstruction Method (CReM) takes as input  $P(\mathbf{A})$ , i.e. the distribution over the space of binary configurations: this is treated as prior information and can be computed by using any available method. Clearly, given the superior performance of the dcGM as summarized above, we will select that particular model in our own analysis, but nonetheless we want to keep the method as general as possible by allowing for any input  $P(\mathbf{A})$ . Moreover, the CReM requires as input a set of weighted constraints  $\vec{C}(\mathbf{W})$  representing the available information about the system at hand. The observed numerical value of these constraints will be denoted by  $\vec{C}^*$ . The true, unobserved matrix will be denoted with  $\mathbf{W}^*$  and the associated binary projection with  $\mathbf{A}^*$ . Clearly  $\vec{C}(\mathbf{W}^*) = \vec{C}^*$ .

*Output.* The goal of the CReM is to derive the distribution over the ensemble of weighted configurations conditional on the prior information concerning the binary ensemble. To this aim, we look for a functional form of  $Q(\mathbf{W}|\mathbf{A})$  such that  $Q(\mathbf{W}|\mathbf{A}) = 0$  for  $\mathbf{W} \notin \mathbb{W}_{\mathbf{A}}$  and otherwise maximizing the *conditional entropy* [26]

$$S(\mathcal{W}|\mathcal{A}) = - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}|\mathbf{A}) \log Q(\mathbf{W}|\mathbf{A}) d\mathbf{W} \quad (3.1)$$

under the set of constraints

$$1 = \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}|\mathbf{A}) d\mathbf{W}, \quad \forall \mathbf{A} \in \mathbb{A} \quad (3.2)$$

$$\langle C_{\alpha} \rangle = \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}|\mathbf{A}) C_{\alpha}(\mathbf{W}) d\mathbf{W} = C_{\alpha}^*, \quad \forall \alpha \quad (3.3)$$

(notice that  $\langle \cdot \rangle$  denotes an average with respect to  $Q(\mathbf{W})$ ). Equation (3.2) defines the normalization of the conditional probability and ensures that the unconditional probability

$$Q(\mathbf{W}) = \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) Q(\mathbf{W}|\mathbf{A}) \quad (3.4)$$

is also normalized as

$$\int_{\mathbb{W}} Q(\mathbf{W}) d\mathbf{W} \equiv \sum_{\mathbf{A} \in \mathbb{A}} \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}) d\mathbf{W} = \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) = 1;$$

equation (3.3), instead, sets the target values  $\vec{C}^*$  of our constraints. The problem Lagrangean can be, thus, written as the following generalization of the Lagrangean valid in the unconditional case (see appendix A):

$$\begin{aligned} \mathcal{L} = & S(\mathcal{W}|\mathcal{A}) \\ & + \sum_{\mathbf{A} \in \mathbb{A}} \mu(\mathbf{A}) \left( 1 - \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}|\mathbf{A}) d\mathbf{W} \right) \\ & + \sum_{\alpha} \lambda_{\alpha} \left( C_{\alpha}^* - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}|\mathbf{A}) C_{\alpha}(\mathbf{W}) d\mathbf{W} \right). \end{aligned} \quad (3.5)$$

Differentiating with respect to  $Q(\mathbf{W}|\mathbf{A})$  and equating the result to zero leads to

$$Q_{\vec{\lambda}}(\mathbf{W}|\mathbf{A}) = \begin{cases} \frac{e^{-H_{\vec{\lambda}}(\mathbf{W})}}{Z_{\mathbf{A}, \vec{\lambda}}} & \mathbf{W} \in \mathbb{W}_{\mathbf{A}}, \\ 0 & \mathbf{W} \notin \mathbb{W}_{\mathbf{A}} \end{cases}, \quad (3.6)$$

where  $H_{\vec{\lambda}}(\mathbf{W}) = \sum_{\alpha} \lambda_{\alpha} C_{\alpha}(\mathbf{W})$  is the Hamiltonian and  $Z_{\mathbf{A}, \vec{\lambda}} = \int_{\mathbb{W}_{\mathbf{A}}} e^{-H_{\vec{\lambda}}(\mathbf{W})} d\mathbf{W}$  is the partition function for fixed  $\mathbf{A}$ . Note that we have introduced the subscript  $\vec{\lambda}$  to stress the dependence of the quantities on the Lagrange multipliers. The explicit functional form of  $Q_{\vec{\lambda}}(\mathbf{W}|\mathbf{A})$  can be obtained only by further specifying the functional form of the constraints as well.

*Parameters estimation.* The conditional probability distribution defined in equation (3.6) depends on the vector of unknown parameters  $\vec{\lambda}$ : a recipe is, thus, needed to provide their numerical estimation. In alignment with previous results [27–29], we now extend the maximum-likelihood recipe to deal with the conditional probability distribution we are considering here. Indeed, since we do not have access to the empirical adjacency matrix  $\mathbf{A}^*$ , it is not possible for us to compute the usual likelihood function  $Q_{\vec{\lambda}}(\mathbf{W}^*|\mathbf{A}^*)$  as a function of the parameters  $\vec{\lambda}$ . However we can go back to the more general problem from which the usual maximization of the likelihood derives, i.e. the maximization of the constrained entropy—in our case, the conditional expression (3.5)—and obtain a corresponding generalized likelihood that requires only the available information about the network.

Let us define  $\vec{\lambda}^*$  as the value of the parameters for which the constraints are satisfied, that is  $\langle \vec{C} \rangle^* = \vec{C}^*$ . By construction, the value  $\vec{\lambda}^*$  is such that the gradient of the Lagrangean  $\mathcal{L}$  is zero. Importantly, in the appendix we show that it is also the value that maximizes the *generalized likelihood*

$$\mathcal{G}(\vec{\lambda}) = -H_{\vec{\lambda}}(\langle \mathbf{W} \rangle^*) - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \log Z_{\mathbf{A}, \vec{\lambda}}, \quad (3.7)$$

where  $\langle \mathbf{W} \rangle^*$  indicates the unconditional ensemble average of  $\mathbf{W}$  when the desired constraints are satisfied. The use of this notation is legitimate by the fact that throughout this study we will consider linear constraints of the form

$$H_{\vec{\lambda}}(\mathbf{W}) = \sum_{j \neq i} \lambda_{ij} w_{ij}, \quad (3.8)$$

so that  $\langle H_{\vec{\lambda}} \rangle = \sum_{j \neq i} \lambda_{ij} \langle w_{ij} \rangle = H_{\vec{\lambda}}(\langle \mathbf{W} \rangle)$ . The definition (3.7) is justified by the relationship between likelihood and entropy proved below. Let us focus on the expression of the conditional entropy defined in equation (3.1): using equation (3.6) we can rearrange (and rename) it as

$$S(\vec{\lambda}) = - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) [-\langle H_{\vec{\lambda}} \rangle_{\mathbf{A}} - \log Z_{\mathbf{A}, \vec{\lambda}}]. \quad (3.9)$$



By evaluating  $S(\vec{\lambda})$  in  $\vec{\lambda}^*$ , we obtain

$$\begin{aligned} S(\vec{\lambda}^*) &= \langle H_{\vec{\lambda}^*} \rangle + \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \log Z_{\mathbf{A}, \vec{\lambda}^*} \\ &= H_{\vec{\lambda}^*}(\langle \mathbf{W} \rangle^*) + \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \log Z_{\mathbf{A}, \vec{\lambda}^*} \\ &= -\mathcal{G}(\vec{\lambda}^*). \end{aligned} \quad (3.10)$$

Notice that the starting point of our derivation was the definition of conditional entropy involving two ensemble averages, over both sets of binary configurations and link weight assignments. After its evaluation in  $\vec{\lambda}^*$ , the average over the set of link weight assignments has reduced to a single term, i.e.  $H_{\vec{\lambda}^*}(\langle \mathbf{W} \rangle^*)$ , while the average over the space of binary configurations has survived.

### 3.1. Constraining the strengths: the CReM<sub>A</sub> model

Let us now instantiate our CReM framework for the set of constraints usually considered (and empirically accessible) for the case of the reconstruction of financial networks, i.e. the out- and in-strength sequences,  $s_i^{\text{out}}(\mathbf{W}) = \sum_{j \neq i} w_{ij}$ ,  $\forall i$  and  $s_i^{\text{in}}(\mathbf{W}) = \sum_{j \neq i} w_{ji}$ ,  $\forall i$ . Imposing these constraints means introducing the Hamiltonian

$$\begin{aligned} H(\mathbf{W}) &= \sum_{i=1}^N [\beta_i^{\text{out}} s_i^{\text{out}}(\mathbf{W}) + \beta_i^{\text{in}} s_i^{\text{in}}(\mathbf{W})] \\ &= \sum_{i=1}^N \sum_{j \neq i} (\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}, \end{aligned} \quad (3.11)$$

which induces the partition function

$$\begin{aligned} Z_{\mathbf{A}} &= \prod_{i=1}^N \prod_{j \neq i} \left[ \int_0^{\infty} e^{-(\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}} dw_{ij} \right]^{a_{ij}} \\ &= \prod_{i=1}^N \prod_{j \neq i} \left( \frac{1}{\beta_i^{\text{out}} + \beta_j^{\text{in}}} \right)^{a_{ij}}. \end{aligned} \quad (3.12)$$

Using equation (3.6), we can write

$$Q(\mathbf{W}|\mathbf{A}) = \prod_{i=1}^N \prod_{j \neq i} q_{ij}(w_{ij}|a_{ij}) \quad (3.13)$$

where  $q_{ij}(w = 0|a_{ij} = 1) = 0$  and

$$q_{ij}(w|a_{ij} = 1) = \begin{cases} (\beta_i^{\text{out}} + \beta_j^{\text{in}}) e^{-(\beta_i^{\text{out}} + \beta_j^{\text{in}})w} & w > 0 \\ 0 & w \leq 0 \end{cases} \quad (3.14)$$

for each positive weight  $w_{ij}$ , showing that each pair-specific weight distribution conditional on the existence of the link is exponential with parameter  $\beta_i^{\text{out}} + \beta_j^{\text{in}}$ .

Now, in order to determine the values of the vectors of parameters  $\vec{\beta}^{\text{out}}$  and  $\vec{\beta}^{\text{in}}$  we maximize the generalized likelihood defined in equation (3.7), which reads

$$\begin{aligned} \mathcal{G}_{\text{CReM}_A} &= -\sum_{i=1}^N [s_i^{\text{out}*}(\mathbf{W}) \beta_i^{\text{out}} + s_i^{\text{in}*}(\mathbf{W}) \beta_i^{\text{in}}] \\ &\quad + \sum_{i=1}^N \sum_{j \neq i} f_{ij} \log(\beta_i^{\text{out}} + \beta_j^{\text{in}}) \end{aligned} \quad (3.15)$$

where the quantity

$$f_{ij} \equiv \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) a_{ij} = \langle a_{ij} \rangle \quad (3.16)$$

represents the expected value of  $a_{ij}$  over the ensemble of binary configurations, i.e. the *marginal* probability of a directed edge existing from node  $i$  to node  $j$  in the reconstructed binary ensemble, irrespective of whether edges are generated independently of each other by the binary reconstruction method. This makes our formulation entirely general with respect to the binary reconstruction method taken as input, as we made no assumption on the structure of  $P(\mathbf{A})$ . In particular, the *joint* probability  $P(\mathbf{A})$  for all links in network  $\mathbf{A}$  collectively appearing need not necessarily factorize as  $P(\mathbf{A}) = \prod_{i=1}^N \prod_{i \neq j} f_{ij}^{a_{ij}} (1 - f_{ij})^{1-a_{ij}}$  as in models with independent edges. For instance, in the case of the microcanonical binary configuration model (defined by sharply constraining the degrees),  $P(\mathbf{A})$  is the uniform distribution over all graphs with the a given degree sequence and cannot be factorized, as all links are mutually dependent through the sharp constraint. On the other hand, when the

canonical binary configuration model is considered,  $P(\mathbf{A})$  factorizes and  $f_{ij}$  coincides with the connection probability  $p_{ij}$  defining the model itself (see also the [appendix](#)) [1]. Both variants of the binary configuration model, as well as any other binary reconstruction method, can be taken as input into our conditional reconstruction method by specifying the corresponding  $P(\mathbf{A})$ . Note that, in cases where the explicit expression for  $P(\mathbf{A})$  is not available, one can still sample this distribution by taking multiple outputs of the binary reconstruction method and replacing averages over  $P(\mathbf{A})$  with sample averages.

Now, differentiating equation (3.15) with respect to  $\beta_i^{\text{out}}$  and  $\beta_i^{\text{in}}$  yields the system of  $2N$  coupled equations

$$\begin{cases} \langle s_i^{\text{out}} \rangle = \sum_{j \neq i} \frac{f_{ij}}{\beta_i^{\text{out}} + \beta_j^{\text{in}}} = s_i^{\text{out}*}, \forall i \\ \langle s_i^{\text{in}} \rangle = \sum_{j \neq i} \frac{f_{ji}}{\beta_j^{\text{out}} + \beta_i^{\text{in}}} = s_i^{\text{in}*}, \forall i \end{cases} \quad (3.17)$$

where  $\langle w_{ij} \rangle = \frac{f_{ij}}{\beta_i^{\text{out}} + \beta_j^{\text{in}}}$  and  $f_{ij}$  is taken as given—therefore excluded from the estimation procedure.

In what follows, we consider explicitly the case where *no entry* of the empirical adjacency matrix  $\mathbf{A}^*$  is known, so that all entries have to be dealt with probabilistically, in line with previous research in the field. However, an important feature of our framework is that it can incorporate the knowledge of *any set* of entries of  $\mathbf{A}^*$  as well. This means that, if we are certain about the presence ( $a_{ij}^* = 1$ ) or absence ( $a_{ij}^* = 0$ ) of certain edges, this deterministic knowledge will be reflected in the corresponding marginal connection probability being  $f_{ij} = a_{ij}^*$ . As an extreme example, let us consider the case in which *all entries* are known. In this case, we are led to the maximally informative specification  $f_{ij} \equiv a_{ij}^*, \forall i \neq j$  further implying that the system to be solved becomes

$$\begin{cases} \langle s_i^{\text{out}} \rangle = \sum_{j(\neq i)} \frac{a_{ij}^*}{\beta_i^{\text{out}} + \beta_j^{\text{in}}} = s_i^{\text{out}*}, \forall i \\ \langle s_i^{\text{in}} \rangle = \sum_{j(\neq i)} \frac{a_{ji}^*}{\beta_j^{\text{out}} + \beta_i^{\text{in}}} = s_i^{\text{in}*}, \forall i. \end{cases} \quad (3.18)$$

As a final general observation before moving to specific results, we would like to stress that the framework defining the CReM<sub>A</sub> model admits, as a particular case, the *Directed Enhanced Configuration Model* (DECM), i.e. the directed version of the continuous ECM [24]. For more properties of the CReM<sub>A</sub> model, see also the [appendix](#). The code to run the CReM<sub>A</sub> model is freely available at [30].

### 3.2. Testing the CReM<sub>A</sub> model

Let us now explicitly test the effectiveness of the CReM<sub>A</sub> model in reproducing two real-world systems, i.e. the World Trade Web (WTW) in the year 1990 [31] and e-MID in the year 2010 [32] (see the same references for a detailed description of the two data sets). In order to do so, we need to specify a functional form for the coefficients  $\{f_{ij}\}_{i,j=1}^N$ . As a first choice, let us implement the recipe

$$f_{ij} \equiv p_{ij}^{\text{dcGM}} = \frac{zs_i^{\text{out}} s_j^{\text{in}}}{1 + zs_i^{\text{out}} s_j^{\text{in}}}, \forall i \neq j \quad (3.19)$$

that defines the dcGM. Upon solving the system of equations (2.20), we obtain the numerical value of the parameters  $\vec{\beta}^{\text{out}}$  and  $\vec{\beta}^{\text{in}}$  by means of which we can analytically compute the expectation of any quantity of interest (via the so-called *delta method*, see also [28]). In particular, we have focused on (one of the four versions of) the *average nearest neighbors strength* (ANNS) [33]

$$s_i^{\text{out/out}} = \frac{\sum_j a_{ij} s_j^{\text{out}}}{k_i^{\text{out}}} \quad (3.20)$$

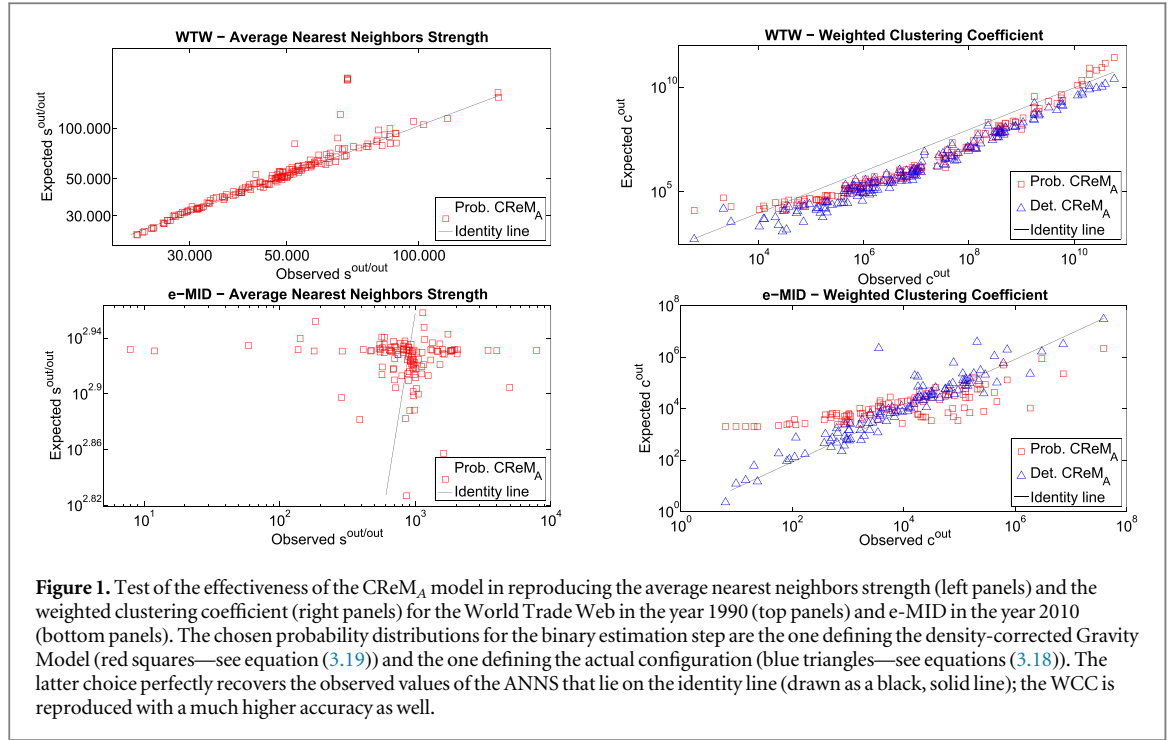
and on (one of the four versions of) the *weighted clustering coefficient* (WCC) [33]

$$c_i^{\text{out}} = \frac{\sum_{j(\neq i)} \sum_{k(\neq i,j)} w_{ij} w_{jk} w_{ik}}{k_i^{\text{out}}(k_i^{\text{out}} - 1)}; \quad (3.21)$$

the comparison between the observed and the expected value of the quantities above is shown in figure 1 for both systems. As a second choice, let us implement the deterministic recipe

$$f_{ij} \equiv a_{ij}^*, \forall i \neq j \quad (3.22)$$

whose effectiveness in reproducing the ANNS and the WCC is shown in figure 1 as well. Notice that the WCC is now reproduced much more accurately, a result further confirming that binary constraints affects the weights estimation as well; on the other hand, the ANNS is perfectly reproduced: specifying the network topology and the weighted marginals is, in fact, enough to recover the observed values.



### 3.3. Comparing binary reconstruction methods

As we have seen, our framework allows for any probability distribution to be taken as input to address the topology reconstruction step. We may, thus, ask what is the best recipe to reconstruct a given real-world network binary structure. In order to provide an answer, let us consider again the score function

$$\ln Q(W^*) = \ln P(A^*) + \ln Q(W^*|A^*) \quad (3.23)$$

and focus on the addendum  $\ln P(A^*)$ . Three prototypical distributions can be considered and compared:

- *deterministic distribution*: this choice implements the maximally informative position  $P(A) = \delta_{A,A^*}$  (see also equation (3.22)) and is equivalent to assuming that the empirical adjacency matrix  $A^*$  is known;
- *uniform probability distribution*: this choice corresponds to the maximally uninformative position  $P(A^*) = 2^{-N(N-1)}$  (i.e.  $f_{ij} \equiv \frac{1}{2}, \forall i \neq j$ );
- *dcGM probability distribution*: this choice implements the recipe  $P(A^*) = \prod_i \prod_{j \neq i} f_{ij}^{a_{ij}^*} (1 - f_{ij})^{1 - a_{ij}^*}$ , with  $f_{ij} \equiv p_{ij}^{\text{dcGM}} = \frac{z_i^{\text{out}} s_j^{\text{in}}}{1 + z_i^{\text{out}} s_j^{\text{in}}}, \forall i \neq j$  defining the dcGM.

In order to test the performance of the three competing models above, let us invoke the Akaike Information Criterion (AIC) [34] to select the model with the best trade-off between accuracy and parsimony. The AIC value is defined as

$$\text{AIC}_m = 2k_m - 2\mathcal{L}_m \quad (3.24)$$

for each model  $m$  in the basket, with  $\mathcal{L}_m$  indicating the log-likelihood value of model  $m$  and  $k_m$  indicating the number of parameters defining it (and to be estimated<sup>6</sup>). For the three alternatives above we have that

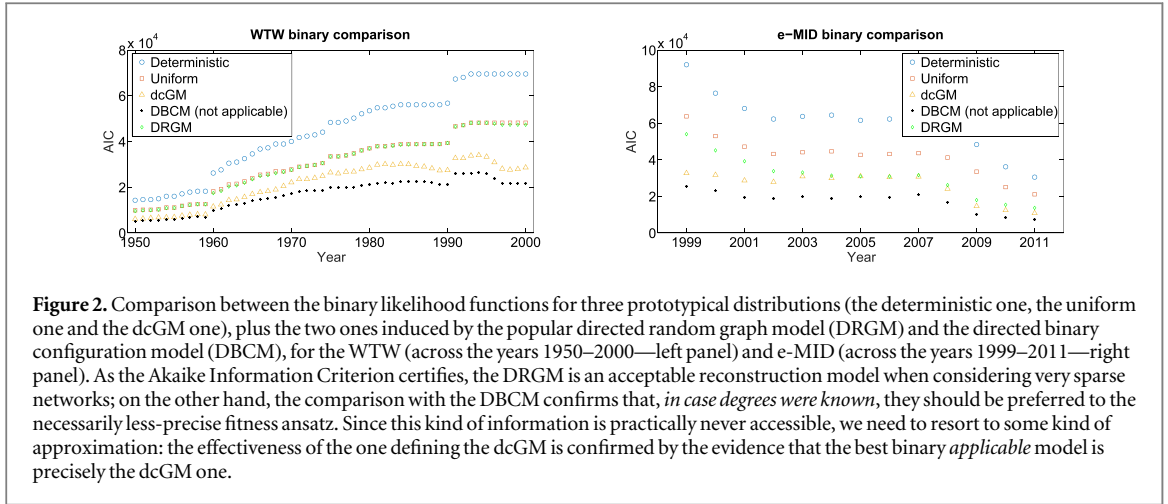
$$\text{AIC}_{\text{deterministic}} = 2N(N-1), \quad (3.25)$$

$$\text{AIC}_{\text{uniform}} = 2N(N-1) \ln 2, \quad (3.26)$$

$$\text{AIC}_{\text{dcGM}} = 2(1 - \mathcal{L}_{\text{dcGM}}), \quad (3.27)$$

where we have used the fact that the uniform model is non-parametric (i.e.  $k_{\text{uniform}} = 0$ ) and is characterized by the log-likelihood  $\mathcal{L}_{\text{uniform}} = -N(N-1) \ln 2$ , while the deterministic model is characterized by a probability  $P(A^*) = 1$  (implying  $\mathcal{L}_{\text{deterministic}} = 0$ ) and a number of parameters equal to  $k_{\text{deterministic}} = N(N-1)$  (all off-diagonal entries of  $A^*$  are separately specified). Moreover, we have added the comparison with the directed

<sup>6</sup> Note that, since we assume from the beginning that the in- and out-strengths of all nodes are known independently of the model used, we do not count the strengths among the model parameters.



**Figure 2.** Comparison between the binary likelihood functions for three prototypical distributions (the deterministic one, the uniform one and the dcGM one), plus the two ones induced by the popular directed random graph model (DRGM) and the directed binary configuration model (DBCM), for the WTW (across the years 1950–2000—left panel) and e-MID (across the years 1999–2011—right panel). As the Akaike Information Criterion certifies, the DRGM is an acceptable reconstruction model when considering very sparse networks; on the other hand, the comparison with the DBCM confirms that, *in case degrees were known*, they should be preferred to the necessarily less-precise fitness ansatz. Since this kind of information is practically never accessible, we need to resort to some kind of approximation: the effectiveness of the one defining the dcGM is confirmed by the evidence that the best binary *applicable* model is precisely the dcGM one.

random graph model and the directed binary configuration model, respectively, defined by

$$AIC_{\text{RGM}} = 2(1 - \mathcal{L}_{\text{DRGM}}), \quad (3.28)$$

$$AIC_{\text{DBCM}} = 2(2N - \mathcal{L}_{\text{DBCM}}), \quad (3.29)$$

where  $\mathcal{L}_{\text{DRGM}} = L \ln p + (N(N-1) - L) \ln(1-p)$ , with  $p = \frac{L}{N(N-1)}$  and  $\mathcal{L}_{\text{DBCM}} = \sum_i \sum_{j(\neq i)} a_{ij} \ln p_{ij}^{\text{DBCM}} + (1 - a_{ij}) \ln(1 - p_{ij}^{\text{DBCM}})$  with  $p_{ij}^{\text{DBCM}} = \frac{x_i y_j}{1 + x_i y_j}$ ,  $\forall i \neq j$  [1]. The criterion prescribes to prefer the model whose AIC value is minimum. Upon looking at figure 2, one realizes that the DRGM is a poorly-performing reconstruction model when the network link density is close to 0.5 as its performance cannot be distinguished from the one of the uniform model. When considering very sparse networks, on the other hand, knowing the link density means adding a non-trivial piece of information, potentially reducing the uncertainty about a given network structure to a large extent: this seems indeed to be the case for several temporal snapshots of eMID. On the other hand, the comparison with the DBCM confirms that, *in case degrees were known*, they should be preferred to the necessarily less precise fitness ansatz: however, as they are never known, including the DBCM is a merely academic exercise<sup>7</sup>. Nevertheless, the previous analysis still conveys an important message: the structure of real-world networks is characterized by a large amount of redundancy, as evident by noticing that the AIC value of the DBCM is much lower than the AIC of the fully deterministic model. Hence, the structure of complex networks can indeed be explained by only constraining a bunch of statistics, as exemplified by the degrees: however, since this kind of information is practically never accessible, we need to resort to some kind of approximation—whence our definition of the dcGM.

The importance of employing a method able to provide a reliable estimate of a network topology becomes evident when considering the problem of quantifying systemic risk (see [1] and references therein). To this aim, let us consider the *triangular loops* arising from various ‘risky’ triadic motifs connected to the underestimation of counterparty risk due to over-the-counter linkages in interbank networks [35]. An aggregate measure of incidence of such patterns is quantified by

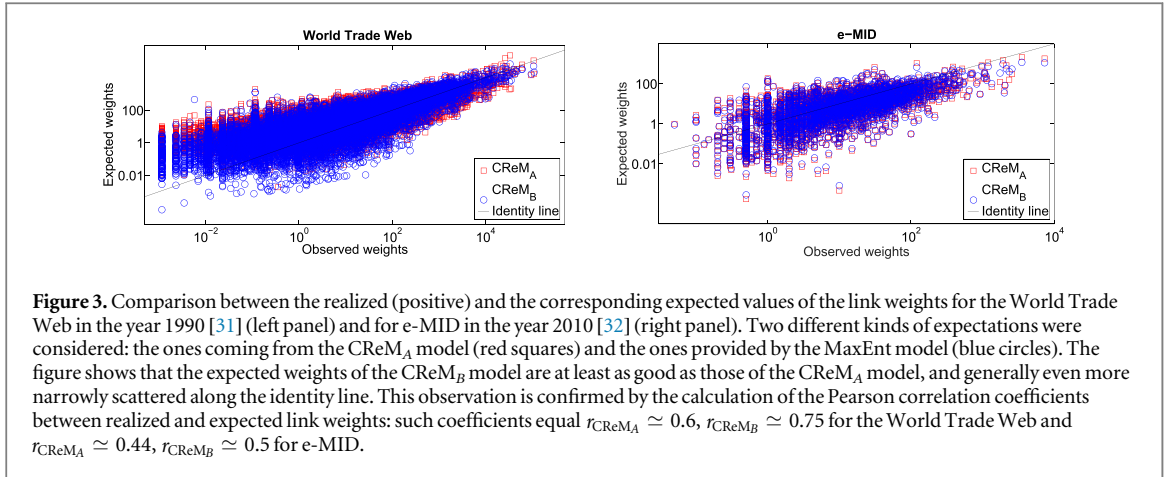
$$\bar{w}_{\Delta} = \frac{\sum_i \sum_{j(\neq i)} \sum_{k(\neq i, j)} w_{ij} w_{jk} w_{ki}}{\sum_i \sum_{j(\neq i)} \sum_{k(\neq i, j)} a_{ij} a_{jk} a_{ki}} = \frac{w_{\Delta}}{N_{\Delta}} \quad (3.30)$$

i.e. the average weight per loop. Notice that the expected value of such a quantity calls for the estimation of the probability that nodes  $i$ ,  $j$  and  $k$  establish a connection. For the sake of illustration, let us discuss the application of either the MaxEnt method or the minimum-density method to provide such an estimation. As previously discussed, the fully-connected topology output by the ME leads to  $N_{\Delta} \simeq N^3$ , i.e. to overestimating the number of cycles, in turn leading to an underestimation of systemic risk; on the other hand, the very sparse topology output by the MD leads to  $N_{\Delta} \simeq O(1)$ , i.e. to underestimating the number of cycles, in turn leading to an overestimation of systemic risk.

### 3.4. Further structuring the model: the CReM<sub>B</sub> model

In the previous sections we have introduced a novel framework for network reconstruction that has led to the definition of the CReM<sub>A</sub> model. Although this model provides an accurate reconstruction of real-world economic and financial networks, its implementation still requires the resolution of  $2N$  coupled nonlinear equations. Moreover, as it makes the maximally random hypothesis about link weights, given the empirical

<sup>7</sup> In this case, in fact, one would have enough information to run the DECM—a model that we have explicitly excluded from our comparison since completely unrealistic.



in-strength and out-strength of all nodes, the model does not allow to incorporate any further ansatz or assumption about the empirical relationship between link weights and the node strengths themselves. We now ask ourselves if it is possible to simplify the computational requirements of the CReM<sub>A</sub> model, while more flexibly constraining its randomness (again via entropy maximization) around a structured relationship that captures some empirical regularity relating link weights to the node strengths, thus improving the accuracy of the reconstruction of the weighted network as a whole.

To this aim, let us now specify a model potentially constraining the whole set of expected weights to given values  $\{w_{ij}^*\}$ , where in this case the asterisk denotes the ‘target’ value, which is not necessarily an observable one. In order to do so, let us formally constrain the unconditional expected values of all link weights, i.e. consider a Hamiltonian reading  $H(\mathbf{W}) = \sum_{j \neq i} \beta_{ij} w_{ij}$ . The derivation is analogous to the previous case and leads to the expression  $Q(\mathbf{W}|\mathbf{A}) = \prod_{j \neq i} q_{ij}(w_{ij}|a_{ij})$  with  $q_{ij}(w_{ij} = 0|a_{ij} = 1) = 0$  and

$$q_{ij}(w_{ij}|a_{ij} = 1) = \beta_{ij} e^{-\beta_{ij} w_{ij}}, \quad w_{ij} > 0 \quad (3.31)$$

i.e. to a conditional pair-specific weight distribution,  $q_{ij}(w_{ij}|a_{ij} = 1)$ , that is exponential with parameter  $\beta_{ij}$ . Analogously, the generalized likelihood function can be expressed as

$$\mathcal{G}_{\text{CReM}_B} = -\sum_{j \neq i} w_{ij}^* \beta_{ij} + \sum_{j \neq i} f_{ij} \log \beta_{ij} \quad (3.32)$$

and differentiating it with respect to  $\beta_{ij}$  leads to the equations

$$\langle w_{ij} \rangle = \frac{f_{ij}}{\beta_{ij}} = w_{ij}^*, \quad \forall i \neq j \quad (3.33)$$

that define the CReM<sub>B</sub> model.

Actual weights, however, can rarely be observed: hence, in order to implement the CReM<sub>B</sub> model, we need to replace  $\{w_{ij}^*\}_{i,j=1}^N$  with a set of accessible quantities. To this aim, we look for an additional ansatz based on empirical regularities relating link weights to node strengths in the data. In particular, as we already mentioned we notice that the MaxEnt model introduced in equation (2.1) provides good estimates of the realized (i.e. positive) link weights (despite the impossibility of generating zero link weights). Figure 3 shows the comparison between the observed, positive weights of the WTW in the year 1990 [31] and e-MID in the year 2010 [32] and two expectations: the ones coming from the CReM<sub>A</sub> model and the ones coming from the MaxEnt model of equation (2.1). One can see that the MaxEnt model produces expected weights that are more narrowly scattered around the empirical ones than the CReM<sub>A</sub> model. The calculation of the Pearson correlation coefficient between the empirical and expected weights from the two models confirms that the estimates coming from the MaxEnt model show a better agreement with the data (see caption of figure 3), throughout the considered time intervals.

The reason for the improved estimate in the MaxEnt model comes from the fact that the CReM<sub>A</sub> model makes the maximally random hypothesis about link weights, based on the empirical values of the in- and out-strengths of nodes. Real data turn out to be more structured than this completely random expectation, the MaxEnt model better capturing the structured relation. At the same time, while the original MaxEnt model would assume the same positive expression (2.1) for all link weights, the generalized framework used here allows us to embed the MaxEnt estimate into a conditional expectation for the link weight, given that the link is realized with the marginal probability  $f_{ij}$  implied by the desired prior distribution  $P(\mathbf{A})$ . This is easily done by replacing the set of target expected weights  $\{w_{ij}^*\}_{i,j=1}^N$  with the MaxEnt ansatz  $\{\hat{w}_{ij}^{\text{ME}}\}_{i,j=1}^N$  given by equation (2.1) and

inverting equation (3.33) to find the corresponding tensor of coefficients  $\beta$ . This yields

$$\beta_{ij} = \frac{f_{ij}}{\hat{w}_{ij}^{\text{ME}}} = \frac{Wf_{ij}}{s_i^{\text{out}*} s_j^{\text{in}*}}, \quad \forall i \neq j. \quad (3.34)$$

Notice that this choice only requires, as input, the out- and in-strength sequences of the actual network: as a consequence, the sufficient statistics for the  $\text{CReM}_A$  and  $\text{CReM}_B$  models coincide. Notice also that implementing the  $\text{CReM}_B$  model requires the resolution of  $O(N^2)$  decoupled equations.

Although the choice leading to equation (3.34) guarantees that non-negative strengths are preserved only in case  $f_{ij} > 0$ ,  $\forall i \neq j$ , in principle, one can set  $w_{ij}^*$  equal to the outcome of any other deterministic model for the link weights (e.g. IPF), not only the MaxEnt one. This would relax the requirements about the connection probability between nodes—hence allowing for zero-probability links as well—and ‘dress’ the chosen model with a weight distribution centered around the same value generated by the deterministic implementation (thereby turning the deterministic model into a probabilistic one). The code to run the version of the  $\text{CReM}_B$  model discussed here is freely available at [30].

For instance, we may use a more refined recipe improving the MaxEnt ansatz to higher order. To explain this point, we need to emphasize that the MaxEnt ansatz  $w_{ij}^* = \hat{w}_{ij}^{\text{ME}} = s_i^{\text{out}*} s_j^{\text{in}*} / W^*$  introduced in equation (2.1) has a disadvantage: it replicates the in- and out-strengths only if a self-loop with intensity  $\hat{w}_{ii}^{\text{ME}} = s_i^{\text{out}*} s_i^{\text{in}*} / W^*$  is added to each node  $i$ . This is easy to see by summing  $\hat{w}_{ij}^{\text{ME}}$  over  $i$  or  $j$  to produce the resulting  $s_j^{\text{in}*}$  or  $s_i^{\text{out}*}$ , respectively. In order to avoid adding self-loops, one may iteratively ‘redistribute’ the weight  $s_i^{\text{out}*} s_i^{\text{in}*} / W^*$  to all the other links. This generates a sequence of improved weights  $w_{ij}^* = \hat{w}_{ij}^{\text{ME}} + \hat{w}_{ij}^{(l)}$  for any desired order  $l$  of approximation [36]. To this aim, at least two different recipes can be devised. The first one prescribes to redistribute the terms  $s_i^{\text{out}*} s_i^{\text{in}*} / W^*$  on a complete graph with no self-loops via the IPF algorithm. In this way, margins are correctly reproduced in the limit  $l \rightarrow \infty$ , with the improved weights reading  $w_{ij}^* = \hat{w}_{ij}^{\text{ME}} + \hat{w}_{ij}^{(\infty)}$ :  $\hat{w}_{ij}^{(\infty)}$  can be estimated numerically, according to the iterative recipe described in [1, 18]; although the final result of this procedure achieves a refined match to the enforced margins, it makes the model no longer under complete analytical control. The second one prescribes to redistribute the terms  $s_i^{\text{out}*} s_i^{\text{in}*} / W^*$  on a fully connected matrix via the IPF algorithm, discard the diagonal terms and redistribute the latter ones in an iterative fashion; in this way, the correction term is always under analytical control even if this second variant requires the explicit generation of self loops to ensure that margins are reproduced at each iteration step: for example, the full prescription of the second method, at the second iteration, reads

$$w_{ij}^* = \begin{cases} \hat{w}_{ij}^{\text{ME}} + \hat{w}_{ij}^{(2)} & \forall i \neq j \\ \hat{w}_{ij}^{(2)} & \forall i = j \end{cases} \quad (3.35)$$

where  $\hat{w}_{ij}^{(2)} = s_i^{\text{out}*} s_i^{\text{in}*} s_j^{\text{out}*} s_j^{\text{in}*} / (W^* \sum_k s_k^{\text{out}*} s_k^{\text{in}*})$ . It is therefore up to the researcher to make the optimal choice between a more accurate and a more explicit version of the method, depending on the situation. Since the IPF algorithm cannot univocally determine a way to redistribute weights (as we have seen, the answer provided by the IPF algorithm depends on how one chooses to decompose the constraints) here we have decided to use the more explicit recipe  $w_{ij}^* = \hat{w}_{ij}^{\text{ME}}$ , provided its agreement with the empirical weights.

Let us now compare the effectiveness of the  $\text{CReM}_A$  and the  $\text{CReM}_B$  models in reproducing the two systems under consideration. In order to carry out the most general comparison possible, let us consider again our likelihood-based score function and focus on the second term, i.e. the proper conditional likelihood

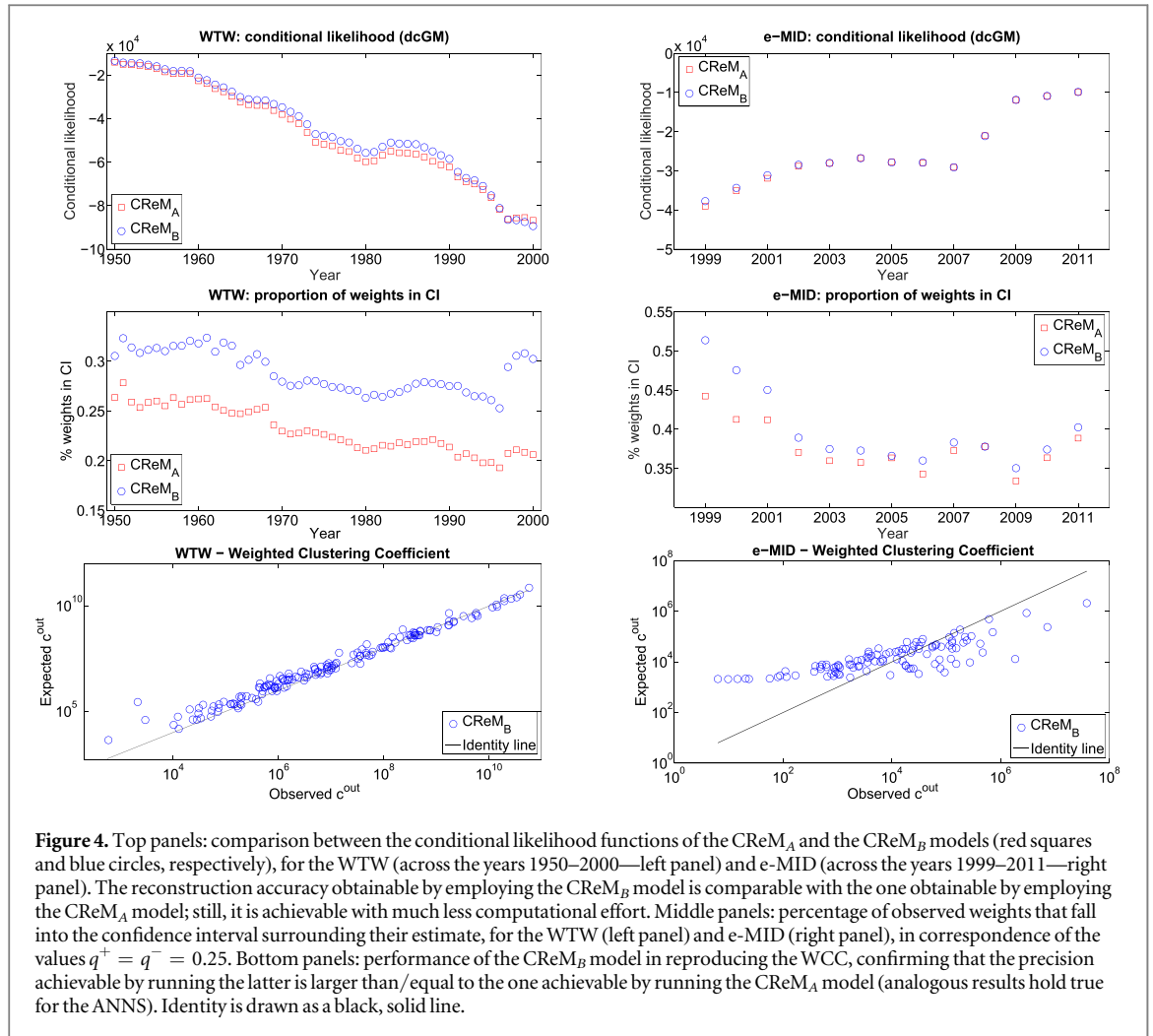
$$\mathcal{G}(\vec{\lambda}) = \ln Q(W^* | A^*); \quad (3.36)$$

we have employed the symbol  $\mathcal{G}$  since the expression of the conditional likelihood can be recovered by specifying the binary probability distribution  $P(A) = \delta_{A, A^*}$  in the expression of the generalized likelihood, i.e. equation (3.7). In this case,  $\mathcal{G}(\vec{\lambda})$  quantifies the effectiveness of a given model in reproducing the weighted structure of a network given its topology.

The performance of the  $\text{CReM}_A$  and the  $\text{CReM}_B$  models is, then, evaluated by comparing their conditional likelihood numerical values. The latter depend, respectively, on the parameters  $\beta_{\text{CReM}_A}$  and  $\beta_{\text{CReM}_B}$ ; thus, in order to compare our two models, we first solve equations (2.20) and (3.34) (with  $f_{ij} \equiv p_{ij}^{\text{dcGM}}$ ,  $\forall i \neq j$ , to capitalize on the result of the comparison between binary reconstruction algorithms) and then substitute  $\beta_{\text{CReM}_A}^*$  and  $\beta_{\text{CReM}_B}^*$  back in equation (3.36). We also explicitly notice that the sufficient statistics for the  $\text{CReM}_A$  and the  $\text{CReM}_B$  models coincide (they are, in fact, represented by the vectors of out- and in-strengths); hence, the AIC test would yield the same ranking as the one obtained by just comparing the likelihood functions.

Results are shown in figure 4 that confirm what we expected from the  $\text{CReM}_B$  model, i.e. a reconstruction accuracy that is comparable with the one of the  $\text{CReM}_A$  model but still achievable with much less computational





effort (in the case of the WTW, however, an even better agreement obtainable by running the  $\text{CRem}_B$  model can be clearly appreciated).

Let us now compare the  $\text{CRem}_A$  and  $\text{CRem}_B$  models by calculating the percentage of real weights that fall into the confidence intervals surrounding their estimates, by employing the same  $q^-$  and  $q^+$  values (see figure 4 and the appendix for the details of the calculations): the  $\text{CRem}_B$  model outperforms the  $\text{CRem}_A$  model in providing reliable estimates of actual weights. Notice that although the discrete versions of both the ECM and the DECM can provide error estimates, their computation is much easier within the novel, continuous framework considered here.

#### 4. Discussion

The extension of the ERG framework to account for conditional probability distributions aims at filling a methodological gap: defining a recipe for *unbiased weight assessment*, currently missing within the class of network reconstruction models.

The vast majority of the algorithms that have been proposed so far, in fact, combine methodologically different steps to estimate the purely topological network structure and the link weights, potentially distorting the entire procedure: as the derivation of our conditional reconstruction method proves, the topological information (summed up by the set of coefficients  $\{f_{ij}^N\}_{i,j=1}^N$ ) affects the estimation of link weights as well—see equations (2.20), (3.18), (3.33), etc.

These observations point out that a first source of bias is encountered whenever a *probabilistic* recipe for topological reconstruction is forced to output a *single* outcome instead of considering the entire ensemble of admissible configurations. Indeed, (mis)using a probabilistic method by implementing it as a deterministic one leads to an (arbitrary) privilege for a single configuration instead of correctly accounting for the entire support of the probability distribution defining the method itself. Since the expectation of any quantity of interest should be taken over the entire set of admissible configurations, privileging a particular realized topology will, in general,

lead to a wrong estimate of the inspected properties. Such an ‘extreme’ choice is allowed only when the number of admissible configurations indeed reduces to one, i.e. only in the limiting case in which the network topology is known exactly (i.e.  $f_{ij} = a_{ij}^*$ ,  $\forall i \neq j$ ).

A second source of bias is encountered when link weights are *deterministically* imposed via a recipe like the IPF algorithm, again because of the non-maximum-entropy nature of any deterministic algorithm. As a result, even in the extreme case in which *all* links of a given real-world network are known *ab initio*, the probability density of reproducing the weighted network with IPF-assigned weights would still be zero. By contrast, our calculations show that the correct procedure when all the topology is known is to assign weights probabilistically using equation (3.14), with parameters fixed by equation (3.18).

Our framework overcomes both limitations. The proposed CReM<sub>A</sub> and CReM<sub>B</sub> models, in fact, are fully probabilistic in nature and allow for the generation of network configurations characterized by continuous weights. Remarkably, for what concerns the binary estimation step, only the *marginal probability distributions*  $\{f_{ij}\}_{i,j=1}^N$  describing the behavior of the random variables  $\{a_{ij}\}_{i,j=1}^N$  are needed, a result that holds true irrespectively from the algorithm employed to derive the set of coefficients above.

Although it may be argued that the observations above hold true for the continuous version of the DECM as well, let us notice that its applicability is limited by the amount of information required to solve it, i.e. the knowledge of both the out- and in-degree sequences—a piece of information that is practically never accessible. On a more practical level, the numerical resolution of the CReM<sub>A</sub> and CReM<sub>B</sub> models is much less costly than the numerical resolution of the DECM. Moreover, our framework allows us to further simplify the problem of finding a numerical solution of the system of equations (2.20), by providing a recipe to solve *rescaled* versions of it: such a recipe can be employed to simplify calculations whenever a solution of the system above cannot be easily found at the considered scale (see also the [appendix](#)).

The comparison between our two competing models, then, reveals that the best performance is achieved by the CReM<sub>B</sub> that is the clear winner both in terms of accuracy and simplicity of implementation—as it does not require the resolution of any system of equations; even more so, each parameter of the CReM<sub>B</sub> model can be computed *independently* from the others, thus making the entire procedure parallelizable. To sum up, in order to achieve a fast and efficient reconstruction of weighted networks, we recommend the use of the CReM<sub>B</sub> model both in case of full uncertainty about the network topology and if the existence of some links is certain.

The codes to run both the CReM<sub>A</sub> and the CReM<sub>B</sub> versions of our method are freely available at [30].

## Acknowledgments

TS acknowledges support from the EU project SoBigData-PlusPlus (Grant No. 871042). DG acknowledges support from the Dutch Econophysics Foundation (Stichting Econophysics, Leiden, the Netherlands) and the Netherlands Organization for Scientific Research (NWO/OCW).

## Declarations

## Availability of data and materials

Data concerning the World Trade Web are described in KS Gleditsch [31] and can be found at the address <http://privatewww.essex.ac.uk/~ksg/exptradegdp.html>. Data concerning e-MID cannot be shared because of privacy issues preventing them from being publicly available. The codes implementing the conditional reconstruction algorithm can be found in [30].

## Competing Interests

The authors declare no competing financial interests.

## Authors Contributions

FP, TS and DG developed the method. FP performed the analysis. FP, TS and DG wrote the manuscript. All authors reviewed and approved the manuscript.

## Appendix A. The relationship between entropy and likelihood

*Standard case.* Let us revise the relationship between Shannon entropy and likelihood in the standard case. The maximization of Shannon entropy

$$S(\mathcal{W}) = - \sum_{\mathbf{A} \in \mathbb{A}} \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}) \log Q(\mathbf{W}) d\mathbf{W} \quad (\text{A.1})$$

under the set of constraints  $\vec{C}(\mathbf{W})$  leads to recover the functional form [24]

$$Q_{\vec{\lambda}}(\mathbf{W}) = \frac{e^{-H_{\vec{\lambda}}(\mathbf{W})}}{Z_{\vec{\lambda}}} \quad (\text{A.2})$$

where  $H_{\vec{\lambda}}(\mathbf{W}) = \sum_{\alpha} \lambda_{\alpha} C_{\alpha}(\mathbf{W})$  is a linear combination of the constraints and  $Z_{\vec{\lambda}} = \int_{\mathbb{W}} e^{-H_{\vec{\lambda}}(\mathbf{W})} d\mathbf{W}$  is the partition function. Upon substituting  $Q_{\vec{\lambda}}(\mathbf{W})$  into the Lagrangean function

$$\mathcal{L}(Q, \vec{\lambda}) = S(\mathcal{W}) + \mu \left( 1 - \sum_{\mathbf{A} \in \mathbb{A}} \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}) \right) + \sum_{\alpha} \lambda_{\alpha} (C_{\alpha}^* - \langle C_{\alpha} \rangle) \quad (\text{A.3})$$

(with  $\langle C_{\alpha} \rangle = \sum_{\mathbf{A} \in \mathbb{A}} \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}) C_{\alpha}(\mathbf{W})$ ) one recovers the expression

$$\mathcal{L}(Q(\vec{\lambda}), \vec{\lambda}) = \log Z_{\vec{\lambda}} + \sum_{\alpha} \lambda_{\alpha} C_{\alpha}^* = -\ln Q_{\vec{\lambda}}(\mathbf{W}^*) \quad (\text{A.4})$$

the last passage being valid for any graph  $\mathbf{W}^*$  such that  $\vec{C}(\mathbf{W}^*) = \vec{C}^*$ ; in other words, the equation above states that the Lagrangean functional is ‘minus’ the likelihood function of any graph  $\mathbf{W}^*$  on which the constraints  $\vec{C}(\mathbf{W}^*)$  assume the values  $\vec{C}^*$ .

Notice that in case the Lagrangean functional is evaluated in  $\vec{\lambda}^*$ , i.e. the parameters values ensuring that  $\langle \vec{C}_{\vec{\lambda}^*} \rangle = \vec{C}^*$ , the result

$$\mathcal{L}(Q(\vec{\lambda}^*), \vec{\lambda}^*) = S(\vec{\lambda}^*) = -\ln Q_{\vec{\lambda}^*}(\mathbf{W}^*) \quad (\text{A.5})$$

is recovered.

*Conditional case.* In case the functional

$$S(\mathcal{W}|\mathcal{A}) = - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \int_{\mathbb{W}_{\mathbf{A}}} Q(\mathbf{W}|\mathbf{A}) \log Q(\mathbf{W}|\mathbf{A}) d\mathbf{W} \quad (\text{A.6})$$

is, instead, considered, the expression  $Q_{\vec{\lambda}}(\mathbf{W}|\mathbf{A}) = \frac{e^{-H_{\vec{\lambda}}(\mathbf{W})}}{Z_{\mathbf{A}, \vec{\lambda}}}$  is recovered, with obvious meaning of the symbols. Upon substituting  $Q_{\vec{\lambda}}(\mathbf{W}|\mathbf{A})$  into the Lagrangean function defined in equation (3.5) one recovers the expression

$$\mathcal{L}(Q(\vec{\lambda}), \vec{\lambda}) = \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \log Z_{\mathbf{A}, \vec{\lambda}} + \sum_{\alpha} \lambda_{\alpha} C_{\alpha}^* = -\mathcal{G}(\vec{\lambda}) \quad (\text{A.7})$$

the last passage being valid for any graph  $\mathbf{W}^*$  such that  $\langle \vec{C} \rangle^* = \vec{C}(\mathbf{W}^*) = \vec{C}^*$ . Notice that in case the Lagrangean functional is evaluated in  $\vec{\lambda}^*$ , i.e. the parameters values ensuring that  $\langle \vec{C}_{\vec{\lambda}^*} \rangle = \vec{C}^*$ , the result

$$\mathcal{L}(Q(\vec{\lambda}^*), \vec{\lambda}^*) = S(\vec{\lambda}^*) = - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \ln Q_{\vec{\lambda}^*}(\langle \mathbf{W} \rangle^* | \mathbf{A}) = -\mathcal{G}(\vec{\lambda}^*) \quad (\text{A.8})$$

is recovered.

## Appendix B. The binary reconstruction step: deriving $P(\mathbf{A})$

The conditional reconstruction method works for any choice of  $P(\mathbf{A})$ . Here we derive two possible alternatives, to be selected according to the available information. The first possibility is deriving  $P(\mathbf{A})$  by maximizing Shannon entropy

$$S(\mathcal{A}) = - \sum_{\mathbf{A} \in \mathbb{A}} P(\mathbf{A}) \log P(\mathbf{A}), \quad (\text{B.1})$$

under the constraints represented by the out- and in-degree sequences  $\{k_i^{\text{out}}\}_{i=1}^N, \{k_i^{\text{in}}\}_{i=1}^N$ . This is the DBCM model [1] inducing a probability reading

$$P(\mathbf{A}) = \prod_{j \neq i} p_{ij}^{a_{ij}} (1 - p_{ij})^{1 - a_{ij}} \quad (\text{B.2})$$

with  $p_{ij} = \frac{x_i^{\text{out}} x_j^{\text{in}}}{1 + x_i^{\text{out}} x_j^{\text{in}}}$ . Unfortunately, this model is not viable for network reconstruction as the information about the degrees is practically never accessible. However, we can rest upon a certain approximately linear empirical linear relation generally found between the strengths and the Lagrange multipliers controlling for the

degrees (i.e.  $x_i^{\text{out}} \approx s_i^{\text{out}} \sqrt{a}$  and  $x_i^{\text{in}} \approx s_i^{\text{in}} \sqrt{b}$  [5]) to define a more realistic model. This so-called ‘fitness ansatz’ leads to our second model, whose linkage probability reads

$$P_{ij} = \frac{zs_i^{\text{out}}s_j^{\text{in}}}{1 + zs_i^{\text{out}}s_j^{\text{in}}} \quad (\text{B.3})$$

(having defined  $z \equiv \sqrt{ab}$ ), with the parameter  $z$  tuned to reproduce the empirical link density:

$\langle L \rangle = \sum_{j \neq i} P_{ij} = \sum_{j \neq i} \frac{zs_i^{\text{out}}s_j^{\text{in}}}{1 + zs_i^{\text{out}}s_j^{\text{in}}} = L^*$ . Equation (B.3) is the one characterizing the dcGM [5]. Other possible choices for  $P(\mathbf{A})$  are discussed in [1].

## Appendix C. Solving the rescaled CReM<sub>A</sub> problem

Let us now prove that our framework easily allows one to find a solution of the system of equations (2.20) in case the sufficient statistics is rescaled, i.e. divided by an opportunely defined factor (e.g.  $\kappa$ ). To this aim, let us consider the rescaled system

$$\begin{cases} \sum_{j(\neq i)} \frac{f_{ij}}{\beta_i^{\text{out}}(\kappa) + \beta_j^{\text{in}}(\kappa)} = \frac{s_i^{\text{out}*}}{\kappa}, \forall i \\ \sum_{j(\neq i)} \frac{f_{ji}}{\beta_j^{\text{out}}(\kappa) + \beta_i^{\text{in}}(\kappa)} = \frac{s_i^{\text{in}*}}{\kappa}, \forall i \end{cases}, \quad (\text{C.1})$$

where the symbols  $\vec{\beta}^{\text{out}}(\kappa)$  and  $\vec{\beta}^{\text{in}}(\kappa)$  stress that the searched solutions are functions of the chosen rescaling parameter  $\kappa$ . A solution of the system above reads

$$\beta_i^{\text{out}*}(\kappa) = \kappa \beta_i^{\text{out}*}, \forall i \quad (\text{C.2})$$

$$\beta_i^{\text{in}*}(\kappa) = \kappa \beta_i^{\text{in}*}, \forall i \quad (\text{C.3})$$

as it can be proven upon substituting it back into equations (C.1) and noticing that  $\{\beta_i^{\text{out}*}\}_{i=1}^N$  and  $\{\beta_i^{\text{in}*}\}_{i=1}^N$  are solutions of the system of equations (2.20). As our likelihood maximization problem admits a unique, global maximum, the prescription above allows us to easily identify it. Such a recipe turns out to enhance the chance of finding a solution to the system of equations (2.20) by solving a related problem at a more convenient scale.

## Appendix D. A ‘golden standard’ for reconstruction models

Let us notice that our framework allows us to define a sort of benchmark or ‘golden standard’ for reconstruction models, defined by the assumptions that both the actual network topology and the entire set of weights are accessible. Upon considering that  $\beta_{ij} = \frac{f_{ij}}{w_{ij}^*}$ ,  $\forall i \neq j$ , implementing both assumptions leads to the conditional probability distribution

$$q_{ij}(w_{ij}^* | a_{ij} = 1) = \frac{f_{ij}}{\langle w_{ij} \rangle} e^{-\frac{f_{ij}}{\langle w_{ij} \rangle} w_{ij}^*} = \frac{a_{ij}}{w_{ij}^*} e^{-\frac{a_{ij}}{w_{ij}^*} w_{ij}^*} = \frac{1}{e w_{ij}^*} \quad (\text{D.1})$$

further leading to

$$\mathcal{G}(\vec{\lambda}) = \ln Q(\mathbf{W}^* | \mathbf{A}^*) = \sum_{\{a_{ij}=1\}} (-1 - \ln w_{ij}^*) = -L - \sum_{\{a_{ij}=1\}} \ln w_{ij}^*$$

i.e. to the maximum value of the likelihood attainable by a canonical model with local constraints, still preserving the strengths.

## Appendix E. Confidence intervals-based comparison of network models

Both reconstruction models introduced in this manuscript induce pair-specific weight distributions: this allows a confidence interval to be defined around the expected value of each link weight. Since the procedure described below is valid for both the CReM<sub>A</sub> and the CReM<sub>B</sub> models, let us consider a generic exponential distribution whose parameter is  $\beta_{ij}$ .

In order to analytically estimate the searched confidence interval  $[w^-, w^+]$ , let us solve the equation

$$\int_{w^-}^{w_{ij}^* | a_{ij}=1} \beta_{ij} e^{-\beta_{ij} w_{ij}} dw_{ij} = q^- \quad (\text{E.1})$$

inverting which we find the left bound  $w^-$ ; upon noticing that  $\langle w_{ij} | a_{ij} = 1 \rangle = \frac{1}{\beta_{ij}}$  the result

$$w^- = -\frac{\ln[e^{-\beta_{ij}\langle w_{ij} | a_{ij} = 1 \rangle} + q^-]}{\beta_{ij}} = -\frac{\ln[e^{-1} + q^-]}{\beta_{ij}} \tag{E.2}$$

is obtained. Analogously, the second equation to be solved is

$$\int_{w=\langle w_{ij} | a_{ij} = 1 \rangle}^{w^+} \beta_{ij} e^{-\beta_{ij} w_{ij}} dw_{ij} = q^+ \tag{E.3}$$

leading to the result

$$w^+ = -\frac{\ln[e^{-\beta_{ij}\langle w_{ij} | a_{ij} = 1 \rangle} - q^+]}{\beta_{ij}} = -\frac{\ln[e^{-1} - q^+]}{\beta_{ij}}. \tag{E.4}$$

Thus, upon fixing the desired confidence levels  $q^-$  and  $q^+$ , the confidence interval  $[w^-, w^+]$  accompanying the estimation of the conditional expected weight  $\langle w_{ij} | a_{ij} = 1 \rangle$  is recovered. Generally speaking, such a confidence interval is not symmetric, given the peculiar form of the underlying probability distribution (i.e. the exponential one).

As the comparison shown in figure 4 reveals, the  $\text{CReM}_B$  model outperforms the  $\text{CReM}_A$  model in providing reliable estimates of actual weights.

### Appendix F. Deriving the continuous DECM

The continuous DECM is obtained by maximizing the entropy  $S(\mathcal{W})$  in equation (A.1) under the constraints represented by the out- and in-degree sequences and the out- and in-strength sequences:

$$\begin{cases} \langle k_i^{\text{out}} \rangle = \sum_{A \in \mathbb{A}} \int_{\mathbb{W}_A} Q(\mathbf{W}) k_i^{\text{out}}(\mathbf{W}) d\mathbf{W} = k_i^{\text{out}*} \\ \langle k_i^{\text{in}} \rangle = \sum_{A \in \mathbb{A}} \int_{\mathbb{W}_A} Q(\mathbf{W}) k_i^{\text{in}}(\mathbf{W}) d\mathbf{W} = k_i^{\text{in}*} \\ \langle s_i^{\text{out}} \rangle = \sum_{A \in \mathbb{A}} \int_{\mathbb{W}_A} Q(\mathbf{W}) s_i^{\text{out}}(\mathbf{W}) d\mathbf{W} = s_i^{\text{out}*} \\ \langle s_i^{\text{in}} \rangle = \sum_{A \in \mathbb{A}} \int_{\mathbb{W}_A} Q(\mathbf{W}) s_i^{\text{in}}(\mathbf{W}) d\mathbf{W} = s_i^{\text{in}*}. \end{cases} \tag{F.1}$$

The continuous DECM is defined by a probability distribution reading  $Q(\mathbf{W}) = \frac{e^{-H(\mathbf{W})}}{Z}$  where

$$H(\mathbf{W}) = \sum_{j \neq i} [(\alpha_i^{\text{out}} + \alpha_j^{\text{in}}) \Theta(w_{ij}) + (\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}] \tag{F.2}$$

and

$$\begin{aligned} Z &= \sum_{A \in \mathbb{A}} \int_{\mathbb{W}_A} e^{-H(\mathbf{W})} d\mathbf{W} \\ &= \prod_{j \neq i} \int_0^\infty [\delta(w_{ij} - 0) + \Theta(w_{ij})] e^{-(\alpha_i^{\text{out}} + \alpha_j^{\text{in}}) \Theta(w_{ij}) - (\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}} dw_{ij} \\ &= \prod_{j \neq i} \left[ 1 + e^{-(\alpha_i^{\text{out}} + \alpha_j^{\text{in}})} \int_0^\infty e^{-(\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}} dw_{ij} \right] \\ &= \prod_{j \neq i} \left[ 1 + \frac{e^{-(\alpha_i^{\text{out}} + \alpha_j^{\text{in}})}}{\beta_i^{\text{out}} + \beta_j^{\text{in}}} \right] \end{aligned} \tag{F.3}$$

( $d\mathbf{W}$  stands for  $\prod_{j \neq i} dw_{ij}$ , in the first passage) which finally leads to

$$Q(\mathbf{W}) = \prod_{j \neq i} q_{ij}(w_{ij}) = \prod_{j \neq i} \frac{(x_i^{\text{out}} x_j^{\text{in}})^{\Theta(w_{ij})} e^{-(\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}}}{1 + x_i^{\text{out}} x_j^{\text{in}} / (\beta_i^{\text{out}} + \beta_j^{\text{in}})}. \tag{F.4}$$

As we said in the main text, the  $\text{CReM}_A$  model admits the DECM as a particular case. In fact

$$q_{ij}(w_{ij}) = p_{ij}^{\text{DECM}} (\beta_i^{\text{out}} + \beta_j^{\text{in}}) e^{-(\beta_i^{\text{out}} + \beta_j^{\text{in}}) w_{ij}} \tag{F.5}$$

for any positive weight and with  $p_{ij}^{\text{DECM}}$  representing the probability that a link pointing from  $i$  to  $j$  exists. Indeed, the functional forms of the expected weights under the  $\text{CReM}_A$  and the DECM models coincide as well: in fact,

$\langle w_{ij} \rangle_{\text{DECM}} = \frac{p_{ij}^{\text{DECM}}}{\beta_i^{\text{out}} + \beta_j^{\text{in}}}$ , with  $p_{ij}^{\text{DECM}} = p_{ij}^{\text{DECM}}(\alpha_i, \alpha_j, \beta_i, \beta_j)$ , the latter expression explicitly showing the joint

role played by binary and weighted constraints in determining the topological structure of the network at hand  $p_{ij}$  is, in fact, defined by the Lagrange multipliers associated with both the out- and in-degrees and the out- and in-strengths of nodes  $i$  and  $j$ .

## ORCID iDs

Tiziano Squartini  <https://orcid.org/0000-0001-9011-966X>

Diego Garlaschelli  <https://orcid.org/0000-0001-6035-1783>

## References

- [1] Squartini T, Caldarelli G, Cimini G, Gabrielli A and Garlaschelli D 2018 Reconstruction methods for networks: the case of economic and financial systems *Phys. Rep.* **757** 1–47
- [2] Cimini G, Squartini T, Saracco F, Garlaschelli D, Gabrielli A and Caldarelli G 2019 The statistical physics of real-world networks *Nat. Rev. Phys.* **1** 58–71
- [3] Mastrandrea R, Squartini T, Fagiolo G and Garlaschelli D 2014 Enhanced reconstruction of weighted networks from strengths and degrees *New J. Phys.* **16** 043022
- [4] Cimini G, Squartini T, Gabrielli A and Garlaschelli D 2015 Estimating topological properties of weights networks from limited information *Phys. Rev. E* **92** 040802
- [5] Cimini G, Squartini T, Garlaschelli D and Gabrielli A 2015 Systemic risk analysis on reconstructed economic and financial networks *Sci. Rep.* **5** 15758
- [6] Andrecut M 2016 Systemic risk, maximum entropy and interbank contagion *Int. J. Mod. Phys. C* **27** 1650148
- [7] Halaj G and Kok C 2013 Assessing interbank contagion using simulated networks *Comput. Manage. Sci.* **10** 157–86
- [8] Bacharach M 1965 Estimating nonnegative matrices from marginal data *Int. Econ. Rev.* **6** 294–310
- [9] Gandy A and Veraart L 2016 A Bayesian methodology for systemic risk assessment in financial networks *Manage. Sci.* **63** 12
- [10] Wells S 2005 Financial interlinkages in the United Kingdom's interbank market and the risk of contagion *Bank of England* **2004** 230
- [11] Upper C 2011 Simulation methods to assess the danger of contagion in interbank markets *J. Financ. Stab.* **7** 111–25
- [12] Mistrulli P E M 2011 Assessing financial contagion in the interbank market: maximum entropy versus observed interbank lending patterns *J. Bank. Financ.* **35** 1114–27
- [13] Anand K, Craig B and von Peter G 2015 Filling in the blanks: network structure and interbank contagion *Quant. Finance* **15** 625–36
- [14] Baral P and Figue J P 2012 Estimation of bilateral exposures *A Copula Approach* (Huntingdon: Mimeo)
- [15] Drehmann M and Tarashev N 2013 Measuring the systemic importance of interconnected banks *J. Financ. Intermediation* **22** 586–607
- [16] Montagna M and Lux T 2017 Contagion risk in the interbank market: a probabilistic approach to cope with incomplete structural information *Quant. Finance* **17** 101–20
- [17] Mastromatteo I, Zarinelli E and Marsili M 2012 Reconstruction of financial networks for robust estimation of systemic risk *J. Stat. Mech.* **P03011**
- [18] Di Gangi D, Lillo F and Pirino D 2016 Assessing systemic risk due to fire sales spillover through maximum entropy network reconstruction arXiv:1509.00607
- [19] Moussa A 2016 Contagion and systemic risk in financial networks *PhD Thesis* Columbia University
- [20] Anand K et al 2018 The missing links: a global study on uncovering financial network structures from partial data *J. Financ. Stab.* **35** 1114–27
- [21] Mazarisi P and Lillo F 2017 Methods for reconstructing interbank networks from limited information: a comparisons *Econophysics and Sociophysics: Recent Progress and Future Directions* (Berlin: Springer) pp 201–15
- [22] Ramadiah A, Caccioli F and Fricke D 2017 Reconstructing and stress testing credit networks *ESRB Working Paper Series* **84** 44
- [23] Lebacher M, Cook S, Klein N and Kauermann G 2019 In search of lost edges: a case study on reconstructing financial networks arXiv:1909.01274
- [24] Gabrielli A, Mastrandrea R, Caldarelli G and Cimini G 2019 The grand canonical ensemble of weighted networks *Phys. Rev. E* **99** 030301
- [25] Squartini T, Mastrandrea R and Garlaschelli D 2015 Unbiased sampling of network ensembles *New J. Phys.* **17** 023052
- [26] Cover T M and Thomas J A 2006 *Elements of Information Theory* (New York: Wiley)
- [27] Garlaschelli D and Loffredo M I 2008 Maximum likelihood: extracting unbiased information from complex networks *Phys. Rev. E* **78** 015101(R)
- [28] Squartini T and Garlaschelli D 2011 Analytical maximum-likelihood method to detect patterns in real networks *New J. Phys.* **13** 083001
- [29] Squartini T and Garlaschelli D 2017 Reconnecting statistical physics and combinatorics beyond ensemble equivalence arXiv:1710.11422
- [30] <http://meh.imtlucca.it>
- [31] Gleditsch K S 2002 Expanded trade and GDP data *J. Conflicts Resolution* **46** 712–24
- [32] Iori G, De Masi G, Vasilis Precup O, Gabbi G and Caldarelli G 2006 A network analysis of the Italian overnight money market *J. Econ. Dyn. Control* **32** 259–78
- [33] Squartini T, Fagiolo G and Garlaschelli D 2011 Randomizing world trade: II. A weighted network analysis *Phys. Rev. E* **84** 046118
- [34] Akaike H 1974 A new look at the statistical model identification *IEEE Trans. Autom. Control* **19** 716–23
- [35] Squartini T, van Lelyveld I and Garlaschelli D 2013 Early-warning signals of topological collapse in interbank networks *Sci. Rep.* **3** 3357
- [36] Squartini T, Cimini G, Gabrielli A and Garlaschelli D 2017 Network reconstruction via density sampling *Appl. Netw. Sci.* **2** 3