Randomized reference models for temporal networks

L. Gauvin,[1] M. Génois,[2,3] M. Karsai,[4] M. Kivelä,[5] T. Takaguchi,[6] E. Valdano,[7] and C. L. Vestergaard[2,8,9,*]

[1] *Data Science Lab, ISI Foundation, Turin, Italy.*
[2] *Aix Marseille Univ., Univ. Toulon, CNRS, CPT, Marseille, France.*
[3] *GESIS, Leibniz-Institute for the Social Sciences, Cologne, Germany.*[†]
[4] *Univ Lyon, ENS de Lyon, Inria, CNRS, Université Claude Bernard Lyon 1, LIP, F-69342, LYON Cedex 07, France.*
[5] *Department of Computer Science, Aalto University School of Science, P.O. Box 12200, FI-00076 Aalto, Finland.*
[6] *National Institute of Information and Communications Technology, 4-2-1 Nukui-Kitamachi, Koganei, Tokyo 184-8795, Japan.*
[7] *Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Tarragona, Spain.*
[8] *Decision and Bayesian Computation, Pasteur Institute, CNRS UMR 3571, 25-28 rue du Dr Roux, 75015 Paris, France.*[†]
[9] *Bioinformatics and Biostatistics Hub, C3BI, Pasteur Institute, CNRS USR 3756, 25-28 rue du Dr Roux, 75015 Paris, France.*[†]

(Dated: June 12, 2018)

Many real-world dynamical systems can successfully be analyzed using the temporal network formalism. Empirical temporal networks and dynamic processes that take place in these situations show heterogeneous, non-Markovian, and intrinsically correlated dynamics, making their analysis particularly challenging. Randomized reference models (RRMs) for temporal networks constitute a versatile toolbox for studying such systems. Defined as ensembles of random networks with given features constrained to match those of an input (empirical) network, they may be used to identify statistically significant motifs in empirical temporal networks (i.e. overrepresented w.r.t. the null random networks) and to infer the effects of such motifs on dynamical processes unfolding in the network. However, the effects of most randomization procedures on temporal network characteristics remain poorly understood, rendering their use non-trivial and susceptible to misinterpretation. Here we propose a unified framework for classifying and understanding microcanonical RRMs (MRRMs). We use this framework to propose a canonical naming convention for existing randomization procedures, classify them, and deduce their effects on a range of important temporal network features. We furthermore show that certain classes of *compatible* MRRMs may be applied in sequential composition to generate more than a hundred new MRRMs from existing ones surveyed in this article. We provide a tutorial for the use of MRRMs to analyze an empirical temporal network and we review applications of MRRMs found in literature. The taxonomy of MRRMs we have developed provides a reference to ease the use of MRRMs, and the theoretical foundations laid here may further serve as a base for the development of a principled and systematic way to generate and apply randomized reference null models for the study of temporal networks.

Randomized reference models (RRMs) [1, 2] (also known as *null models* [1, 3–16]; *reshuffling methods* [17]; *randomization techniques* [14, 18], *procedures* [4, 9, 14, 18, 19], *strategies* [9], *schemes* [14] or *methods* [20]; or simply *randomizations* [1, 14, 18–23]) deal with the controlled destruction of given temporal or topological structures in a complex network in order to create a probability distribution of reference networks retaining certain features of the original network. In practice, this is often done by shuffling the original (empirical) temporal network while keeping the features in question constrained. The resulting ensemble of randomized networks serves as a null reference for the original temporal network, which can be used to infer whether given features the network are significantly different from random when other features of the network are constrained [5, 11, 24–26], to investigate how different features of a temporal network affect dynamical processes unfolding on top of it [2, 3, 9, 10, 12, 13, 16, 17, 19, 27–35], and to study their effect on the network's controllability [18, 36]. For example, by comparing how a given dynamical process evolves on the randomized network with how it evolves on the original network, we may identify how the network's particular characteristics affect the dynamical process.

RRMs have emerged as a powerful toolbox for the study of the dynamics on and of temporal networks, and have been used to study complex systems in a broad range of fields ranging from sociology, over epidemiology, infrastructure, and economics, to biology. They have namely been applied to temporal networks of: human face-to-face interactions [2, 9, 10, 13, 19, 20, 22, 23, 29, 33, 35] and physical proximity [12, 16, 35, 37]; prostitution [10, 27, 33, 35]; brain function [37]; human mobility [38]; livestock transport [4]; mobile phone calls [2, 3, 11, 28, 30] and text messages [2]; email correspondences [2, 3, 10, 18, 19, 21, 23, 33]; online communities [10, 21, 33, 37, 39, 40]; editing of Wikipedia pages [8]; and world trade [6, 7, 15]. This witnesses to the very general applicability of such RRMs. RRMs defined by numerical shuffling procedures may notably be

* cvestergaard@gmail.com
† Present address

applied to systems for which no realistic generative model exists, which is the case for most real systems.

A multitude of different RRMs for temporal networks exists. However, they are often represented as shuffling algorithms suitable for numerical calculations, and their effects on temporal network features are rarely investigated systematically and remain poorly understood. This lack of systematic understanding of the methods is not only a theoretical problem but it has lead to severe practical problems in the literature. First, there are no unified naming conventions for the RRMs, which makes comparing the methods used in different studies difficult and has lead to a situation where the algorithms producing equivalent RRMs are given a multitude of different names. Second, researchers are confronted with the nontrivial problems of how to choose and develop randomization techniques, in which order they should be applied, and how to interpret the results in order to be able to identify important characteristics for each given dynamical phenomenon and for each temporal network under study.

We review the temporal network RRMs used in the literature and find that most of them fall into a class of methods which give a uniform probability of sampling all networks which have a given set of features constrained to the same value as that of the original data [30]. Inspired by the concept of microcanonical ensembles in statistical physics [41, 42] we will call these methods *microcanonical* randomized reference models (MRRMs), and represent them in a formal framework where they are fully defined by the set of features they constrain. This principled approach has several advantages over the algorithmic representation: As MRRMs are completely defined by the constraints that they impose, we propose an unambiguous naming convention for MRRMs of temporal networks based on these constraints. Further, this framework enables us to build a taxonomy of existing MRRMs, which lists their effects on important temporal network features and (partially) orders them by the amount of features they constrain. This hierarchy allows researchers to apply MRRMs so that the fixed features of the original data are systematically reduced. Finally, we show how and when new MRRMs can be devised by applying one previously implemented algorithm after another.

Reference models or null models which keep parts of the features of original data and shuffle the rest are clearly widely applicable outside of temporal networks, which are a relatively new topic even within network science. For example, MRRMs are closely related to exact (permutation) tests of classical statistics [43] and to conditionally uniform graph tests (CUGTs) found in the sociology literature [21, 44, 45]. Furthermore, even though we are mostly concerned with temporal networks, the framework of MRRMs we use here is directly applicable to a far more general class of systems, namely any system that can be considered a realization of a state in a predefined microcanonical state space.

It is our hope that, in addition to categorizing previous RRMs and surveying the literature, the unified framework and taxonomy we present would serve as a starting point for the development of a general and principled randomization-based approach for the characterization and analysis of networked dynamical systems. To this end, we provide a practical guide to researchers who want to apply MRRMs to analyze temporal network data. We provide (at `https://github.com/mgenois/RandTempNet`) a pure python library implementing the MRRMs presented in this article; further, for applications to larger networks we provide (at `https://github.com/bolozna/Events`) a fast Python library with core functions written in C++ implementing most of the MRRMs.

Roadmap to this article

This article is organized in a way that we start from the theoretical foundations of temporal networks and randomized reference models and build up to the applications and literature review. Readers who are solely interested in finding a catalog and classification of existing MRRMs may jump directly to Section IV and refer to Tables I and II for notation). A brief overview and a walk-through example of how to apply MRRMs for the analysis of temporal networks are found in Section V, while a literature review of applications of MRRMs is given in Section VI.

In Section I we provide the basic representations of temporal networks and lay out the general framework of microcanonical RRMs. Notably, we introduce the concepts and definitions needed to order and build hierarchies of MRRMs used in Section IV. Further, we develop a theory for composing MRRMs by applying one algorithm after another, which is used in Section III to implement shuffling methods and can be used to create new MRRMs not found in the literature yet.

In Section II we list a number of important features of temporal networks and establish their hierarchy. In Section III we describe how MRRMs are implemented in practice, and we define general classes of generators of MRRMs. This classification is based on the various temporal network representations defined in Section I. Further, we see how the compositions of MRRMs discussed earlier only in theory can be used in practice. In Sec. IV we use the theory and features defined earlier to present a naming convention for MRRMs for temporal networks, and we classify MRRMs found in the literature and list their effects on important temporal network features.

In Sec. V we give a procedure for statistical analysis and hypothesis testing using MRRMs and provide a walk-through example of how to apply them to analyze an empirical temporal network. In Sec. VI we finally give a review of applications of MRRMs found in the literature.

Finally, the two appendices extend the framework to directed temporal networks (Appendix A) and provide pseudocode examples for algorithmic implementation of

MRRMs (Appendix B).

I. FUNDAMENTAL DEFINITIONS AND GENERAL RESULTS

In this section we define fundamental concepts and derive general results for temporal networks and microcanonical randomized reference models (MRRMs). We focus on microcanonical RRMs as these represent the class of maximum entropy models that can be obtained directly by randomly sampling constrained permutations of an empirically observed network. We note in passing that, while we here apply the formalism to the study of temporal networks, the results derived in subsections I B and I C apply more generally to MRRMs for any system with a finite and discrete state space.

Subsection I A provides definitions of a temporal network and of the important notion of a temporal network *feature*, defined as any function that takes a temporal network as input and which is used as the constraint imposed by a MRRM. The subsection also presents two-level representations of a temporal network which facilitate definition of temporal network features (Section II) and the implementation of many MRRMs (Section III). Subsection I B provides a rigorous definition of a MRRM and introduces their basic properties. It also formalizes the concept that one MRRM shuffles more than another, and the partial order of MRRMs induced by this notion of *comparability* allows us to build a hierarchy of MRRMs (Section IV). These hierarchies are also useful in the practical employment of MRRMs (Section V).

Finally, Subsection I C considers how we may combine a pair of MRRMs to form a new MRRM. *Composition* of two reference models by applying one shuffling method after another is a practical way of creating new reference models. However, not all MRRMs are *compatible* in a way that their composition would produce another MRRM, and in this last subsection we develop theory needed to show that two MRRMs are compatible and to indentify the MRRMs resulting from compositions. The theorems developed in this subsection are instrumental for defining the important classes of compatible MRRMs employed in practice in Section III.

A. Temporal network

We consider a system consisting of N individual nodes engaging in intermittent dyadic interactions observed over a period of time from $t = t_{\min}$ to $t = t_{\max}$; in a social network, for example, the nodes are persons, in an ecological network nodes are species, while in a transport network they are locations. A temporal network is our representation of such an observation.

Definition I.1. *Temporal network.* A temporal network $G = (\mathcal{V}, \mathcal{C})$ is defined by the set of nodes $\mathcal{V} = \{1, 2, ..., N\}$ and a set of events $\mathcal{C} = \{c_1, c_2, \ldots, c_C\}$, where each event $c_q = (i_q, j_q, t_q, \tau_q)$ denotes an interaction between nodes i_q and j_q during the time-interval $[t_q, t_q + \tau_q)$.

Definition I.1 encompasses both temporal networks with directed interactions (e.g. for phone-call or instant messaging networks) and undirected interactions (e.g. for face-to-face or proximity networks). For directed networks, we may adopt the convention that the direction of an event c_q is from i_q to j_q. For undirected networks, the presence of the event (i_q, j_q, t_q, τ_q) implies the symmetric interaction (j_q, i_q, t_q, τ_q), and in practice we may impose $i_q < j_q$ for efficient data storage. For simplicity, we consider undirected temporal networks in examples, but all models and methods may directly be applied to directed networks and may easily be extended to take into account the directionality of interactions (see Appendix A).

For some systems, e.g., email communications or instant messaging [2, 3, 10, 18, 19, 21, 23, 33], events are instantaneous; in other cases, event durations are so short compared to the time-intervals between them, the *inter-event durations*, that they may be treated as instantaneous [3, 30]. Both cases are included in the above framework by setting $\tau = 0$. We may then reduce our representation of the sequence of events to a sequence of reduced instantaneous events, leading us to the definition below.

Definition I.2. *Instant-event temporal network.* An instant-event temporal network $G = (\mathcal{V}, \mathcal{E})$ is defined by the set of nodes $\mathcal{V} = \{1, 2, ..., N\}$ and a set of reduced events $\mathcal{E} = \{e_1, e_2, \ldots, e_E\}$, where each event $e_q = (i_q, j_q, t_q)$ describes an interaction between nodes i_q and j_q at time t_q, but where the duration is implicit.

Some systems with continuous dyadic activity, notably face-to-face interaction and proximity networks [46, 47], are recorded with a coarse time resolution at evenly spaced points in time, $t = t_{\min}, t_{\min} + \Delta t, t_{\min} + 2\Delta t, \ldots, t_{\max} - \Delta t$. In this case we may either represent the system as an *instant-event network*, where the events mark a beginning of an activity at each measurement time and the time-resolution $\tau = \Delta t$ is implicit. Alternatively, and more commonly, the system is represented as a temporal network, where consecutive measurements of activity between the same pair of nodes are merged into a single event with the duration τ indicating the length of the event.

Definition I.3. *Temporal network feature.* A *feature* $\mathbf{x}$ is any function that takes as an input any temporal network G. Formally, given a space of temporal networks, the *state space* $\mathcal{G}$, $\mathbf{x}$ is a function $\mathbf{x} : \mathcal{G} \to \mathcal{X}$, where $\mathcal{X}$ is an arbitrary set.

Typically a feature is a vector-valued function. In this case $\mathcal{X} = \mathbb{R}^d$, where d is the dimension of the feature. However, the definition allows for more general function, e.g., a space of networks. An important temporal network feature is the *static graph*, which summarizes the time-aggregated topology of a temporal network.

Definition I.4. *Static graph.* The static graph, G^{stat}, is a function which returns a simple unweighted graph $G^{\text{stat}}(G) = (\mathcal{V}, \mathcal{L})$ with the same set of nodes $\mathcal{V}$ as the original temporal network G and *links* $\mathcal{L} = \{(i,j) : (i,j,t,\tau) \in \mathcal{C}\}$, which includes all pairs of nodes (i,j) that interact at least once in G.

Note that by Definition I.4 the static graph is a feature of a temporal network. Conversely, we may see the temporal network as a direct generalization of the static graph to include information about the time-evolution of the system's topology [1]. Note that here we have defined a static graph as an unweighted graph, but one could also use the number of times the contact occurs or the total duration of contact as an edge weight (see Section II).

1. *Two-level temporal network representations*

Sometimes it is useful to separate the static structure and the temporal aspect in the definition of temporal network as opposed to having them mixed together like in definitions I.1 and I.2. This can be done by separating these two aspects into two levels, either by first defining the network structure and then how it changes in time, or by first defining the sequence of activation times and then the network structure at each of those times. We call the first of these options a *link-timeline network* and the second a *snapshot-graph sequence*. These two-level temporal network representations will be practical for visualizing temporal networks, defining important temporal network features (Sec. II), and for designing and implementing MRRMs for temporal networks (Sec. III).

Definition I.5. *Link-timeline network.* A link-timeline network $G_{\mathcal{L}} = (G^{\text{stat}}, \boldsymbol{\Theta})$ represents a temporal network by using the static graph $G^{\text{stat}} = (\mathcal{V}, \mathcal{L})$ to indicate the pairs of nodes that interact at least once during the observation period (Def. I.4). To each link $(i,j) \in \mathcal{L}$ we associate a *timeline* $\Theta_{(i,j)} \in \boldsymbol{\Theta}$, which indicates when the corresponding nodes interact. Each timeline is given by a sequence

$$\Theta_{(i,j)} = \left(\left(t^1_{(i,j)}, \tau^1_{(i,j)}\right), \left(t^2_{(i,j)}, \tau^2_{(i,j)}\right), \ldots, \left(t^{n_{(i,j)}}_{(i,j)}, \tau^{n_{(i,j)}}_{(i,j)}\right) \right) \tag{1}$$

where $t^m_{(i,j)}$ is the start of the mth event on link (i,j), with $\tau^m_{(i,j)}$ its duration, and $n_{(i,j)}$ is the total number of events taking place over the link [48].

Example I.1. Figure 1 shows an example link-timeline network of a temporal network consisting of four nodes and recorded at finite time-resolution. Panel (a) shows the static graph G^{stat}, while panel (b) shows the link timelines $\boldsymbol{\Theta}$.

Alternative to the link timeline networks we may think of a temporal network as a time-varying sequence of instantaneous graph snapshots. This leads to the following definition:

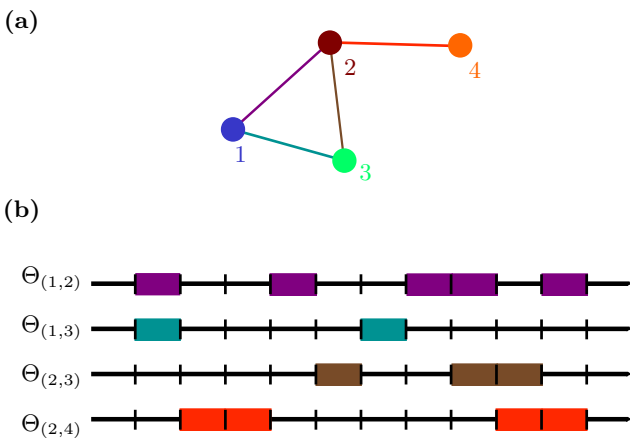


FIG. 1: **Link-timeline network.** Graphical representation of the link-timeline network $G_{\mathcal{L}}$ of an illustrative temporal network consisting of four nodes and recorded in discrete time. (a) Static graph G^{stat} showing links between nodes. Links are drawn between pairs (i,j) of nodes that interact. (b) Timelines $\boldsymbol{\Theta} = \{\Theta_{(1,2)}, \Theta_{(1,3)}, \Theta_{(2,3)}, \Theta_{(2,4)}\}$ of the links $\mathcal{L}$ in the graph showing when each link is active.

Definition I.6. *Snapshot-graph sequence.* A *snapshot-graph sequence*, $G_{\mathcal{T}} = (\mathcal{T}, \boldsymbol{\Gamma})$, represents a temporal network using a sequence of times, $\mathcal{T} = (t_1, t_2, \ldots, t_T)$, and a sequence of snapshot graphs,

$$\boldsymbol{\Gamma} = (\Gamma^1, \Gamma^2, \ldots, \Gamma^T) \ , \tag{2}$$

where for each $m = 1, 2, \ldots, T$, $\Gamma^m \in \boldsymbol{\Gamma}$ is associated to $t_m \in \mathcal{T}$. The *snapshot graphs* are defined as graphs $\Gamma^m = (\mathcal{V}, \mathcal{E}^{t_m})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}^t$ is the set of edges for which there is an event taking place at time t,

$$\mathcal{E}^t = \{(i,j) : (i,j,t) \in \mathcal{E}\} \ . \tag{3}$$

Instantaneous-event networks can be represented as snapshot-graph sequences by constructing the sequence of times $\mathcal{T}$ as the times at which at least one event takes place. This is a natural representation especially for networks which are recorded with fixed time-resolution (as the sequences of times become $\mathcal{T} = (\Delta t, 2\Delta t, \ldots, T)$) and if the time resolution is coarse enough so that the individual snapshot graphs do not become too sparse.

We use the shorthand Γ^t to refer to the snapshot graph associated with the time $t \in \mathcal{T}$ (i.e. Γ^{t_m} for m such that $t_m = t$).

Example I.2. Figure 2 shows the snapshot-graph sequence for the same temporal network as in Fig. 1.

Note that the two-level temporal networks do not add anything new to the temporal network structure, but they are simply alternative ways of representing them, because any temporal network can be uniquely

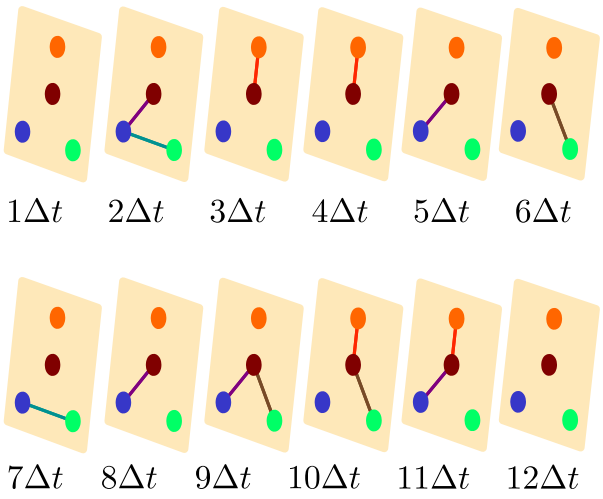


FIG. 2: **Snapshot-graph sequence.** Sequence of snapshot graphs of the temporal network shown in Fig. 1.

represented as a link-timeline network, and any instant-event temporal network can be uniquely represented as a snapshot-graph sequence. Despite this, the representations are often used for specific type of systems and they come with their own perspective on temporal networks.

The link-timeline networks are often used for data that is sparse in time such that only a few links are active at each time instant. Further, because the static network is made explicit in its definition it is easy to think that the temporal network has a latent static network which manifests as activation events of the links. For example, for an email communication data represented as link-timeline network one might consider the static graph as an aquintance or friendship network where each social tie is activated during the communication events. The structure also guides the generation of randomized reference models, because it is easy to either randomize the static graph and keep the link timelines, or randomize the timelines while keeping the static graph.

The snapshot-graph sequences are more likely used for data that is dense in time such that each snapshot graph contains a reasonable number of links. Further, this representation is natural for networks that change in time while the network nature of the system is still important at each separate time instance. For example, networks in which the structure changes in the same or longer timescales as dynamics on that network, it is important to be able to look at the topology of the network at each time instance. Here, again, the structure guides the construction of randomized reference models: It is convenient to define shuffling methods where the order of the snapshot graphs change or each of them are independently randomized.

The two-level temporal network representations provide convenient ways to define and generate MRRMs that constrain certain overall properties. We will explore this in detail in Section III and see that many RRMs found in the literature are implemented this way in Section IV.

B. Randomized reference model

Here we give a rigorous definition of a microcanonical RRM (MRRM). We furthermore develop several related concepts and properties of MRRMs. We will use these to consistently describe and hierarchically rank MRRMs in Sections III and IV.

1. *Basic definitions*

Consider a predetermined finite [49] set $\mathcal{G}$ of possible *states*, i.e. temporal networks, and a single observation from the state space, the input network $G^* \in \mathcal{G}$. Many procedures leading to a sampling from a conditional probability distribution $P(G|G^*)$ defined on $\mathcal{G}$ that are used for comparison against G^* could be considered to be *randomized reference models* (RRMs). In order for such models to be useful for testing hypothesis and finding effect sizes they need to retain some of the properties of the original network G^* and randomize others in a controlled way. In the context of graphs the most popular choices of RRMs include Erdős-Renyi (ER) models [50, 51], configuration models [51, 52], and exponential random graph models [51].

Here we will focus on models that exactly preserve certain features but are otherwise maximally random, and call these *microcanonical randomized reference models* (MRRMs). In the context of graphs the variant of the ER model that returns uniformly at random a graph with N nodes and L edges, and the variant of the configuration model that returns uniformly randomly selected graph with degree sequence $\mathbf{k} = (k_i)_{i\in\mathcal{V}}$, also known as the *Maslov-Sneppen* model [53], are MRRMs for graphs.

We will next formally define MRRMs and show that they have several attractive properties. We shall also see in Section IV that most RRMs for temporal networks generated by reshuffling an input network are of this type.

Definition I.7. *Microcanonical randomized reference model (MRRM).* Consider any function $\mathbf{x}$ that has the set $\mathcal{G}$ as the domain (i.e. a temporal network feature, Def. I.3). A MRRM, denoted by $\mathrm{P}[\mathbf{x}]$, is then a model which given $G^* \in \mathcal{G}$ returns $G \in \mathcal{G}$ with probability:

$$P_{\mathbf{x}}(G|G^*) = \frac{\delta_{\mathbf{x}(G),\mathbf{x}(G^*)}}{\Omega_{\mathbf{x}}(G^*)} \quad , \tag{4}$$

where δ is the Kronecker delta function, and $\Omega_{\mathbf{x}}(G^*) = \sum_{G\in\mathcal{G}} \delta_{\mathbf{x}(G),\mathbf{x}(G^*)}$ is a normalization constant.

We will some times use shorthand notation $\mathbf{x}^* = \mathbf{x}(G^*)$, and $\Omega_{\mathbf{x}^*} = \Omega_{\mathbf{x}}(G^*)$. Furthermore, because the conditional probability depends only on the value of the feature of G^* we can define the notation $P_{\mathbf{x}}(G|\mathbf{x}^*) = P_{\mathbf{x}}(G|G^*)$.

In the above definition the feature function $\mathbf{x}$ defines the features of G^* that are retained in the randomized reference model. In statistical physics terms $\Omega_{\mathbf{x}^*}$ is the *microcanonical partition function* and is equal to the *multiplicity* of microstates.

Note that restricting ourselves to a single feature entails no loss in generality since any number of distinct features may be combined into one tuple-valued feature, e.g., for two distinct features $\mathbf{x}$ and $\mathbf{y}$, we may simply define a third tuple-valued feature $\mathbf{z} = (\mathbf{x}, \mathbf{y})$ (see Def. I.11). Furthermore, limiting ourselves to finite ensembles in Definition I.7 does not limit its applicability in practice. While continuously varying feature values in principle induce an uncountable number of states, ensembles are always finite in practice due to finite time-resolution of recordings and finite computer precision.

Example I.3. The Maslov-Sneppen model [53] is an example of a MRRM for static graphs. On the state space of static graphs, it is defined as P[$\mathbf{k}$]. It maps an input network G^* to a microcanonical ensemble of networks that all have the same sequence of node degrees $\mathbf{k}^* = \mathbf{k}(G^*)$ but are otherwise uniformly random.

While the definition of MRRMs is written as a conditional probability it is often useful to use alternative representation of MRRMs.

Definition I.8. *MRRM representations:*

1. *Transition matrix.* A MRRM is a linear stochastic operator mapping the state space $\mathcal{G}$ to itself. For a given indexing of the state space $\mathcal{G}$, we represent a MRRM by a transition matrix $\mathbf{P}^{\mathbf{x}}$ with elements

$$\mathbf{P}^{\mathbf{x}}_{ij} = P_{\mathbf{x}}(G_j|G_i) \ . \tag{5}$$

   $\mathbf{P}^{\mathbf{x}}$ is always a block diagonal matrix where inside each block the elements have the same value.

2. *Partition of the state space.* The feature function defines an equivalence relation and thus partitions the state space [54]: Given $\mathbf{x}$ one can construct a partition of the state space $\{\mathcal{G}_i\}$ (i.e., a set of subsets of $\mathcal{G}$ such that each element of $\mathcal{G}$ is in exactly one subset) such that $G, G' \in \mathcal{G}_i$ if $\mathbf{x}(G) = \mathbf{x}(G')$. The set where $G^* \in \mathcal{G}$ belongs to in this partition is the $\mathbf{x}$-*equivalence class* of G^* and is denoted by $\mathcal{G}_{\mathbf{x}^*} = \{G \in \mathcal{G} : \mathbf{x}(G) = \mathbf{x}^*\}$. Note that the partition function is the cardinality of this set, $\Omega_{\mathbf{x}^*} = |\mathcal{G}_{\mathbf{x}^*}|$.

3. *Permutation method.* An algorithm that transforms G^* into G according to the Definition I.7. These algorithms often permute (i.e. shuffle) some elements of G^*. Note that multiple algorithms or permutation procedures might correspond to the same MRRM and in this case these are considered here to be the same permutation method.

All of these representations are equivalent in a sense that they completely and uniquely specify a MRRM (in the case of permutation methods this is by definition). For example, each $P[\mathbf{x}]$ defines exactly one partition and each partition is defined by exactly one $P[\mathbf{x}]$. Note that two feature functions $\mathbf{x}$ and $\mathbf{y}$ might correspond to the same partition, transition matrix, or permutation method, but in this case the two functions also give the same conditional probabilities in Definition I.7 and we say that $P[\mathbf{x}] = P[\mathbf{y}]$. The above representations will be used interchangeably in the following text.

2. *Hierarchies of MRRMs*

Some MRRMs shuffle more (i.e. keep less) structure than others. We will next formalize this notion which allows us to compare MRRMs and build hierarchies between them. Such hierarchies turn out to be useful for classification of MRRMs. Further, sequences of MRRMs where each one shuffles slightly more structure than the previous one are often used in practice [26, 30]. A central concept is that of *comparability* of MRRMs.

Definition I.9. *Comparability.* We will write $\mathrm{P}[\mathbf{x}] \leq \mathrm{P}[\mathbf{y}]$ for two MRRMs if there exists a function $\mathbf{f}$ for which $\mathbf{y}(G) = \mathbf{f}(\mathbf{x}(G))$ for all states $G \in \mathcal{G}$. We say that P[$\mathbf{x}$] and P[$\mathbf{y}$] are *comparable* if $\mathrm{P}[\mathbf{x}] \leq \mathrm{P}[\mathbf{y}]$ or $\mathrm{P}[\mathbf{y}] \leq \mathrm{P}[\mathbf{x}]$.

The intuition behind this notation is that when $\mathrm{P}[\mathbf{x}] \leq \mathrm{P}[\mathbf{y}]$, then the P[$\mathbf{y}$] shuffles more of the given structure than P[$\mathbf{x}$]. Due to the one-to-one correspondence between a MRRM and its feature function, we shall likewise use the notation $\mathbf{x} \leq \mathbf{y}$ when referring to the feature functions.

Example I.4. In the space of all static graphs with N nodes, one can define the MRRMs corresponding to the Erdős-Rényi random graph model [50], P[L], and the Maslov-Sneppen model [53], P[$\mathbf{k}$]. We have $L = \sum_i k_i/2$, so L is a function of $\mathbf{k}$ and $\mathrm{P}[L] \geq \mathrm{P}[\mathbf{k}]$. Conversely, $\mathbf{k}$ is not a function of L, so $\mathrm{P}[L] \not\leq \mathrm{P}[\mathbf{k}]$, as networks with different degree sequences can have the same number of links.

Representing MRRMs as partitions (see Definition I.8) provides a useful and intuitive way of thinking about comparisons of MRRMs, because the MRRM comparison relation is exactly the natural comparison relation of the partitions.

Proposition I.1. *Equivalence with partition refinements.* $\mathrm{P}[\mathbf{x}] \leq \mathrm{P}[\mathbf{y}]$ if and only if the partition $\mathcal{G}_{\mathbf{x}}$ is finer than $\mathcal{G}_{\mathbf{y}}$.

Proof. The existence of a function $\mathbf{f}$ that allows to calculate $\mathbf{y}$ solely from $\mathbf{x}$ means that all networks in a given $\mathbf{x}$-equivalence class $\mathcal{G}_{\mathbf{x}^*}$ (Def. I.8) correspond to the same value of $\mathbf{y}$. Conversely, each $\mathbf{y}$-value may correspond to multiple values for $\mathbf{x}$, so different networks in a given $\mathbf{y}$-equivalence class $\mathcal{G}_{\mathbf{y}^*}$ may correspond to different values

of $\mathbf{x}$. Thus $\mathcal{G}_{\mathbf{x}(G^*)} \subseteq \mathcal{G}_{\mathbf{y}(G^*)}$ for all $G^* \in \mathcal{G}$. In terms of partitions this means that $\mathbf{x}$ leads to a finer partition of the state space than $\mathbf{y}$. □

Borrowing the terminology from the theory of set partitions, we say for $\mathrm{P}[\mathbf{x}] \leq \mathrm{P}[\mathbf{y}]$ that $\mathrm{P}[\mathbf{x}]$ is *finer* than $\mathrm{P}[\mathbf{y}]$ and equivalently that $\mathrm{P}[\mathbf{y}]$ is *coarser* than $\mathrm{P}[\mathbf{x}]$. We will also refer to $\mathrm{P}[\mathbf{x}]$ as a *refinement* of $\mathrm{P}[\mathbf{y}]$ and to $\mathrm{P}[\mathbf{y}]$ as a *coarsening* of $\mathrm{P}[\mathbf{x}]$.

The partition representation is especially useful here as the properties of refinements of set partitions are inherited to the comparison relation of the MRRMs: for example, we can now see that the use of the notation $\geq$ is approriate as the relation it denotes is indeed a partial order:

Corollary I.1. *Comparability induces a partial order.* The relation $\geq$ is a partial order over the state space.

The above corollary follows immediately from the fact that partition refinements relations give partial orders. As with any partially ordered set, one can draw Hasse diagrams to display the relationships between different MRRMs, and this turns out to be a convenient way of visually organizing the various MRRMs found in the literature (see Section IV).

The set partitions always have uniquely defined minimum and maximum partitions, and these are meaningful in the case of MRRMs. We call them the zero and unity elements.

Definition I.10. *Zero and unity elements.* The *zero element*, $\mathrm{P}[0] = \mathrm{P}[G]$, is the MRRM which shuffles nothing, i.e. the one that always returns the input network and where the feature returns the entire temporal network. The *unity element*, $\mathrm{P}[1]$, is the MRRM that shuffles everything, i.e. the one that returns all networks in the state space with equal probability and where the feature is constant and does not depend on the input.

The zero element corresponds to the partition where each network is in its own set and the unity element to the partition where there is only a single set. The zero and unity elements are always in the top and bottom of a hierarchy of MRRMs: $\mathrm{P}[0] \leq \mathrm{P}[\mathbf{x}] \leq \mathrm{P}[1]$ for any $\mathbf{x}$.

Example I.5. Continuing from Example I.4, but limiting the state space to the set of static graphs, $\{G_1, G_2, G_3\}$, consisting of 3 nodes, $\mathcal{V} = \{1, 2, 3\}$, and 2 links, $\mathcal{L}(G_1) = ((1, 2); (1, 3))$, $\mathcal{L}(G_2) = ((1, 2); (2, 3))$, $\mathcal{L}(G_2) = ((1, 3); (2, 3))$. Since the number of links is the same in all graphs, the partition of the ER model contains only one set $\mathcal{G}_L = \{\{G_1, G_2, G_3\}\}$. However, the degree sequences of the networks differ, $\mathbf{k}(G_1) = (2, 1, 1)$, $\mathbf{k}(G_2) = (1, 2, 1)$, and $\mathbf{k}(G_2) = (1, 1, 2)$, so the partition related to the Maslov-Sneppen model separates all networks $\mathcal{G}_{\mathbf{k}} = \{\{G_1\}, \{G_2\}, \{G_3\}\}$. The partition $\mathcal{G}_{\mathbf{k}}$ is a refinement of $\mathcal{G}_L$ and thus $\mathbf{k} \leq L$. Note that for this state space the Maslov-Sneppen model is the zero element $\mathrm{P}[\mathbf{k}] = \mathrm{P}[0]$ and the ER model is the unity element $\mathrm{P}[L] = \mathrm{P}[1]$.

The interesting hierarchical structure is found between the zero and unity elements, and as we will see, the structure can very rich. Again, the set partition representation gives us a glimpse of the theoretical understanding of this structure. The total number of possible MRRMs for a given state space is the same as the number of set partitions. This is given by the Bell number $B_{|\mathcal{G}|}$ [55], which grows faster than exponentially with the state space size $|\mathcal{G}|$ [56]. We also know that even though the number of MRRMs in the hierarchy can be large, it can only be relatively flat as compared to this number: The largest possible number of MRRMs all satisfying a total order (i.e. for which $\mathrm{P}[\mathbf{x}_1] \geq \mathrm{P}[\mathbf{x}_2] \geq \mathrm{P}[\mathbf{x}_3] \ldots$) is the maximum chain length in the set of partitions of the state space, which is equal to $|\mathcal{G}| + 1$. Thus, if we insist on selecting a collection of MRRMs that is totally ordered, we can at most include an exponentially vanishing part of the possible MRRMs. However, in practice these theoretical limitations are not of much concern as the number of possible networks, $|\mathcal{G}|$, is often extremely large.

C. Combining microcanonical randomized reference models

In this section we explore how two different MRRMs may be combined to generate another MRRM. In particular by *intersection*, which generates a MRRM that shuffles less than either of the two, and by *composition*, which generates a RRM that is not necessarily microcanonical, but if it is, it shuffles more than either of the two. Especially the type of compositions which produce MRRMs are of practical interest as they provide a way of producing new shuffling methods by applying one shuffling algorithm after another.

The latter part of this section will be devoted to exploring under which conditions the composition of two MRRMs is microcanonical (we then say that the two MRRMs are *compatible*). We develop a concept called *conditional independence given a common coarsening* and show that it characterizes compatibility. Finally, we show that specific types of refinements of compatible MRRMs are also compatible and identify the way the resulting MRRM inherits the features of the two input models. This result provides a practical way of constructing new MRRM algorithms by combining existing ones.

1. Intersections and compositions

For any two given features $\mathbf{x}$ and $\mathbf{y}$ and associated MRRMs $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$, we define the *intersection* of the MRRMs, $\mathrm{P}[\mathbf{x}, \mathbf{y}]$, as the MRRM that constrains both features simultaneously. We can write the following defintion:

Definition I.11. *Intersection of randomized reference models.* The *intersection* of $\mathbf{x}$ and $\mathbf{y}$ is $(\mathbf{x}, \mathbf{y})$, and for the models we write $\mathrm{P}[\mathbf{x}, \mathbf{y}] = \mathrm{P}[(\mathbf{x}, \mathbf{y})]$

Note that the intersection is only defined for MRRMs and that by defintion the result is another MRRM. In terms of conditional probabilities the intersection becomes

$$P_{G|(\mathbf{x},\mathbf{y})}(G|\mathbf{x}^*,\mathbf{y}^*) = \frac{\delta_{\mathbf{x}(G),\mathbf{x}^*}\delta_{\mathbf{y}(G),\mathbf{y}^*}}{\Omega_{(\mathbf{x}^*,\mathbf{y}^*)}} \ , \qquad (6)$$

where $\mathbf{x}^* = \mathbf{x}(G^*)$, $\mathbf{y}^* = \mathbf{y}(G^*)$, and G^* is the input network.

The partition of $\mathcal{G}$ induced by $\mathrm{P}[\mathbf{x},\mathbf{y}]$ is trivially given by the set of pairwise intersections between the $\mathbf{x}$-equivalence classes and the $\mathbf{y}$-equivalence classes, i.e. $\mathcal{G}_{(\mathbf{x},\mathbf{y})}(G) = \mathcal{G}_{\mathbf{x}}(G) \cap \mathcal{G}_{\mathbf{y}}(G)$ for all $G \in \mathcal{G}$, and $\Omega_{(\mathbf{x}^*,\mathbf{y}^*)} = |\mathcal{G}_{\mathbf{x}^*} \cap \mathcal{G}_{\mathbf{y}^*}| = \sum_{G\in\mathcal{G}} \delta_{\mathbf{x}(G),\mathbf{x}^*}\delta_{\mathbf{y}(G),\mathbf{y}^*}$ That is, the partition $\mathcal{G}_{(\mathbf{x},\mathbf{y}}$ is finer than $\mathcal{G}_{\mathbf{x}}$ or $\mathcal{G}_{\mathbf{y}}$, and the MRRM $\mathrm{P}[\mathbf{x},\mathbf{y}]$ shuffles less (or equally) than $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$.

The effects of intersection with the zero and unity elements are also easy to see. The unity is a neutral element that has no effect on the intersection $\mathrm{P}[\mathbf{x},1] = \mathrm{P}[\mathbf{x}]$, because adding a constant to the feature function output doesn't affect the partitioning of the networks at all. The zero is an absorbing element $\mathrm{P}[\mathbf{x},0] = \mathrm{P}[0]$, because adding extra information to the feature function that already contains the full network doesn't change anything. In fact, from set partitions we know that the intersection gives the greatest lower bound of the two partitions.

Another, more interesting, way to combine two MRRMs is by first applying the permutation method of one to the input network G^*, and then applying the second permutation method to the outputs of the first. This defines a composition of the two permutation methods.

Definition I.12. *Composition of randomized reference models.* Consider two MRRMs $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ and an input network $G^* \in \mathcal{G}$. The (sequential) composition of $\mathrm{P}[\mathbf{y}]$ on $\mathrm{P}[\mathbf{x}]$, denoted $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$, is defined by the conditional probability:

$$\begin{aligned} P_{\mathbf{y}\circ\mathbf{x}}(G|G^*) &= \sum_{G'\in\mathcal{G}} P_{\mathbf{y}}(G|G')P_{\mathbf{x}}(G'|G^*) \\ &= \sum_{G'\in\mathcal{G}} \frac{\delta_{\mathbf{y}(G),\mathbf{y}(G')}}{\Omega_{\mathbf{y}}(G)} \frac{\delta_{\mathbf{x}(G'),\mathbf{x}^*}}{\Omega_{\mathbf{x}^*}} \ . \end{aligned} \qquad (7)$$

For a given indexing of the state space $\mathcal{G}$, Eqs. (5) and (7) show that the transition matrix for the composition of $\mathrm{P}[\mathbf{y}]$ on $\mathrm{P}[\mathbf{x}]$ is simply the matrix product of the individual transition matrices, $\mathbf{P}^{\mathbf{y}\circ\mathbf{x}} = \mathbf{P}^{\mathbf{y}}\mathbf{P}^{\mathbf{x}}$.

Definition I.13. *Compatibility.* We say that two MRRMs $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ are *compatible* if their composition $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$ is also a MRRM.

The notion of compatibility is central as it defines which MRRMs we may combine through composition to define a new MRRM.

Proposition I.2. *Compatible randomized reference models commute.* If two MRRMs, $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$, are compatible then $\mathrm{P}[\mathbf{x}]\mathrm{P}[\mathbf{y}] = \mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$.

Proof. This can be shown by direct calculation by noting that the transition matrices of MRRMs are symmetric. For any two compatible MRRMs, $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$, their compositions $\mathrm{P}[\mathbf{x}]\mathrm{P}[\mathbf{y}]$ and $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$ both define a MRRM. So the associated transition matrices must be symmetric (Def. I.8), and

$$\mathbf{P}^{\mathbf{y}}\mathbf{P}^{\mathbf{x}} = (\mathbf{P}^{\mathbf{y}}\mathbf{P}^{\mathbf{x}})^{\intercal} = (\mathbf{P}^{\mathbf{x}})^{\intercal}(\mathbf{P}^{\mathbf{y}})^{\intercal} = \mathbf{P}^{\mathbf{x}}\mathbf{P}^{\mathbf{y}} \ . \qquad (8)$$

□

Proposition I.2 means that it does not matter in which order we apply two compatible MRRMs in the composition, and consequently that $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{x}]\mathrm{P}[\mathbf{y}]$ define the same MRRM if $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ are compatible. It also means that in order to show that two MRRMs are not compatible, it suffices to show that they do not commute.

Example I.6. Let the state space $\mathcal{G}$ be all static graphs with 3 nodes. We number the 8 states such that G_1 is the graph with 0 links, G_2 , G_3 , and G_4 are the states with 1 link, G_5 , G_6 , and G_7 are the states with 2 links and G_8 is the state with 3 links. Let us now define two MRRMs for this state space:

1. The MRRM $\mathrm{P}[L]$, corresponding to the Erdős-Rényi random graph model $G(3,L)$, is defined by the feature L which returns the number of edges in the network, and partitions the state space in 4 sets $\mathcal{G}_0 = \{G_1\}$, $\mathcal{G}_1 = \{G_2, G_3, G_4\}$, $\mathcal{G}_2 = \{G_5, G_6, G_7\}$, and $\mathcal{G}_3 = \{G_8\}$.

2. The MRRM $\mathrm{P}[\delta_{G,G_4}]$ keeps the state G_4 and shuffles all the others. It is defined by the feature δ_{G,G_4} which returns 1 if the state is G_4 and 0 for all other states. The MRRM partitions the state space into two partitions $\mathcal{G}_1 = \{G_4\}$ and $\mathcal{G}_0 = \{G_1, G_2, G_3, G_5, G_6, G_7, G_8\}$.

With these definitions $\mathrm{P}[L]\mathrm{P}[\delta_{G,G_4}] \neq \mathrm{P}[\delta_{G,G_4}]\mathrm{P}[L]$; for example, for G_4 the target space of $\mathrm{P}[L]\mathrm{P}[\delta_{G,G_4}]$ is $\{G_2, G_3, G_4\}$, while the target space of $\mathrm{P}[\delta_{G,G_4}]\mathrm{P}[L]$ is the entire $\mathcal{G}$. So the two MRRMs do not commute and are thus not compatible. Consequently, the ensembles obtained by composition of $\mathrm{P}[\delta_{G,G_4}]$ and $\mathrm{P}[L]$ are not microcanonical. It is in the above example also easy to verify that the states generated by $\mathrm{P}[\delta_{G,G_4}]\mathrm{P}[L]$ on G_4 are not equiprobable ($P_{\delta_{G,G_4}\circ L}(G_4) = 1/3$ while $P_{\delta_{G,G_4}\circ L}(G_i) = 2/21$ for all other states).

2. *Comparability and compatibility*

In order for the concept of compatibility to be practically useful we need to be able to find out which MRRMs are compatible and what the result of their compositions is. Comparable MRRMs are an easy special case in this regard, as all comparable MRRMs turn out to be compatible and their composition simply yields the MRRM that shuffles more.

Proposition I.3. *Comparable microcanonical randomized reference models are compatible.* Let $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ be two MRRMs and $\mathrm{P}[\mathbf{x}] \leq \mathrm{P}[\mathbf{y}]$. Then they are compatible and their composition gives $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}] = \mathrm{P}[\mathbf{y}]$.

Proof. Since $\mathbf{y}$ is a function of $\mathbf{x}$, its value is the same for all G which satisfy $\mathbf{x}(G) = \mathbf{x}^*$ and equals $\mathbf{y}^* = \mathbf{y}(G^*)$. This means that Eq. (7) reduces to

$$\begin{aligned} P_{\mathbf{y}\circ\mathbf{x}}(G|\mathbf{x}^*) &= P_{\mathbf{y}}(G|\mathbf{y}^*) \sum_{G' \in \mathcal{G}_{\mathbf{x}^*}} P_{\mathbf{x}}(G'|\mathbf{x}^*) \\ &= P_{\mathbf{y}}(G|\mathbf{y}^*) \ , \end{aligned} \tag{9}$$

where the second equality is obtained from the requirement that $P_{\mathbf{x}}$ must be normalized on $\mathcal{G}_{\mathbf{x}^*}$. So $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}] = \mathrm{P}[\mathbf{y}]$, which is a MRRM, showing that $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ are compatible. □

Example I.7. Consider again the MRRMs $\mathrm{P}[L]$ and $\mathrm{P}[\mathbf{k}]$ from Example I.4. Since they are comparable, they are compatible according to Proposition I.3. Consequently they commute (Proposition I.2) and $\mathrm{P}[\mathbf{k}]\mathrm{P}[L] = \mathrm{P}[L]\mathrm{P}[\mathbf{k}] = \mathrm{P}[L]$.

Proposition I.4. *Composition of two compatible MRRMs always results in a MRRM which does not shuffle less.* Consider two compatible MRRMs, $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$. Their composition, $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$, is coarser (or equal) than both $\mathrm{P}[\mathbf{y}]$ and $\mathrm{P}[\mathbf{x}]$, i.e. $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}] \geq \mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}] \geq \mathrm{P}[\mathbf{y}]$, even if $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ are not comparable.

Proof. Since $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ are compatible, $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$ is a MRRM by the definition of compatibility (Def. I.13). Since G itself is by definition in the target set of any MRRM applied to G, the target set of $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}]$ can never be smaller than the target set of $\mathrm{P}[\mathbf{x}]$. Thus, $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}] \geq \mathrm{P}[\mathbf{x}]$. Using that compatible MRRMs commute (Proposition I.2) gives that $\mathrm{P}[\mathbf{y}]\mathrm{P}[\mathbf{x}] \geq \mathrm{P}[\mathbf{y}]$. □

The effect of the composition operation seems to work in opposite manner to the intersection operation. This also reflects to the composition of zero and unity elements (which are compatible with all MRRMs by Proposition I.3) for which zero is the neutral element $\mathrm{P}[\mathbf{x}]\mathrm{P}[0] = \mathrm{P}[\mathbf{x}]$ and unity is the absorbing element $\mathrm{P}[1]\mathrm{P}[\mathbf{x}] = \mathrm{P}[1]$. Further, by Proposition I.4, the composition gives an upper bound for the two MRRMS. In fact, the bound is the least upper bound, and any set of compatible MRRMs forms a *lattice* [55], but this connection to the theory of partially ordered sets is not pursued further here.

3. Conditional independence and compatibility

Our aim is to be able to compose MRRMs to produce new ones, and even though comparable MRRMs are always compatible they are not useful for this purpose as they do not produce a new MRRM. There are more interesting compositions, but in order to be able to access these we need a way of characterizing which pairs of MRRMs are compatible outside of comparable ones. We will next define the concept of conditional independence between two features given a common coarsening of these and show that it is equivalent to compatibility.

Before we can define this concept, we first need to define the conditional probability and conditional independence of a feature.

Definition I.14. *Conditional probability of a feature.* The conditional probability of a feature $\mathbf{y}$ given another feature $\mathbf{x}$ is the probability $P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*)$ that the feature $\mathbf{y}$ takes the value $\mathbf{y}^\dagger$ conditioned on the value $\mathbf{x}^*$ of the feature $\mathbf{x}$. It is given by

$$\begin{aligned} P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*) &= \sum_{G' \in \mathcal{G}} \delta_{\mathbf{y}^\dagger, \mathbf{y}(G')} P_{\mathbf{x}}(G'|\mathbf{x}^*) \\ &= \frac{\Omega_{(\mathbf{y}^\dagger, \mathbf{x}^*)}}{\Omega_{\mathbf{x}^*}} \ . \end{aligned} \tag{10}$$

The conditional probability of a feature satisfies all properties of usual conditional probabilities. We may notably relate the composition of two MRRMs to the conditional probability of their features using the law of total probability as $P_{\mathbf{y}\circ\mathbf{x}}(G|\mathbf{x}^*) = \sum_{\mathbf{y}^\dagger} P_{\mathbf{y}}(G|\mathbf{y}^\dagger) P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*)$ (see the proof of Theorem 1). It also allows us to define the conditional independence in the usual sense as when $P_{\mathbf{y}|\mathbf{x},\mathbf{z}}(\mathbf{y}^\dagger|\mathbf{x}^\ddagger, \mathbf{z}^*) = P_{\mathbf{y}|\mathbf{z}}(\mathbf{y}^\dagger|\mathbf{z}^*)$ for a given a third feature $\mathbf{z}$. We shall here be concerned with a stricter version of conditional independence which is satisfied when the feature $\mathbf{z}$ is coarser than both $\mathbf{y}$ and $\mathbf{x}$. As we show below, this *conditional independence given a common coarsening* is equivalent to $\mathbf{x}$ and $\mathbf{y}$ being compatible.

Definition I.15. *Conditional independence given a common coarsening.* If there exist a feature $\mathbf{z}$ that is a common coarsening of $\mathbf{x}$ and $\mathbf{y}$, i.e. $\mathbf{z} \geq \mathbf{x}$ and $\mathbf{z} \geq \mathbf{y}$, such that $P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}(G^*)) = P_{\mathbf{y}|\mathbf{z}}(\mathbf{y}^\dagger|\mathbf{z}(G^*))$ for all $G^* \in \mathcal{G}$, we will say that $\mathbf{y}$ is conditionally independent of $\mathbf{x}$ given their common coarsening $\mathbf{z}$.

As for the usual conditional independence, the conditional independence given a common coarsening defined above is symmetric in $\mathbf{x}$ and $\mathbf{y}$, we show this below.

Proposition I.5. *Symmetry of the conditional independence given a common coarsening.* If $\mathbf{x}$ is conditionally independent of $\mathbf{y}$ given a common coarsening $\mathbf{z}$ then $\mathbf{y}$ is conditionally independent of $\mathbf{x}$ given $\mathbf{z}$

Proof. We note that since $\mathbf{z}$ is coarser than $\mathbf{x}$, it follows that $P_{\mathbf{y}|\mathbf{x}} = P_{\mathbf{y}|\mathbf{x},\mathbf{z}}$ (since $\mathcal{G}_{\mathbf{x}^*} \subseteq \mathcal{G}_{\mathbf{z}^*}$, conditioning only on $\mathbf{x}^*$ is equivalent to conditioning on both $\mathbf{x}^*$ and $\mathbf{z}^*$), and equivalently that $P_{\mathbf{x}|\mathbf{y}} = P_{\mathbf{x}|\mathbf{y},\mathbf{z}}$. Thus, the symmetry of the conditional independence given a common coarsening follows directly from the symmetry of the traditional conditional independence. For completeness we demonstrate the symmetry of conditional independence below.

Consider that $\mathbf{x}$ is independent of $\mathbf{y}$ conditioned on $\mathbf{z}$, i.e. $P_{\mathbf{x}|\mathbf{z}}(\mathbf{x}^\dagger|\mathbf{z}^*) = P_{\mathbf{x}|\mathbf{y},\mathbf{z}}(\mathbf{x}^\dagger|\mathbf{y}^\ddagger, \mathbf{z}^*)$. To show that this

implies the symmetric relation, we write out the relation using Eq. (10):

$$\begin{aligned}\frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}}{\Omega_{\mathbf{z}^*}} &= \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{y}^\ddagger,\mathbf{z}^*)}}{\Omega_{(\mathbf{y}^\ddagger,\mathbf{z}^*)}} \\ &= \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{y}^\ddagger,\mathbf{z}^*)}}{\Omega_{(\mathbf{y}^\ddagger,\mathbf{z}^*)}} \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}}{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}} \frac{\Omega_{\mathbf{z}^*}}{\Omega_{\mathbf{z}^*}} \\ &= \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{y}^\ddagger,\mathbf{z}^*)}}{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}} \frac{\Omega_{\mathbf{z}^*}}{\Omega_{(\mathbf{y}^\ddagger,\mathbf{z}^*)}} \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}}{\Omega_{\mathbf{z}^*}} \\ &= \frac{P_{\mathbf{y}|\mathbf{x},\mathbf{z}}(\mathbf{y}^\ddagger|\mathbf{x}^\dagger,\mathbf{z}^*)}{P_{\mathbf{y}|\mathbf{z}}(\mathbf{y}^\ddagger|\mathbf{z}^*)} \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}}{\Omega_{\mathbf{z}^*}} \ . \end{aligned} \tag{11}$$

This relation (which must be true) is only satisfied if $P_{\mathbf{y}|\mathbf{x},\mathbf{z}}(\mathbf{y}^\ddagger|\mathbf{x}^\dagger,\mathbf{z}^*) = P_{\mathbf{y}|\mathbf{z}}(\mathbf{y}^\ddagger|\mathbf{z}^*)$, i.e. if $\mathbf{y}$ is independent of $\mathbf{x}$ conditioned on $\mathbf{z}$, thus completing the proof. □

Because of the symmetry, we can simply say that $\mathbf{x}$ and $\mathbf{y}$ are conditionally independent given the common coarsening $\mathbf{z}$.

As we stated above, the concept of conditional independence given a common coarsening is important because it is a characterisation of compatibility. This will be used in Section III to show that certain important classes of MRRMs are compatible. It will also allow us to say certain things about which features are conserved by a composition of compatible MRRMs.

Theorem 1. *Conditional independence given a common coarsening is equivalent to compatilibity.* P[$\mathbf{x}$] and P[$\mathbf{y}$] are compatible if and only if they are conditionally independent. Further, they are conditionally independent given $\mathbf{z} = \mathbf{x} \circ \mathbf{y}$

Proof. To show that conditional independence given a common coarsening is equivalent to compatilibity, we first show that the former implies the latter and then that the latter implies the former. To avoid clutter, we will in the following use the notation $\mathbf{y}'$ as short for $\mathbf{y}(G')$ and, similarly, let $\mathbf{y}'' = \mathbf{y}(G'')$ and $\mathbf{x}'' = \mathbf{x}(G'')$.

Conditional independence given a common coarsening implies compatibility. We will use the law of total probability and Proposition I.3 to show this. We first show that a law of total probability applies to the probabilities of features. Taking the probability distribution defining the composition of P[$\mathbf{x}$] and P[$\mathbf{y}$] [Eq. (7)] and multiplying by the term $\sum_{\mathbf{y}^\dagger} \delta_{\mathbf{y}^\dagger,\mathbf{y}'} = 1$, we get:

$$\begin{aligned} P_{\mathbf{y}\circ\mathbf{x}}(G|\mathbf{x}^*) &= \sum_{G'\in\mathcal{G}} \sum_{\mathbf{y}^\dagger} \delta_{\mathbf{y}^\dagger,\mathbf{y}'} \frac{\delta_{\mathbf{y}(G),\mathbf{y}'}}{\Omega_{\mathbf{y}}(G)} P_{\mathbf{x}}(G'|\mathbf{x}^*) \\ &= \sum_{\mathbf{y}^\dagger} \frac{\delta_{\mathbf{y}(G),\mathbf{y}^\dagger}}{\Omega_{\mathbf{y}}(G)} \sum_{G'\in\mathcal{G}} \delta_{\mathbf{y}^\dagger,\mathbf{y}'} P_{\mathbf{x}}(G'|\mathbf{x}^*) \\ &= \sum_{\mathbf{y}^\dagger} P_{\mathbf{y}}(G|\mathbf{y}^\dagger) P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*) \ . \end{aligned} \tag{12}$$

To obtain the second equality above we used the property of Kronecker delta functions $\delta_{a,b}\delta_{b,c} = \delta_{a,c}\delta_{b,c}$, and the last equality was obtained from the definitions of $P_{\mathbf{y}}(G|\mathbf{y}^\dagger)$ (Def. I.7) and $P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*)$ (Def. I.14). Using the law of total probability, we now expand $P_{\mathbf{y}\circ\mathbf{x}}(G|\mathbf{x}^*)$ to get:

$$\begin{aligned} P_{\mathbf{y}\circ\mathbf{x}}(G|\mathbf{x}^*) &= \sum_{\mathbf{y}^\dagger} P_{\mathbf{y}}(G|\mathbf{y}^\dagger) P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*) \\ &= \sum_{\mathbf{y}^\dagger} P_{\mathbf{y}}(G|\mathbf{y}^\dagger) P_{\mathbf{y}|\mathbf{z}}(\mathbf{y}^\dagger|\mathbf{z}^*) \\ &= P_{\mathbf{y}\circ\mathbf{z}}(G|\mathbf{z}^*) \\ &= P_{\mathbf{z}}(G|\mathbf{z}^*) \ . \end{aligned} \tag{13}$$

Here, the second equality follows from the independence of $\mathbf{x}$ and $\mathbf{y}$ (Def. I.15), the second-to-last equality follows from the law of total probability, and the last from Proposition I.3 when noting that $\mathbf{z} \geq \mathbf{y}$.

Compatibility implies conditional independence given a common coarsening. Because $\mathbf{x}$ and $\mathbf{y}$ are compatible their composition is an MRRM and we can choose $\mathbf{z} = \mathbf{x} \circ \mathbf{y}$, which by construction is a common coarsening of $\mathbf{x}$ and $\mathbf{y}$. The conditional independence of $\mathbf{x}$ and $\mathbf{y}$ given this $\mathbf{z}$ can now be shown from its definition via a direct calculation:

$$\begin{aligned} P_{\mathbf{y}|\mathbf{x}\circ\mathbf{y}}(\mathbf{y}^\dagger|\mathbf{x}\circ\mathbf{y}(G^*)) &= \sum_{G'\in\mathcal{G}} \sum_{G''\in\mathcal{G}} \frac{\delta_{\mathbf{x}'',\mathbf{x}^*}\delta_{\mathbf{y}^\dagger,\mathbf{y}'}\delta_{\mathbf{y}',\mathbf{y}''}}{\Omega_{\mathbf{y}''}\Omega_{\mathbf{x}^*}} \\ &= \sum_{G''\in\mathcal{G}} \frac{\delta_{\mathbf{x}'',\mathbf{x}^*}\delta_{\mathbf{y}^\dagger,\mathbf{y}''}}{\Omega_{\mathbf{y}''}\Omega_{\mathbf{x}^*}} \sum_{G'\in\mathcal{G}} \delta_{\mathbf{y}',\mathbf{y}''} \\ &= \sum_{G''\in\mathcal{G}} \frac{\delta_{\mathbf{x}'',\mathbf{x}^*}\delta_{\mathbf{y}^\dagger,\mathbf{y}''}}{\Omega_{\mathbf{x}^*}} \\ &= P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*) \ . \end{aligned} \tag{14}$$

In the second equality we again used the property of Kronecker delta functions $\delta_{a,b}\delta_{b,c} = \delta_{a,c}\delta_{b,c}$, and in the second-to-last equality we used the definition of the partition function. □

We shall see (in Section III) that conditional independence will be important in practice for designing MRRMs that randomize both the topology and the time-domain of a temporal network by implementing them as compositions of MRRMs that randomize different levels in the two-level temporal network representations introduced in Section I A 1. The simple example below illustrates the concept of independence and how it leads to compatibility as well as the abstract nature of the problem.

Example I.8. Consider a state space with 9 states, $\mathcal{G} = \{G_1, \ldots, G_9\}$, which are placed into a square formation such that the states 1 to 3 are in the first row, 4 to 6 in the second row and 7 to 9 in the third row. Now we can define two features: f_r that returns the row number, and f_c that returns the column number. The partitions these two features induce are $\mathcal{G}_{f_r} = \{\{G_1, G_2, G_3\}, \{G_4, G_5, G_6\}, \{G_7, G_8, G_9\}\}$ for f_r and $\mathcal{G}_{f_c} = \{\{G_1, G_4, G_7\}, \{G_2, G_5, G_8\}, \{G_3, G_6, G_9\}\}$ for f_c. The two features f_r and f_c are conditionally independent given 1 (the unity element, i.e., a constant

function). Thus, the corresponding MRRMs, $\mathrm{P}[f_r]$ and $\mathrm{P}[f_c]$, are compatible and their composition is the unity element $\mathrm{P}[f_r]\mathrm{P}[f_c] = \mathrm{P}[f_c]\mathrm{P}[f_r] = \mathrm{P}[1]$, which shuffles everything. Indeed, direct computation shows that their composition is the unity element $\mathbf{P}^{f_r}\mathbf{P}^{f_c} = \mathbf{P}^{1}$.

The above example illustrates the abstract nature of the problem of combining MRRMs. In terms of temporal networks, $\mathcal{G}$ may be thought of as the space of all networks consisting of 3 nodes and a single event that takes place during one of three possible snapshots. The two features may be identified with the features G^{stat} returning the static network and $p_{\mathcal{L}}(\mathbf{\Theta})$ returning the number events on each link and their timings. If we let $f_r = G^{\text{stat}}$ and $f_c = p_{\mathcal{L}}(\mathbf{\Theta})$, then the row number determines the placement of the link in the static network and the column number the snapshot during which the event takes place.

Our main aim when defining compositions has been to be able to produce new useful MRRMs. With the help of the concept of conditional independence we are now ready to write down a theorem, and a proof, that will allow us to compose non-comparable MRRMs and know the features of the resulting model.

Theorem 2. *Combining refinements of compatible MRRMs.* Consider two compatible MRRMs $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$, and any two MRRMs, $\mathrm{P}[\mathbf{y}, \mathbf{f}(\mathbf{x})]$ and $\mathrm{P}[\mathbf{x}, \mathbf{g}(\mathbf{y})]$, that shuffle less. Then $\mathrm{P}[\mathbf{y}, \mathbf{f}(\mathbf{x})]$ and $\mathrm{P}[\mathbf{x}, \mathbf{g}(\mathbf{y})]$ are compatible, and their composition is given by $\mathrm{P}[\mathbf{x} \circ \mathbf{y}, \mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})]$.

Proof. To prove the above theorem, it is sufficient to prove that $(\mathbf{y}, \mathbf{f}(\mathbf{x}))$ and $\mathbf{x}$ are independent conditioned on their common coarsening $(\mathbf{y} \circ \mathbf{x}, \mathbf{f}(\mathbf{x}))$. From this it then immediately follows that $(\mathbf{y}, \mathbf{f}(\mathbf{x}))$ and $(\mathbf{x}, \mathbf{g}(\mathbf{y}))$ are independent conditioned on their common coarsening $(\mathbf{y} \circ \mathbf{x}, \mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y}))$ and thus that $\mathrm{P}[\mathbf{y},\mathbf{f}(\mathbf{x})]$ and $\mathrm{P}[\mathbf{x},\mathbf{g}(\mathbf{y})]$ are compatible with $\mathrm{P}[\mathbf{y}, \mathbf{f}(\mathbf{x})]\mathrm{P}[\mathbf{x}, \mathbf{g}(\mathbf{y})] = \mathrm{P}[\mathbf{y} \circ \mathbf{x}, \mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})]$. To develop the proof, we consider the conditional probability of $\mathbf{x}$ given $(\mathbf{y}, \mathbf{f}(\mathbf{x}))$:

$$P_{\mathbf{x}|\mathbf{y},\mathbf{f}(\mathbf{x})}(\mathbf{x}^\dagger|\mathbf{y}^*, \mathbf{f}(\mathbf{x}^*)) = \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{y}^*,\mathbf{f}(\mathbf{x}^*))}}{\Omega_{(\mathbf{y}^*,\mathbf{f}(\mathbf{x}^*))}} \ . \tag{15}$$

Since $\mathbf{x} \leq \mathbf{f}(\mathbf{x})$, we have $\Omega_{(\mathbf{x}^\dagger,\mathbf{y}^*,\mathbf{f}(\mathbf{x}^*))} = \Omega_{(\mathbf{x}^\dagger,\mathbf{y}^*)}$ whenever $\mathbf{f}(\mathbf{x}^\dagger) = \mathbf{f}(\mathbf{x}^*)$. We use this to rewrite the equation above, multiplying by the factor $\Omega_{\mathbf{y}^*}/\Omega_{\mathbf{y}^*} = 1$, along the way,

$$\begin{aligned} P_{\mathbf{x}|\mathbf{y},\mathbf{f}(\mathbf{x})}(\mathbf{x}^\dagger|\mathbf{y}^*, \mathbf{f}(\mathbf{x}^*)) &= \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{y}^*)}\delta_{\mathbf{f}(\mathbf{x}^\dagger),\mathbf{f}(\mathbf{x}^*)}}{\Omega_{\mathbf{y}^*}} \frac{\Omega_{\mathbf{y}^*}}{\Omega_{(\mathbf{y}^*,\mathbf{f}(\mathbf{x}^*))}} \\ &= \frac{P_{\mathbf{x}|\mathbf{y}}(\mathbf{x}^\dagger|\mathbf{y}^*)\delta_{\mathbf{f}(\mathbf{x}^\dagger),\mathbf{f}(\mathbf{x}^*)}}{P_{\mathbf{f}(\mathbf{x})|\mathbf{y}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{y}^*)} \ . \end{aligned} \tag{16}$$

Now, since $\mathbf{x}$ and $\mathbf{y}$ are compatible, we have that $P_{\mathbf{x}|\mathbf{y}}(\mathbf{x}^\dagger|\mathbf{y}^*) = P_{\mathbf{x}|\mathbf{z}}(\mathbf{x}^\dagger|\mathbf{z}^*)$, with $\mathbf{z} = \mathbf{y} \circ \mathbf{x}$. Furthermore, it also means that $P_{\mathbf{f}(\mathbf{x})|\mathbf{y}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{y}^*) = P_{\mathbf{f}(\mathbf{x})|\mathbf{z}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{z}^*)$. The following calculation shows this:

$$\begin{aligned} P_{\mathbf{f}(\mathbf{x})|\mathbf{y}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{y}^\dagger) &= \sum_{\mathbf{x}^\ddagger} P_{\mathbf{f}(\mathbf{x})|\mathbf{x}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{x}^\ddagger) P_{\mathbf{x}|\mathbf{y}}(\mathbf{x}^\ddagger|\mathbf{y}^\dagger) \\ &= \sum_{\mathbf{x}^\ddagger} P_{\mathbf{f}(\mathbf{x})|\mathbf{x}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{x}^\ddagger) P_{\mathbf{x}|\mathbf{z}}(\mathbf{x}^\ddagger|\mathbf{z}^\dagger) \\ &= P_{\mathbf{f}(\mathbf{x})|\mathbf{z}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{z}^\dagger) \ . \end{aligned} \tag{17}$$

(Note that we do not necessarily have $\mathbf{f}(\mathbf{x}) \leq \mathbf{z}$, though.) Plugging these two identities into Eq. (16) gives:

$$\begin{aligned} P_{\mathbf{x}|\mathbf{y},\mathbf{f}(\mathbf{x})}(\mathbf{x}^\dagger|\mathbf{y}^*, \mathbf{f}(\mathbf{x}^*)) &= \frac{P_{\mathbf{x}|\mathbf{z}}(\mathbf{x}^\dagger|\mathbf{z}^*)\delta_{\mathbf{f}(\mathbf{x}^\dagger),\mathbf{f}(\mathbf{x}^*)}}{P_{\mathbf{f}(\mathbf{x})|\mathbf{z}}(\mathbf{f}(\mathbf{x}^*)|\mathbf{z}^*)} \\ &= \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*)}\delta_{\mathbf{f}(\mathbf{x}^\dagger),\mathbf{f}(\mathbf{x}^*)}}{\Omega_{\mathbf{z}^*}} \frac{\Omega_{\mathbf{z}^*}}{\Omega_{(\mathbf{z}^*,\mathbf{f}(\mathbf{x}^*))}} \\ &= \frac{\Omega_{(\mathbf{x}^\dagger,\mathbf{z}^*,\mathbf{f}(\mathbf{x}^*))}}{\Omega_{(\mathbf{z}^*,\mathbf{f}(\mathbf{x}^*))}} \\ &= P_{\mathbf{x}|\mathbf{z},\mathbf{f}(\mathbf{x})}(\mathbf{x}^\dagger|\mathbf{z}^*, \mathbf{f}(\mathbf{x}^*)) \ . \end{aligned} \tag{18}$$

Since both $\mathbf{z}^* \geq \mathbf{x}$ and $\mathbf{f}(\mathbf{x}^*) \geq \mathbf{x}$, we have $(\mathbf{z}, \mathbf{f}(\mathbf{x})) \geq \mathbf{x}$, which together with Eq. (18) shows that $\mathbf{x}$ is independent of $(\mathbf{y}, \mathbf{f}(\mathbf{x}))$ conditionally on their common coarsening $(\mathbf{z}, \mathbf{f}(\mathbf{x}))$, thus completing the proof. □

The new MRRM created using Theorem 2 always carries the composition, $\mathbf{x} \circ \mathbf{y}$, of the two lower bounding MRRMs in it. It would be desirable to be able to create MRRMs that only contain some combinations of prescribed features within it. This is in principle possible if the two bounding MRRMs are chosen such that they are fully independent, as shown by the following definition and corollary.

Definition I.16. *Independent MRRMs.* $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$ are said to be independent when $P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*) = P_{\mathbf{y}}(\mathbf{y}^\dagger)$, where $P_{\mathbf{y}}(\mathbf{y}^\dagger) = P_{\mathbf{y}|1}(\mathbf{y}^\dagger|1)$.

Corollary I.2. *Combining refinements of independent MRRMs.* Consider two independent MRRMs $\mathrm{P}[\mathbf{x}]$ and $\mathrm{P}[\mathbf{y}]$, and two MRRMs, $\mathrm{P}[\mathbf{y}, \mathbf{f}(\mathbf{x})]$ and $\mathrm{P}[\mathbf{x}, \mathbf{g}(\mathbf{y})]$, that shuffle less. Then $\mathrm{P}[\mathbf{y}, \mathbf{f}(\mathbf{x})]\mathrm{P}[\mathbf{x}, \mathbf{g}(\mathbf{y})] = \mathrm{P}[\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})]$.

Proof. Because independence is a special case of conditional independence the proof is a direct application of Theorem 2: $\mathrm{P}[\mathbf{y}, \mathbf{f}(\mathbf{x})]\mathrm{P}[\mathbf{x}, \mathbf{g}(\mathbf{y})] = \mathrm{P}[\mathbf{x} \circ \mathbf{y}, \mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})] = \mathrm{P}[1, \mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})] = \mathrm{P}[\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})]$. □

II. FEATURES OF TEMPORAL NETWORKS

A typical goal when employing MRRMs is to investigate how given predefined features of temporal network constrain other features of the network or alternatively how they affect a dynamical process unfolding on the network. In this section we define and list a selection of temporal network features that have been shown to play an important role in network dynamics, and which are sufficient to name the MRRMs found in our literature survey and presented in Section IV below.

A feature of a temporal network is any function that takes a network as an input. Clearly, there is a very large number of such functions which could be defined, and as we will see later, a multitude of such functions have been

(often implicitly) used in the literature. Here we attempt to organize this set of functions in a way that it is compatible with the different temporal network representations introduced in Section I and the concept of order of features given by Definition I.9 and Proposition I.1.

Many features are ones returning a sequence of lower-dimensional features, e.g., the degree sequence of a static graph, $\mathbf{k}$, is given by the sequence of the individual node degrees k_i. Temporal network features are often given by a nested sequence where individual features in the sequence themselves are a function of a sequence of scalar features. Sequences and nested sequences can further be turned into distributions and average values in multiple ways.

We begin by introducing the ordered sequence of a collection of features. It retains both the values of the individual features and what they designate in the network. A MRRM that constrains such a sequence thus produces reference networks with exactly the same values and configuration of these features as in the input network. In order to make the notation simpler, and without loss of generality, we will assume that features returning values of multiple named entities, such as nodes or links, return them as sequences that have an arbitrary but fixed order.

Definition II.1. *Sequence of features.* A *sequence* of features is a tuple $\mathbf{x} = (\mathbf{x}_q)_{q\in\mathcal{Q}}$ of individual feature values ordered according to an arbitrary but fixed index $q \in \mathcal{Q}$, where each individual feature $\mathbf{x}_q$ may itself in general be a sequence of features.

Since we typically have no reason to prefer any single element or part of a network over the others, we generally want a sequence of features to cover the entire network G. For example, for features of nodes the index would run over all the nodes, i.e. $\mathcal{Q} = \mathcal{V}$; for features of links in the link-timeline network, $\mathcal{Q} = \mathcal{L}$; and for features of snapshot graphs, $\mathcal{Q} = \mathcal{T}$. Note that the index set $\mathcal{Q}$ may be an unordered set (e.g. the nodes or links), but in this case we fix an arbitrary order.

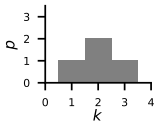
The basic building blocks of many features constructed as sequences and used in MRRMs are scalar features describing single elements of the network. Due to their importance and for ease of reference, we denote these as *characteristics*.

Definition II.2. *Characteristic.* A temporal network *characteristic* is a scalar function x pertaining to a single element of a network, such as a node $i \in \mathcal{V}$, a link $(i,j) \in \mathcal{L}$, a snapshot in time $t \in \mathcal{T}$, an event $c \in \mathcal{C}$, or both a point in time and a node or a link.

We shall use a subscript to index topological characteristics (e.g. for nodes: x_i or links: $x_{(i,j)}$) and superscript for temporal ones (e.g. for snapshots: x^t); for sequences of sequences, individual characteristics have both a subscript and a superscript index (e.g. x_i^m or $x_{(i,j)}^m$, where m refers to a temporal ordering).

Table I lists important features and, in particular, characteristics of temporal networks.

$(2,3,2,1)$



2

(a) Sequence of static degrees $\mathbf{k}$. (b) Distribution of static degrees $p(\mathbf{k})$. (c) Mean static degree $\mu(\mathbf{k})$.

FIG. 3: **Example: Marginals and moments of one-level sequence.** (a) Sequence, (b) distribution, and (c) mean of the static degrees $\mathbf{k}^*$ of the network shown in Fig. 1.

With sequences as a construction rule and characteristics as basic building blocks, we can now design many features which are constrained by MRRMs found in the literature. The simplest class of such features is the *one-level* sequence of characteristics.

Definition II.3. *One-level sequence of characteristics.* A *one-level* sequence is a sequence of single characteristics (i.e. scalar features): $\mathbf{x} = (x_q)_{q\in\mathcal{Q}}$.

A one-level sequence is typically a sequence of aggregated network characteristics: for a set of time-aggregated characteristics of the nodes or links, the one-level sequence is given as $\mathbf{x} = (x_i)_{i\in\mathcal{V}}$ or $\mathbf{x} = (x_{(i,j)})_{(i,j)\in\mathcal{L}}$, respectively; for a set of aggregated characteristics of snapshots, e.g., the cumulative activity A^t of all nodes at each time t, the one-level sequence is of the form $\mathbf{x} = (x^t)_{t\in\mathcal{T}}$.

Example II.1. A well known example of an aggregated graph characteristic is the node degree, k_i, giving the number of nodes in G^{stat} that are connected to the node i. Figure 3(a) shows the (*one-level*) sequence of static degrees $\mathbf{k}^* = (k_i^*)_{i\in\mathcal{V}}$ of the temporal network shown in Fig. 1.

Definition II.4. *Two-level sequence of characteristics.* A *two-level* sequence $\mathbf{x}$ is a sequence of vector-valued features, where each individual feature itself is a sequence of characteristics: $\mathbf{x} = (\mathbf{x}_q)_{q\in\mathcal{Q}}$, with $\mathbf{x}_q = (x_q^r)_{r\in\mathcal{R}_q}$. We call each $\mathbf{x}_q$ a *local* sequence.

A two-level sequence is typically composed of characteristics that depend both on topology and time. Examples of such topological-temporal characteristics are the *instantaneous* degree of a given node at a given time and the *inter-event duration* between two successive events on a link (see examples below). More generally, for a set of time-dependent characteristic of links, their sequence is given by $\mathbf{x} = (\mathbf{x}_{(i,j)})_{(i,j)\in\mathcal{L}} = ((x_{(i,j)}^m)_{m\in\mathcal{M}_{(i,j)}})_{(i,j)\in\mathcal{L}}$. For time-dependent characteristics of nodes, their sequence is given by $\mathbf{x} = (\mathbf{x}_i)_{i\in\mathcal{V}} = ((x_i^m)_{m\in\mathcal{M}_i})_{i\in\mathcal{V}}$.

Example II.2. A generalization of the static degree k_i of a node is the *instantaneous degree* d_i^t. It is given by

TABLE I: Features of temporal networks. Below, "$(\cdot)$" denotes a sequence, "$\{\cdot\}$" denotes a set, "$|\cdot|$" denotes the cardinality of a set, ":" means *for which*' or *such that*, and "$\exists$" is short for *there exists*.

Symbol	*Meaning of symbol*	*Definition*
General definitions		
$[t_{\min}, t_{\max}]$	Period of observation.	
G	Temporal network.	$G = (\mathcal{V}, \mathcal{C})$ (Def. I.1)
G	Instant-event temporal network.	$G = (\mathcal{V}, \mathcal{E})$ (Def. I.2)
$\mathcal{V}$	Set of nodes in G.	$\mathcal{V} = \{1, 2, \ldots, N\}$
$\mathcal{C}$	Set of events in G.	$\mathcal{C} = \{c_1, c_2, \ldots, c_C\}$
$\mathcal{E}$	Set of reduced events.	$\mathcal{E} = \{e_1, e_2, \ldots, e_E\}$
c_q	qth event.	$c_q = (i_q, j_q, t_q, \tau_q)$
e_q	qth reduced event.	$e_q = (i_q, j_q, t_q)$
i, j	Node indices.	
i_q, j_q	Indices for nodes partaking in the qth event.	
t_q	Start time of the qth event.	
τ_q	Duration of the qth event.	
N	Number of nodes in G.	$N = \lvert\mathcal{V}\rvert$
C	Number of events in G.	$C = \lvert\mathcal{C}\rvert$
E	Number of reduced events in G.	$E = \lvert\mathcal{E}\rvert$
link-timeline representation		
$G_{\mathcal{L}}$	link-timeline network.	$G_{\mathcal{L}} = \left(G^{\text{stat}}, \boldsymbol{\Theta}\right)$ (Def. I.5)
G^{stat}	Static graph.	$G^{\text{stat}} = (\mathcal{V}, \mathcal{L})$
$\mathcal{L}$	Links in G^{stat}.	$\mathcal{L} = \{(i,j) : (i,j,t,\tau) \in \mathcal{C}\}$
L	Number of links in G^{stat}.	$L = \lvert\mathcal{L}\rvert$
$\mathcal{V}_i$	Neighborhood of node i.	$\{j : (i,j) \in \mathcal{L}\}$
$\boldsymbol{\Theta}$	Sequence of timelines.	$\boldsymbol{\Theta} = (\Theta_{(i,j)})_{(i,j)\in\mathcal{L}}$
$\Theta_{(i,j)}$	Link timeline.	$\Theta_{(i,j)} = \left(\left(t^1_{(i,j)}, \tau^1_{(i,j)}\right), \left(t^2_{(i,j)}, \tau^2_{(i,j)}\right), \ldots, \left(t^{n_{(i,j)}}_{(i,j)}, \tau^{n_{(i,j)}}_{(i,j)}\right)\right)$
Snapshot-sequence representation		
$G_{\mathcal{T}}$	Snapshot-graph sequence	$G_{\mathcal{T}} = (\mathcal{T}, \boldsymbol{\Gamma})$ (Def. I.6)
$\mathcal{T}$	Sequence of snapshot times.	$\mathcal{T} = (t_m)_{m=1}^{T}$
$\boldsymbol{\Gamma}$	Sequence of snapshot graphs.	$\boldsymbol{\Gamma} = (\Gamma^t)_{t\in\mathcal{T}}$
Γ^t	Snapshot graph at time t.	$\Gamma^t = (\mathcal{V}, \mathcal{E}^t)$
$\mathcal{E}^t$	Events at time t.	$\mathcal{E}^t = \{(i,j) : (i,j,t) \in \mathcal{E}\}$
Topological-temporal (two-level) characteristics		
$t^m_{(i,j)}$	Start time of mth event on timeline ℓ.	(Def. I.5)
$\tau^m_{(i,j)}$	Event duration.	(Def. I.5)
$\Delta\tau^m_{(i,j)}$	Inter-event duration.	$\Delta\tau^m_{(i,j)} = t^{m+1}_{(i,j)} - (t^m_{(i,j)} + \tau^m_{(i,j)})$
$t^w_{(i,j)}$	End time of the last event on timeline.	$t^w_{(i,j)} = t^{n_{(i,j)}}_{(i,j)} + \tau^{n_{(i,j)}}_{(i,j)}$
d^t_i	Instantaneous degree at time t.	$d^t_i = \lvert\{j : (i,j) \in \mathcal{E}^t\}\rvert$
v^m_i	Start time activity interval.	Start time of mth interval of consecutive activity of node i.
α^m_i	Activity duration.	Duration of mth interval of consecutive activity of node i.
$\Delta\alpha^m_i$	Inactivity duration.	$\Delta\alpha^m_i = v^{m+1}_i - (v^m_i + \alpha^m_i)$
Aggregated (one-level) characteristics		
$n_{(i,j)}$	Link contact frequency.	$n_{(i,j)} = \lvert\Theta_{(i,j)}\rvert$
$w_{(i,j)}$	Link cumulative duration.	$w_{(i,j)} = \sum_{m=1}^{n_{(i,j)}} \tau^m_{(i,j)}$
a_i	Node activity.	$a_i = \sum_{j\in\mathcal{V}_i} n_{(i,j)}$.
s_i	Node strength.	$s_i = \sum_{j\in\mathcal{V}_i} w_{(i,j)}$.
k_i	Node degree.	$k_i = \lvert\mathcal{V}_i\rvert$
A^t	Cumulative activity at time t.	$A^t = \sum_{i\in\mathcal{V}} d^t_i = 2\lvert\mathcal{E}^t\rvert$
Special features		
Φ_i	Node timeline.	$\Phi_i = \left((v^1_i, \alpha^1_i), (v^2_i, \alpha^2_i), \ldots, (v^{n^a_i}_i, \alpha^{n^a_i}_i)\right)$.
$\mathbb{I}_\lambda$	Indicator of connectedness of G^{stat}.	$\mathbb{I}_\lambda = 1$ if G^{stat} is connected, $\mathbb{I}_\lambda = 0$ elsewise.
$\text{iso}(\Gamma^t)$	Isomorphism class of Γ^t.	Set of graphs obtained by all permutations of node indices in Γ^t.

the number of nodes that the node i is in contact with at time t. The (two-level) sequence of instantaneous degrees is then $\mathbf{d} = (\mathbf{d}^t)_{t\in\mathcal{T}} = ((d_i^t)_{i\in\mathcal{V}})_{t\in\mathcal{T}}$, or alternatively $\mathbf{d} = (\mathbf{d}_i)_{i\in\mathcal{V}} = ((d_i^t)_{t\in\mathcal{T}})_{i\in\mathcal{V}}$ since the order of the indices i and t does not matter. Figure 4(a) shows the two level sequence of instantaneous degrees $\mathbf{d}^*$ of the temporal network shown in Figs. 1 and 2.

Example II.3. A feature of temporal networks that has been shown to have a profound impact on the propagation of dynamic processes is the durations between consecutive events in a timeline, termed the *inter-event durations* $\Delta\tau_{(i,j)}^m = t_{(i,j)}^{m+1} - (t_{(i,j)}^m + \tau_{(i,j)}^m)$. Their (two-level) sequence is $\boldsymbol{\Delta\tau} = (\boldsymbol{\Delta\tau}_{(i,j)})_{(i,j)\in\mathcal{L}}$, where $\boldsymbol{\Delta\tau}_{(i,j)} = (\Delta\tau_{(i,j)}^m)_{m\in\mathcal{M}_{(i,j)}}$. Here $\mathcal{M}_{(i,j)} = \{1, 2, \ldots, n_{(i,j)} - 1\}$ indexes the inter-event durations in the timeline $\Theta_{(i,j)}$ by temporal order, with $n_{(i,j)}$ the number of events in the timeline. Figure 5(a) shows the two level sequence of inter-event durations $\boldsymbol{\Delta\tau}^*$ of the temporal network shown in Figs. 1 and2.

Instead of constraining an ordered sequence itself, many MRRMs constrain marginal distributions or moments of a sequence. Before we define these marginals and moments in detail for temporal networks, we consider the simpler example of the degree sequence of a static graph.

Example II.4. From the sequence of degrees in a static graph, $\mathbf{k}$, we may calculate their marginal distribution $p(\mathbf{k})$ (equivalent to the multiset of their values, see Def. II.5 below), as well as their mean $\mu(\mathbf{k})$. This leads to three different features, each corresponding to a different MRRM: one that constrains the complete sequence of degrees, P[$\mathbf{k}$], one that constrains their distribution, P[$p(\mathbf{k})$], and one that constrains their mean P[$\mu(\mathbf{k})$] (which is equivalent to P[L] if the number of nodes N is kept constant). (P[$\mathbf{k}$] and P[$p(\mathbf{k})$] are both often referred to as the configuration model or the Maslov-Sneppen model, and P[$\mu(\mathbf{k})$] = P[L] is the Erdős-Rényi model with a fixed number of links.) The three features (and corresponding MRRMs) satisfy a linear order: $\mathbf{k} \leq p(\mathbf{k}) \leq \mu(\mathbf{k})$.

Since we have both a topological and temporal dimension in temporal networks, a much larger number of different ways to marginalize the sequence of characteristics is possible than for a simple static graph. We define the most important ones below and specify them for characteristics of links, nodes, and snapshots in Table II.

Definition II.5. *Distribution of characteristics.* Consider a one- or two-level sequence of characteristics, $\mathbf{x}$. The distribution of the characteristics, denoted $p(\mathbf{x})$, returns the number of times each possible scalar value ξ appears in a measured sequence $\mathbf{x}^* = \mathbf{x}(G^*)$. Formally, the distribution can be defined as the unnormed empirical measure that for each value ξ found in $\mathbf{x}^*$ assigns the value given by the function $\epsilon(\xi|\mathbf{x}^*) = \sum_{x^*\in\mathbf{x}^*} \delta_{\xi,x^*}$. Here the sum runs over all scalar elements in $\mathbf{x}^*$, i.e. over the single index for a one-level sequence and over both indices for a two-level sequence.

For the purpose of defining MRRMs, the distribution of characteristics in a sequence $\mathbf{x}^*$ is entirely equivalent to the multiset containing the values of all elements in $\mathbf{x}^*$ including duplicate values. We will use square brackets to denote the multiset and may then write $p(\mathbf{x}^*) = [x^*]_{x^*\in\mathbf{x}^*}$. This equivalence will be useful for defining distributions of more general multivariate features below and to represent them graphically in Figures 4 and 5.

Example II.5. Figure 3(b) shows the distribution of static degrees, $p(\mathbf{k}^*)$ of the temporal network of Figs. 1 and 2.

Example II.6. Figure 4(k) shows the distribution of the individual instantaneous degrees, $p(\mathbf{d}^*)$, of the temporal network of Figs. 1 and 2.

Example II.7. Figure 5(g) shows the distribution of individual inter-event durations, $p(\boldsymbol{\Delta\tau}^*)$, of the temporal network of Figs. 1 and 2.

We consider three additional ways to marginalize a two-level sequence which all result in features that are finer than the distribution of characteristics defined above. These are obtained as the sequence of the distributions of each local sequence (termed the *local distributions*), as the distribution of the local sequences, or as the distribution of local distributions. When needed to avoid ambiguity, we will refer to the distribution of characteristics as the *global* distribution to distinguish it from these three alternative marginalizations.

Definition II.6. *Sequence of local distributions of characteristics.* Consider a two-level sequence of characteristics, $\mathbf{x} = (\mathbf{x}_q)_{q\in\mathcal{Q}}$. The sequence of local distributions, denoted $\boldsymbol{\pi}_{\mathcal{Q}}(\mathbf{x})$, is given by the ordered tuple $\boldsymbol{\pi}_{\mathcal{Q}}(\mathbf{x}) = (\pi_q(\mathbf{x}_q))_{q\in\mathcal{Q}}$, where each local distribution $\pi_q(\mathbf{x}_q) = p(\mathbf{x}_q)$ is the distribution of the characteristics in the local sequence $\mathbf{x}_q$ as defined in Definition II.5. To avoid confusion between the global distribution p of a nested sequence $\mathbf{x}$ and the distributions of the local sequences $\mathbf{x}_q$, we use the different symbol π_q to designate the latter.

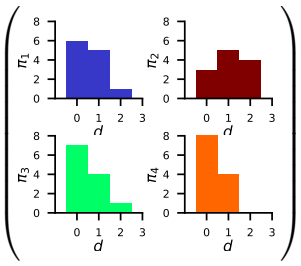
Example II.8. Figures 4(b) and 4(e) show the sequence of local distributions of the instantaneous degrees of each node, $\boldsymbol{\pi}_{\mathcal{V}}(\mathbf{d}^*)$, and of instantaneous degrees in each snapshot, $\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d}^*)$, respectively, of the temporal network of Figs. 1 and 2.

Example II.9. Figure 5(b) shows the sequence of local distributions of inter-event durations, $\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}^*)$, of the temporal network of Figs. 1 and 2.

Definition II.7. *Distribution of local sequences of characteristics.* The distribution of local sequences $p_{\mathcal{Q}}(\mathbf{x})$ reports the number of times each possible (vector-valued)

$$\begin{pmatrix} (0,2,0,0,1,0,1,1,1,0,1,0); \\ (0,2,1,1,1,1,0,1,2,2,2,0); \\ (0,2,0,0,0,1,1,0,1,1,0,0); \\ (0,0,1,1,0,0,0,0,0,1,1,0) \end{pmatrix}$$

(a) Sequence $\mathbf{d}$.



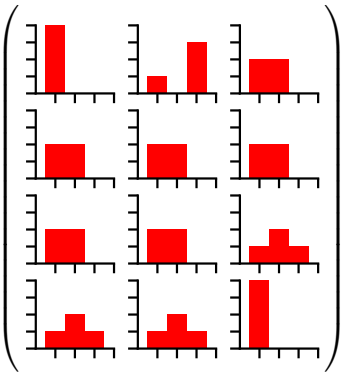
(b) Sequence of local distributions for each node, $\boldsymbol{\pi}_{\mathcal{V}}(\mathbf{d})$.

$$\begin{bmatrix} (0,2,0,0,1,0,1,1,1,0,1,0); \\ (0,2,1,1,1,1,0,1,2,2,2,0); \\ (0,2,0,0,0,1,1,0,1,1,0,0); \\ (0,0,1,1,0,0,0,0,0,1,1,0) \end{bmatrix}$$

(c) Distribution of local sequences on each node, $p_{\mathcal{V}}(\mathbf{d})$.

$$\begin{bmatrix} (0,0,0,0); (2,2,2,0); \\ (0,1,0,1); (0,1,0,1); \\ (1,1,0,0); (0,1,1,0); \\ (1,0,1,0); (1,1,0,0); \\ (1,2,1,0); (0,2,1,0); \\ (1,2,0,1); (0,0,0,0) \end{bmatrix}$$

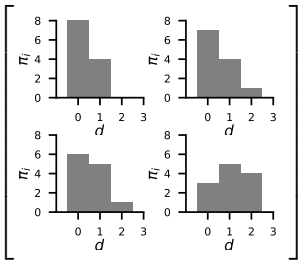
(d) Distribution of local sequences in each snapshot, $p_{\mathcal{T}}(\mathbf{d})$.



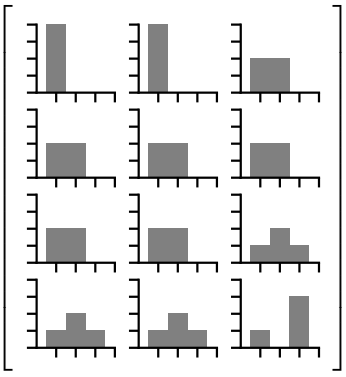
(e) Sequence of local distributions in each snapshot, $\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d})$.

$$\begin{pmatrix} 7/12, 13/12, \\ 1/2, 1/3 \end{pmatrix}$$

(f) Sequence of local means for each node, $\boldsymbol{\mu}_{\mathcal{V}}(\mathbf{d})$.



(g) Distribution of local distributions for each node, $p_{\mathcal{V}}(\boldsymbol{\pi}_{\mathcal{V}}(\mathbf{d}))$.



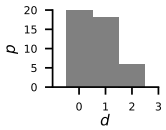
(h) Distribution of local distributions in each snapshot, $p_{\mathcal{T}}(\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d}))$.

$$\begin{pmatrix} 0, 3/2, 1/2, 1/2, 1/2, 1/2, \\ 1/2, 1/2, 1, 1, 1, 0 \end{pmatrix}$$

(i) Sequence of local means in each snapshot, $\boldsymbol{\mu}_{\mathcal{T}}(\mathbf{d})$.

$$\begin{bmatrix} 1/3, 1/2, \\ 7/12, 13/12 \end{bmatrix}$$

(j) Distribution of local means for nodes, $p_{\mathcal{V}}(\boldsymbol{\mu}_{\mathcal{V}}(\mathbf{d}))$.



(k) Global distribution, $p(\mathbf{d})$.

$$\begin{bmatrix} 0, 0, 1/2, 1/2, 1/2, 1/2, \\ 1/2, 1/2, 1, 1, 1, 3/2 \end{bmatrix}$$

(l) Distribution of local means in snapshots, $p_{\mathcal{T}}(\boldsymbol{\mu}_{\mathcal{T}}(\mathbf{d}))$.

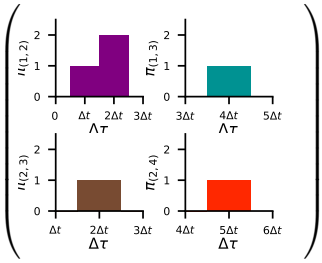
5/8

(m) Global mean, $\mu(\mathbf{d})$.

FIG. 4: **Example: Marginals and moments of the two-level sequence of instantaneous node degrees, d.** (a) Sequence and (b)–(m) marginals and moments of the instantaneous degrees $\mathbf{d}^*$ of the network shown in Figs. 1 and 2 . Distributions are shown as multisets whenever this is most convenient.

$$\begin{pmatrix} (2\Delta t, 2\Delta t, \Delta t); \\ (4\Delta t); \\ (2\Delta t); \\ (5\Delta t) \end{pmatrix}$$

(a) Sequence $\boldsymbol{\Delta\tau}$.



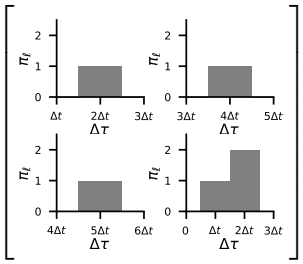
(b) Sequence of local distributions on each link, $\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$.

$$\begin{bmatrix} (2\Delta t); \\ (4\Delta t); \\ (5\Delta t); \\ (2\Delta t, 2\Delta t, \Delta t) \end{bmatrix}$$

(c) Distribution of local sequences on links, $p_{\mathcal{L}}(\boldsymbol{\Delta\tau})$.

$$\begin{pmatrix} 2\Delta t, 4\Delta t, \\ 5\Delta t, 5\Delta t/3 \end{pmatrix}$$

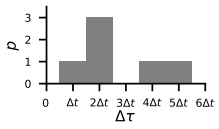
(d) Sequence of local means on each link, $\boldsymbol{\mu}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$.



(e) Distribution of local distributions on links, $p_{\mathcal{T}}(\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}))$.

$$\begin{bmatrix} 2\Delta t, 4\Delta t, \\ 5\Delta t, 5\Delta t/3 \end{bmatrix}$$

(f) Distribution of local means on links, $p_{\mathcal{L}}(\boldsymbol{\mu}_{\mathcal{L}}(\boldsymbol{\Delta\tau}))$.



(g) Global distribution, $p(\boldsymbol{\Delta\tau})$.

$$8\Delta t/3$$

(h) Global mean, $\mu(\boldsymbol{\Delta\tau})$.

FIG. 5: **Example: Marginals and moments of the two-level sequence of inter-event durations, $\boldsymbol{\Delta\tau}$.** (a) Sequence and (b)–(h) marginals and moments of the inter-event durations $\boldsymbol{\Delta\tau}$ for the network shown in Figs. 1 and 2. Distributions are shown as multisets whenever this is most convenient.

value $\boldsymbol{\xi}$ of a local sequence is observed in a two-level sequence $\mathbf{x}^* = (\mathbf{x}^*_q)_{q\in\mathcal{Q}}$. Formally, it is most simply defined as the multiset of local sequences, $p_{\mathcal{Q}}(\mathbf{x}) = [\mathbf{x}_q]_{q\in\mathcal{Q}}$.

Example II.10. Figure 4(c) shows the distribution of local sequences of the instantaneous degrees of each node, $p_{\mathcal{V}}(\mathbf{d}^*)$, and Fig. 4(d) shows the distribution of local sequences of instantaneous degrees of nodes in each snapshot, $p_{\mathcal{T}}(\mathbf{d}^*)$, for the temporal network shown of Figs. 1 and 2.

Example II.11. Figure 5(c) shows the distribution of local sequences of inter-event durations on the links, $p_{\mathcal{L}}(\boldsymbol{\Delta\tau}^*)$, of the temporal network of Figs. 1 and 2.

Definition II.8. *Distribution of local distributions of characteristics.* The distribution of local distributions $p_{\mathcal{Q}}(\boldsymbol{\pi}_{\mathcal{Q}})$ is obtained by marginalizing over the sequence of local distributions. We define $p_{\mathcal{Q}}(\boldsymbol{\pi}_{\mathcal{Q}})$ formally as a function that returns the multiset of the local distributions, $p_{\mathcal{Q}}(\boldsymbol{\pi}_{\mathcal{Q}}(\mathbf{x})) = [\pi_q(\mathbf{x}_q)]_{q\in\mathcal{Q}}$.

As the notation indicates, the distribution of local distributions $p_{\mathcal{Q}}(\boldsymbol{\pi}_{\mathcal{Q}})$ is the composition of the function $p_{\mathcal{Q}}$, which takes a sequence of sequences and returns the distribution if the individual (local) sequences, on $\boldsymbol{\pi}_{\mathcal{Q}}$, which returns the sequence of the local distributions of the individual sequences. This composition should not be confused with the compositions of MRRMs, which does not return a function.

Example II.12. Figure 4(g) shows the distribution of local distributions of the instantaneous degrees of each node, $p_{\mathcal{V}}(\boldsymbol{\pi}_{\mathcal{V}}(\mathbf{d}^*))$, and Fig. 4(h) shows the distribution of local distributions of instantaneous degrees of nodes in each snapshot, $p_{\mathcal{T}}(\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d}^*))$, for the temporal network shown of Figs. 1 and 2.

Example II.13. Figure 5(e) shows the distribution of local distributions of inter-event durations on the links, $p_{\mathcal{T}}(\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}^*))$, of the temporal network of Figs. 1 and 2.

After the above definitions of different local and global marginalizations of a one- or two-level sequence of characteristics, we now consider ways to define its moments. We shall here consider only first-order moments, i.e. means, but note that one may generally consider also higher order moments such as covariances.

Definition II.9. *Mean of a sequence of characteristics.* The mean $\mu(\mathbf{x})$ of a sequence of characteristics is the average over all individual values of the characteristics in $\mathbf{x}$. The mean is given by $\mu(\mathbf{x}^*) = \sum_{x^*\in\mathbf{x}^*} x^*/|\mathbf{x}^*|$, where the sum runs over all scalar elements in $\mathbf{x}^*$ and $|\mathbf{x}^*|$ is their number.

Example II.14. Figure 3(c) shows the mean static degree, $\mu(\mathbf{k}^*)$, of the temporal network in Fig. 1.

Example II.15. Figure 4(m) shows the mean instantaneous degree, $\mu(\mathbf{d}^*)$, of the temporal network of Figs. 1 and 2.

Example II.16. Figure 5(h) shows the mean inter-event duration, $\mu(\boldsymbol{\Delta\tau}^*)$, of the temporal network of Figs. 1 and 2.

As for the case of the distribution, the structure of a two-level sequence makes it natural to consider two additional features based on the means of local sequences. These are obtained as the sequence of the means of each local sequence (the *local means*) and as the distribution of the local means. When needed to avoid ambiguity, we will refer to the mean of characteristics as the *global* mean to distinguish it from those features.

Definition II.10. *Sequence of local means of characteristics.* The sequence of local means is calculated from a two-level sequence $\mathbf{x}^* = (\mathbf{x}^*_q)_{q\in\mathcal{Q}}$ as $\boldsymbol{\mu}_{\mathcal{Q}}(\mathbf{x}^*) = (\mu_q(\mathbf{x}^*_q))_{q\in\mathcal{Q}}$. Here each *local* mean is given by $\mu_q(\mathbf{x}^*_q) = \sum_{x^*\in\mathbf{x}^*_q} x^*/|\mathbf{x}^*_q|$, where $|\mathbf{x}^*_q|$ is the number of scalar elements in $\mathbf{x}^*_q$.

Example II.17. Figure 4(f) shows the sequence of local means of the instantaneous degrees of each node, $\boldsymbol{\mu}_{\mathcal{V}}(\mathbf{d}^*)$, and Fig. 4(i) shows the sequence of local means of the instantaneous degrees in each snapshot, $\boldsymbol{\mu}_{\mathcal{T}}(\mathbf{d}^*)$, of the temporal network of Figs. 1 and 2.

Example II.18. Figure 5(d) shows the sequence of local means of inter-event durations on each link, $\boldsymbol{\mu}_{\mathcal{L}}(\boldsymbol{\Delta\tau}^*)$, of the temporal network of Figs. 1 and 2.

Definition II.11. *Distribution of local means of characteristics.* The distribution of local means $p_{\mathcal{Q}}(\boldsymbol{\mu}_{\mathcal{Q}})$ is obtained by marginalizing over the sequence of local means. It can be thought of as the composition of $p_{\mathcal{Q}}$ on $\boldsymbol{\mu}_{\mathcal{Q}}$, and is defined as the multiset of local means, $p_{\mathcal{Q}}(\boldsymbol{\mu}_{\mathcal{Q}}(\mathbf{x}^*)) = [\mu_q(\mathbf{x}^*_q)]_{q\in\mathcal{Q}}$.

Example II.19. Figure 4(j) shows the distribution of local means of the instantaneous degrees of each node, $p_{\mathcal{V}}(\boldsymbol{\mu}_{\mathcal{V}}(\mathbf{d}^*))$, and Fig. 4(l) shows the distribution of local means of the instantaneous degrees in each snapshot, $p_{\mathcal{T}}(\boldsymbol{\mu}_{\mathcal{T}}(\mathbf{d}^*))$, of the temporal network of Figs. 1 and 2.

Example II.20. Figure 5(f) shows the distribution of local means of the inter-event durations on each link, $p_{\mathcal{L}}(\boldsymbol{\mu}_{\mathcal{L}}(\boldsymbol{\Delta\tau}^*))$, of the temporal network of Figs. 1 and 2.

All of the features defined above are functions of the sequence of characteristics, so they are all coarser than the sequence. Many of them are also comparable (though not all of them), so we can establish a hierarchy between them using Proposition I.1. In Table II we list all specific marginals and moments for characteristics of links, nodes, and snapshots, and we establish their hierarchy in Fig. 6.

By combining Tables I and II, as shown for a static characteristics in Example II.4 and Figure 3 and for topological-temporal characteristics in Figures 4 and 5, we may describe most features constrained by MRRMs found in the literature.

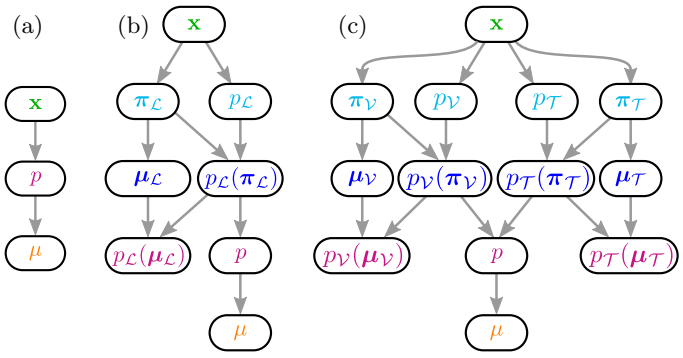


FIG. 6: **Hierarchy of the marginals and moments of a sequence of characteristics.** An arrow from a higher node to a lower one indicates that the former feature is finer than the latter. Thus, a MRRM that conserves the former feature necessarily conserves all downstream features. Conversely, a MRRM that does not conserve a given feature does not conserve any features upstream of it either. (a) For one-level sequences of characteristics, namely the aggregated features $\mathbf{k}$, $\mathbf{a}$, $\mathbf{n}$, $\mathbf{s}$, and $\mathbf{w}$, and $\mathbf{A}$, and the link-timeline features $\mathbf{t}^1$, and $\mathbf{t}^w$. (b) For two-level sequences of topological-temporal characteristics of link timelines, namely $\boldsymbol{\tau}$ and $\boldsymbol{\Delta\tau}$. (c) For two-level sequences of topological-temporal characteristics of nodes and snapshots, namely $\boldsymbol{\alpha}$, $\boldsymbol{\Delta\alpha}$ and $\mathbf{d}$.

Some of the different characteristics listed in Table I are also pairwise comparable. This enables us to construct a hierarchy of the different characteristics listed in Table I together with their marginals and moments (Table II). Figure 7 shows such a hierarchy. It may be used to derive which features are conserved by a MRRM that constrains a given feature: the model conserves all features that are below the constrained feature in the hierarchy.

Note that if two features are not comparable, it does not imply that they are independent. So one cannot conclude from the absence of a link between two features in Fig. 7 that one does not influence the other, only that it does not constrain it completely; the features may be correlated. The correlations between features that are neither comparable nor independent depend on the input temporal network that is considered. Thus, they can only be investigated on a case-by-case basis.

III. PERMUTATION METHODS

We give in this section a description of several important permutation methods. These are used to formulate and generate MRRMs found in practice. We categorize them into different classes depending on which parts of a temporal network they randomize. We furthermore show that some of these classes are compatible, and we discuss

TABLE II: Lookup table for symbols defining different marginals and moments of a sequence of characteristics. This table serves for naming MRRMs (together with Table I), and as a legend for reading Tables III and IV.

Symbol	*Meaning of symbol*	*Definition*	*Example(s)*
$\mathbf{x}$	One-level sequence of link characteristics.	$\mathbf{x} = (x_{(i,j)})_{(i,j)\in\mathcal{L}}$	
	One-level sequence of node characteristics.	$\mathbf{x} = (x_i)_{i\in\mathcal{V}}$	Fig. 3(a)
	One-level sequence of snapshot characteristics.	$\mathbf{x} = (x^t)_{t\in\mathcal{T}}$	
	Two-level sequence of link characteristics.	$\mathbf{x} = (\mathbf{x}_{(i,j)})_{(i,j)\in\mathcal{L}}$ [a]	Fig. 5(a)
	Two-level sequence of node characteristics.	$\mathbf{x} = (\mathbf{x}_i)_{i\in\mathcal{V}}$ [b]	Fig. 4(a)
$\boldsymbol{\pi}_\mathcal{L}$	Sequence of local distributions on links.	$\boldsymbol{\pi}_\mathcal{L}(\mathbf{x}) = (\pi_{(i,j)}(\mathbf{x}_{(i,j)}))_{(i,j)\in\mathcal{L}}$ [d]	Fig. 5(b)
$\boldsymbol{\pi}_\mathcal{V}$	Sequence of local distributions on nodes.	$\boldsymbol{\pi}_\mathcal{V}(\mathbf{x}) = (\pi_i(\mathbf{x}_i))_{i\in\mathcal{V}}$ [e]	Fig. 4(b)
$\boldsymbol{\pi}_\mathcal{T}$	Sequence of local distributions in snapshots.	$\boldsymbol{\pi}_\mathcal{T}(\mathbf{x}) = (\pi^t(\mathbf{x}^t))_{t\in\mathcal{T}}$ [f]	Fig. 4(e)
$p_\mathcal{L}$	Distribution of local sequences on links.	$p_\mathcal{L}(\mathbf{x}) = [\mathbf{x}_{(i,j)}]_{(i,j)\in\mathcal{L}}$ [a]	Fig. 5(c)
$p_\mathcal{V}$	Distribution of local sequences on nodes.	$p_\mathcal{V}(\mathbf{x}) = [\mathbf{x}_i]_{i\in\mathcal{V}}$ [b]	Fig. 4(c)
$p_\mathcal{T}$	Distribution of local sequences in snapshots.	$p_\mathcal{T}(\mathbf{x}) = [\mathbf{x}^t]_{t\in\mathcal{T}}$ [c]	Fig. 4(d)
$\boldsymbol{\mu}_\mathcal{L}$	Sequence of local means on links.	$\boldsymbol{\mu}_\mathcal{L}(\mathbf{x}) = (\mu_{(i,j)}(\mathbf{x}_{(i,j)}))_{(i,j)\in\mathcal{L}}$ [g]	Fig. 5(d)
$\boldsymbol{\mu}_\mathcal{V}$	Sequence of local means on nodes.	$\boldsymbol{\mu}_\mathcal{V}(\mathbf{x}) = (\mu_i(\mathbf{x}_i))_{i\in\mathcal{V}}$ [h]	Fig. 4(f)
$\boldsymbol{\mu}_\mathcal{T}$	Sequence of local means in snapshots.	$\boldsymbol{\mu}_\mathcal{T}(\mathbf{x}) = (\mu^t(\mathbf{x}^t))_{t\in\mathcal{T}}$ [i]	Fig. 4(i)
$p_\mathcal{L}(\boldsymbol{\pi}_\mathcal{L})$	Distribution of local distributions on links.	$p_\mathcal{T}(\boldsymbol{\pi}_\mathcal{L}(\mathbf{x})) = [\pi_{(i,j)}(\mathbf{x}_{(i,j)})]_{(i,j)\in\mathcal{L}}$ [d]	Fig. 5(e)
$p_\mathcal{V}(\boldsymbol{\pi}_\mathcal{V})$	Distribution of local distributions on nodes.	$p_\mathcal{V}(\boldsymbol{\pi}_\mathcal{V}(\mathbf{x})) = [\pi_i(\mathbf{x}_i)]_{i\in\mathcal{V}}$ [e]	Fig. 4(g)
$p_\mathcal{T}(\boldsymbol{\pi}_\mathcal{T})$	Distribution of local distributions in snapshots.	$p_\mathcal{T}(\boldsymbol{\pi}_\mathcal{T}(\mathbf{x})) = [\pi^t(\mathbf{x}^t)]_{t\in\mathcal{T}}$ [f]	Fig. 4(h)
$p_\mathcal{L}(\boldsymbol{\mu}_\mathcal{L})$	Distribution of local means on links.	$p_\mathcal{L}(\boldsymbol{\mu}_\mathcal{L}(\mathbf{x})) = [\mu_{(i,j)}(\mathbf{x}_{(i,j)})]_{(i,j)\in\mathcal{L}}$ [g]	Fig. 5(f)
$p_\mathcal{V}(\boldsymbol{\mu}_\mathcal{V})$	Distribution of local means on nodes.	$p_\mathcal{V}(\boldsymbol{\mu}_\mathcal{V}(\mathbf{x})) = [\mu_i(\mathbf{x}_i)]_{i\in\mathcal{V}}$ [h]	Fig. 4(j)
$p_\mathcal{T}(\boldsymbol{\mu}_\mathcal{T})$	Distribution of local means in snapshots.	$p_\mathcal{T}(\boldsymbol{\mu}_\mathcal{T}(\mathbf{x})) = [\mu^t(\mathbf{x}^t)]_{t\in\mathcal{T}}$ [i]	Fig. 4(l)
p	Distribution of one-level link characteristics.	$p(\mathbf{x}) = [x_{(i,j)}]_{(i,j)\in\mathcal{L}}$	
	Distribution of one-level node characteristics	$p(\mathbf{x}) = [x_i]_{i\in\mathcal{V}}$	Fig. 3(b)
	Distribution of one-level snapshot characteristics	$p(\mathbf{x}) = [x^t]_{t\in\mathcal{T}}$	
	Global distribution of two-level link characteristics.	$p(\mathbf{x}) = \cup_{(i,j)\in\mathcal{L}}\pi_{(i,j)}(\mathbf{x}_{(i,j)})$ [d]	Fig. 5(g)
	Global distribution of two-level node characteristics.	$p(\mathbf{x}) = \cup_{i\in\mathcal{V}}\pi_i(\mathbf{x}_i)$ [e]	Fig. 4(k)
μ	Mean of one-level link characteristics.	$\mu(\mathbf{x}) = \sum_{(i,j)\in\mathcal{L}} x_{(i,j)}/L$	
	Mean of one-level node characteristics.	$\mu(\mathbf{x}) = \sum_{i\in\mathcal{V}} x_i/N$	Fig. 3(c)
	Mean of one-level snapshot characteristics.	$\mu(\mathbf{x}) = \sum_{t\in\mathcal{T}} x^t/T$	
	Global mean of two-level link characteristics.	$\mu(\mathbf{x}) = \sum_{(i,j)\in\mathcal{L}}\sum_{m\in\mathcal{M}_{(i,j)}} x^m_{(i,j)}/(\sum_{(i,j)\in\mathcal{L}} M_{(i,j)})$	Fig. 5(h)
	Global mean of two-level node characteristics.	$\mu(\mathbf{x}) = \sum_{i\in\mathcal{V}}\sum_{m\in\mathcal{M}_i} x^m_i/(\sum_{i\in\mathcal{V}} M_i)$	Fig. 4(m)
–	Feature is not conserved.		

[a] $\mathbf{x}_{(i,j)}$: Local sequence on link, $\mathbf{x}_{(i,j)} = (x^m_{(i,j)})_{m\in\mathcal{M}_{(i,j)}}$.
[b] $\mathbf{x}_i$: Local sequence on node, $\mathbf{x}_i = (x^m_i)_{m\in\mathcal{M}_i}$.
[c] $\mathbf{x}^t$: Local sequence in snapshot, $\mathbf{x}^t = (x^t_i)_{i\in\mathcal{V}}$.
[d] $\pi_{(i,j)}(\mathbf{x}_{(i,j)})$: Local distribution on link, $\pi_{(i,j)}(\mathbf{x}_{(i,j)}) = [x^m_{(i,j)}]_{m\in\mathcal{M}_{(i,j)}}$.
[e] $\pi_i(\mathbf{x}_i)$: Local distribution on node, $\pi_i(\mathbf{x}_i) = [x^m_i]_{m\in\mathcal{M}_i}$.
[f] $\pi^t(\mathbf{x}^t)$: Local distribution in snapshot, $\pi^t(\mathbf{x}^t) = [x^t_i]_{i\in\mathcal{V}}$.
[g] $\mu_{(i,j)}(\mathbf{x}_{(i,j)})$: Local mean on link, $\mu_{(i,j)}(\mathbf{x}_{(i,j)}) = \sum_{m\in\mathcal{M}_{(i,j)}} x^m_{(i,j)}/M_{(i,j)}$.
[h] $\mu_i(\mathbf{x}_i)$: Local mean on node, $\mu_i(\mathbf{x}_i) = \sum_{m\in\mathcal{M}_i} x^m_i/M_i$.
[i] $\mu^t(\mathbf{x}^t)$: Local mean in snapshot, $\mu^t(\mathbf{x}^t) = \sum_{i\in\mathcal{V}} x^t_i/N$.

how this can be used in the design of randomization procedures.

For practical purposes we are interested in comparing randomized networks to the original network. So we are in general only interested in permutation methods that constrain the set of nodes $\mathcal{V}$, the recording interval $[t_{\min}, t_{\max}]$, and the event durations $p(\boldsymbol{\tau})$ (or for instant-event networks the number of events E). We shall call such a permutation method an *event shuffling* or an *instant-event shuffling* based on whether they permute (i.e. *shuffle*) the events in a temporal network while keeping their durations intact or shuffle the instantaneous events in an instant-event network.

Since all MRRMs of practical interest constrain both

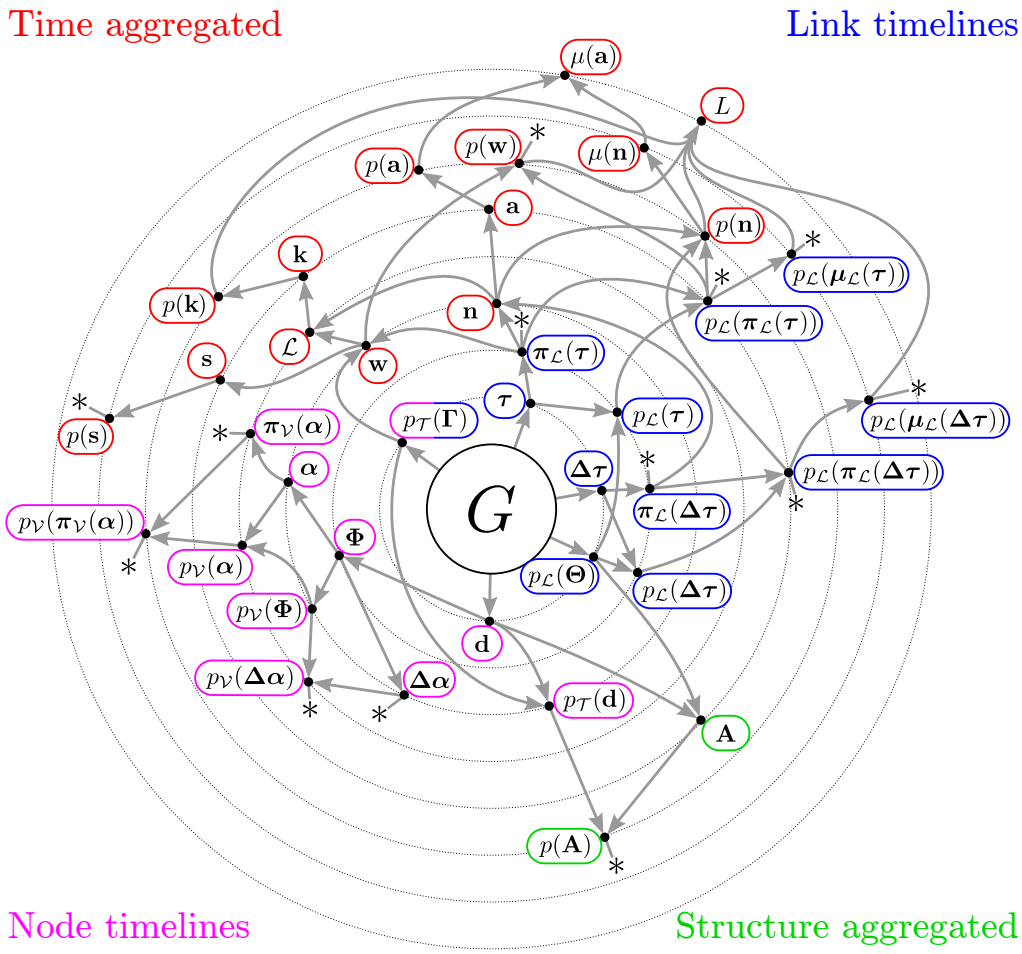


FIG. 7: **Feature hierarchy diagram.** An arrow from a higher ranking (more central) to a lower ranking feature (node in the diagram) indicates that the former feature is finer than the latter. Thus, a MRRM that constrains the feature in a given node also constrains all features of downstream nodes. Conversely, a MRRM that randomizes (i.e. does not constrain) the feature of a given node does not constrain any of the features of upstream nodes either. See Tables I and II for definitions of the features. A star ($*$) emanating from a node indicates that lower hierarchical levels follow as shown in Fig. 6. The color coding shows what type of characteristics the features correspond to: time-aggregated characteristics (i.e. topological and weighted), link-timeline characteristics, node-timeline characteristics, and structure-aggregated characteristics (i.e. purely temporal).

the set of nodes $\mathcal{V}$ and the period of observation $[t_{\min}, t_{\max}]$, we shall omit these from the names of methods below in order to avoid clutter.

Definition III.1. *Event shuffling.* We define a *event shuffling* as a permutation method that generates networks from an input temporal network by randomizing one or multiple of the indices $i \in \mathcal{V}$, $j \in \mathcal{V}$, $t \in [t_{\min}, t_{\max}]$ in all of the events while conserving their durations. An event shuffling thus constrains $\mathcal{V}$, $[t_{\min}, t_{\max}]$, and $p(\boldsymbol{\tau})$.

Definition III.2. *Instant-event shuffling.* We define an *instant-event shuffling* as a permutation method that generates networks by randomizing one or multiple of the indices $i \in \mathcal{V}$, $j \in \mathcal{V}$, $t \in \mathcal{T}$ in all of the events in an input instant-time temporal network. An instant-event shuffling thus constrains $\mathcal{V}$, $\mathcal{T}$, and E.

The coarsest event shuffling is the permutation method that randomizes everything but $\mathcal{V}$, $[t_{\min}, t_{\max}]$, and $p(\boldsymbol{\tau})$. Leaving out $\mathcal{V}$ and $[t_{\min}, t_{\max}]$ since all MRRMs we consider constrain these, it is named $\mathrm{P}[p(\boldsymbol{\tau})]$. Likewise, the coarsest possible instant-event shuffling, which constrains only $\mathcal{V}$, $\mathcal{T}$, and E, is named $\mathrm{P}[E]$.

Formally speaking any event shuffling is a refinement of an instant-event shuffling, so all event shufflings are also instant-event shufflings. However, in practice it is useful to distinguish the two as event shufflings provide a simple way to generate MRRMs that constrain the set of event durations by shuffling events in a temporal network, while instant-event shufflings must be applied to instant-event temporal networks.

While any event shuffling necessarily conserves the set of event durations $p(\boldsymbol{\tau})$, the event durations of a discrete-

time recorded temporal network may be randomized by representing each events in the temporal network by a sequence of consecutive instantaneous events and then then shuffling this instant-event network using an instant-event shuffling.

It shall be of great use to define four further constrained event shufflings, namely *timeline shufflings*, *link shufflings*, *snapshot shufflings*, and *sequence shufflings*. These can be implemented directly using the nested network representations introduced in Section I A 1 (timeline and link shufflings using the link-timeline representation, and snapshot and sequence shufflings using the snapshot-sequence representation). Timeline and link shufflings, as well as snapshot and sequence shufflings, are comparable, allowing us to use them to define compositions that result in microcanonical RRMs.

A. Link and timeline shuffling

Definition III.3. *Link shuffling.* A *link shuffling* $\mathrm{P}[\mathbf{f}(\mathcal{L}), p_{\mathcal{L}}(\boldsymbol{\Theta})]$ is an event shuffling that constrains all the individual timelines, i.e. the multiset $p_{\mathcal{L}}(\boldsymbol{\Theta}) = [\Theta_{(i,j)}]_{(i,j)\in\mathcal{L}}$. It randomizes the values of i and j for each link $(i,j) \in \mathcal{L}$, while respecting a constraint given by any function $\mathbf{f}$ of $\mathcal{L}$.

In practice a link shuffling is implemented by randomizing the links $\mathcal{L}$ in the static graph, using e.g. the Maslov-Sneppen model ($\mathrm{P}[\mathbf{k}]$) or the Erdős-Rényi model ($\mathrm{P}[L]$), and redistributing the timelines $\Theta_{(i,j)} \in p_{\mathcal{L}}(\boldsymbol{\Theta})$ at random on the new links without replacement. All link shufflings constrain $p_{\mathcal{L}}(\boldsymbol{\Theta})$, so the coarsest link shuffling is $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$.

Definition III.4. *Timeline shuffling.* A *timeline shuffling* $\mathrm{P}[\mathcal{L}, E, \mathbf{f}(p_{\mathcal{L}}(\boldsymbol{\Theta}))]$ is an event or instant-event shuffling that constrains $\mathcal{L}$ (and thus also G^{stat}). It shuffles the events on the timelines while respecting a constraint given by any function $\mathbf{f}$ of $p_{\mathcal{L}}(\boldsymbol{\Theta})$.

Timeline shufflings naturally conserve the static graph, i.e. G^{stat}. The coarsest event timeline shuffling is $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$, while the coarsest instant-event timeline shuffling is $\mathrm{P}[\mathcal{L}, E]$.

Proposition III.1. *Link shufflings and timeline shufflings are compatible.* Any link shuffling $\mathrm{P}[\mathbf{f}(\mathcal{L}), p_{\mathcal{L}}(\boldsymbol{\Theta})]$ and timeline shuffling $\mathrm{P}[\mathcal{L}, E, \mathbf{g}(p_{\mathcal{L}}(\boldsymbol{\Theta}))]$ are compatible and their composition is given by $\mathrm{P}[L, E, \mathbf{f}(\mathcal{L}), \mathbf{g}(p_{\mathcal{L}}(\boldsymbol{\Theta}))]$.

Proof. It is clear that the content of the individual timelines $\Theta_{(i,j)} \in p_{\mathcal{L}}(\boldsymbol{\Theta})$ does not in any way constrain what values $\mathcal{L}$ may take, only their number L does. Furthermore, the number of ways that we can distribute the L timelines on the links is independent of the particular configuration of $\mathcal{L}$, so $\Omega_{\mathcal{L}',p_{\mathcal{L}}(\boldsymbol{\Theta}^*)} = \Omega_{\mathcal{L}'',p_{\mathcal{L}}(\boldsymbol{\Theta}^*)}$ for all $\mathcal{L}'$, $\mathcal{L}''$. Similarly the way we can distribute the E instantaneous events on the timelines depends only on $\mathcal{L}$ through L, so also $\Omega_{\mathcal{L}',L^*} = \Omega_{\mathcal{L}'',L^*}$ for all $\mathcal{L}'$, $\mathcal{L}''$. This means that $\Omega_{\mathcal{L}',L^*} \propto \Omega_{\mathcal{L}',p_{\mathcal{L}}(\boldsymbol{\Theta}^*)}$ for all L', and since the conditional probabilities must be normed, that $P_{\mathcal{L}|p_{\mathcal{L}}(\boldsymbol{\Theta})}(\mathcal{L}^\dagger|p_{\mathcal{L}}(\boldsymbol{\Theta}^*)) = P_{\mathcal{L}|L}(\mathcal{L}^\dagger|L^*)$. Since $E \geq p_{\mathcal{L}}(\boldsymbol{\Theta})$ it then follows from Theorem 2 that $(\mathcal{L}, E)$ and $p_{\mathcal{L}}(\boldsymbol{\Theta})$ are independent conditioned on $(L, p(\boldsymbol{\tau}))$. This shows that the coarsest link shuffling, $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$, and the coarsest timeline shuffling, $\mathrm{P}[\mathcal{L}, E]$, are compatible. We next note that any link and timeline shufflings are refinements of $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$ and $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$, respectively. So applying Theorem 2 again gives that any link shuffling $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta}), \mathbf{f}(\mathcal{L})]$ and any timeline shuffling $\mathrm{P}[\mathcal{L}, E, \mathbf{g}(p_{\mathcal{L}}(\boldsymbol{\Theta}))]$ are compatible and their composition is $\mathrm{P}[L, E, \mathbf{f}(\mathcal{L}), \mathbf{g}(p_{\mathcal{L}}(\boldsymbol{\Theta}))]$. □

B. Sequence and snapshot shuffling

We define the particular types of event shufflings termed *sequence* and *snapshot* shufflings. These are naturally defined as instant-event shufflings since they rely on randomizing either the order of temporal snapshots or the events inside each snapshot. Since event durations induce correlations between neighboring snapshots, it is not generally possible to design microcanonical sequence and snapshot shufflings that conserve event durations.

Definition III.5. *Sequence shuffling.* A *sequence shuffling* $\mathrm{P}[\mathbf{f}(\mathbf{t}), p_{\mathcal{T}}(\boldsymbol{\Gamma})]$ is a timeline shuffling that constrains the distribution of instantaneous snapshot graphs, $p_{\mathcal{T}}(\boldsymbol{\Gamma})$. It randomizes the order of snapshots, i.e. the set $\mathcal{T}$, in a manner that may depend on any function of the times of the events, $\mathbf{t}$.

The coarsest sequence shuffling is $\mathrm{P}[p_{\mathcal{T}}(\boldsymbol{\Gamma})]$. As noted in the definition above, all sequence shufflings are timeline shufflings as $p_{\mathcal{T}}(\boldsymbol{\Gamma}) \leq (\mathcal{L}, E)$. They are thus compatible with link shufflings, but it is practical to define them separately as they are furthermore compatible with the snapshot shufflings defined below.

Definition III.6. *Snapshot shuffling.* A *snapshot shuffling* $\mathrm{P}[\mathbf{t}, \mathbf{f}(p_{\mathcal{T}}(\boldsymbol{\Gamma}))]$ is an instantaneous event shuffling that constrains the time of each event, i.e. $\mathbf{t}$. It randomizes each snapshot graph Γ^t individually in a manner that may be constrained by any function $\mathbf{f}$ of $p_{\mathcal{T}}(\boldsymbol{\Gamma})$.

Snapshot shufflings are typically implemented by randomizing the snapshot graphs individually and independently using any permutation method for static graphs. The coarsest snapshot shuffling is $\mathrm{P}[\mathbf{t}]$, which is equivalent to $\mathrm{P}[\mathbf{A}]$ since any permutation of event indices is indistinguishable from another.

Proposition III.2. *Sequence shufflings and snapshot shufflings are compatible.* Any sequence shuffling $\mathrm{P}[\mathbf{f}(\mathbf{t}), p_{\mathcal{T}}(\boldsymbol{\Gamma})]$ and snapshot shuffling $\mathrm{P}[\mathbf{t}, \mathbf{g}(p_{\mathcal{T}}(\boldsymbol{\Gamma}))]$ are compatible and their composition is given by $\mathrm{P}[p(\mathbf{t}), \mathbf{f}(\mathbf{t}), \mathbf{g}(p_{\mathcal{T}}(\boldsymbol{\Gamma}))]$.

Proof. Following the same reasoning as in the proof of Proposition III.1, we note that $\Omega_{\mathbf{t}',p_{\mathcal{T}}(\boldsymbol{\Gamma}^*)} = \Omega_{\mathbf{t}',p(\mathbf{A}^*)}$ for

all $\mathbf{t}'$ and that $p(\mathbf{A})$ satisfies $p(\mathbf{A}) \geq \mathbf{t}$ and $p(\mathbf{A}) \geq p_{\mathcal{T}}(\mathbf{\Gamma})$. Thus, $\mathbf{t}$ is independent of $p_{\mathcal{T}}(\mathbf{\Gamma})$ conditioned on $p(\mathbf{A})$. So the coarsest sequence and snapshot shufflings, $\mathrm{P}[p_{\mathcal{T}}(\mathbf{\Gamma})]$ and $\mathrm{P}[\mathbf{t}]$ are compatible. Consequently, since all sequence shufflings $\mathrm{P}[\mathbf{f}(\mathbf{t}), p_{\mathcal{T}}(\mathbf{\Gamma})]$ and snapshot shufflings $\mathrm{P}[\mathbf{A}, \mathbf{f}(p_{\mathcal{T}}(\mathbf{\Gamma}))]$ are refinements of $\mathrm{P}[p_{\mathcal{T}}(\mathbf{\Gamma})]$ or $\mathrm{P}[\mathbf{t}]$, respectivecly, Theorem 2 tells us that they are compatible and their composition is $\mathrm{P}[p(\mathbf{A}), \mathbf{f}(\mathbf{t}), \mathbf{g}(p_{\mathcal{T}}(\mathbf{\Gamma}))]$. □

Proposition III.3. *Link shufflings and sequence shufflings are compatible.* Any link shuffling $\mathrm{P}[\mathbf{f}(\mathcal{L}), p_{\mathcal{L}}(\mathbf{\Theta})]$ and any sequence shuffling $\mathrm{P}[\mathbf{f}(\mathbf{t}), p_{\mathcal{T}}(\mathbf{\Gamma})]$ are compatible.

Proof. Since sequence shufflings are timeline shufflings, they are by virtue of Proposition III.1 also compatible with link shufflings. □

IV. CLASSIFYING RANDOMIZED REFERENCE MODELS

In this section we describe and classify MRRMs found in the literature as well as some additional models that arise naturally as generalizations of these based on the theory developed in Sections I and III and the features defined in Section II. We provide unambiguous names for the MRRMs (see Definition IV.1 below) and we investigate how they affect the temporal network characteristics listed in Table I.

This is summarized in two tables: Table III lists MRRMs for instant-event temporal networks (networks where events are models as instantaneous, Def. I.2); Table IV lists MRRMs for temporal networks with event durations (Def. I.1). Four figures provide hierarchical orderings of the MRRMs: Fig. 8 shows the hierarchy of link shufflings (Def. III.3); Fig. 9 shows the hierarchy of timeline shufflings (Def. III.4); Fig. 10 shows the hierarchy of sequence shufflings (Def. III.5); Fig. 11 finally shows the hierarchy of snapshot shufflings (Def. III.6).

The features listed in Tables III and IV and used in the names of MRRMs are defined in Tables I and II and are described in more detail in Section II.

Two different MRRMs may be combined to form a new MRRM by applying the second MRRM to each graph in the ensemble generated by the first. This defines a *composition* (Definition I.12) of the MRRMs and results in a model that is shuffles more than either of the two original MRRMs. Not all compositions lead to a microcanonical RRM, however. We refer to pairs of MRRMs whose composition does result in a microcanonical RRM as *compatible* (Definition I.13). As shown in Section III, link shufflings and timeline shufflings are compatible (Proposition III.1), while sequence shufflings are compatible with both snapshot shufflings and link shufflings (Propositions III.2 and III.3). Some MRRMs generated by composition of two other MRRMs are listed in Subsection IV H, but many more may be generated directly from the existing MRRMs described below by applying any pair of compatible ones in composition: The number of new MRRMs we can generate this way is simply the number of different compatible pairs (for the 7 link shufflings, 19 timeline shufflings, 2 sequence shufflings, and 7 snapshot shufflings below this number is 161).

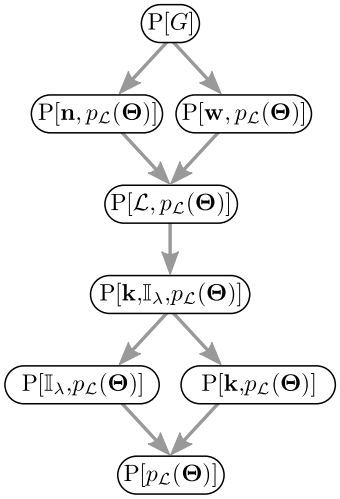


FIG. 8: **Hierarchy of link shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features.

This section is organized as follows. Subsection IV A first defines the naming convention we will use for MRRMs for temporal networks. Subsection IV B presents general event shufflings that do not fit into one of the restricted classes; while Subsections IV C–IV F present the four restricted classes, namely link shufflings (IV C), timeline shufflings (IV D), sequence shufflings (IV E), and snapshot shufflings (IV F). Event and instant-event shufflings are presented in separate lists in each subsection. Subsection IV G describes several MRRMs that can be classified as intersections of link and timeline shufflings or of timeline and snapshot shufflings. Subsection IV H describes compositions of MRRMs that can be found in the literature. Subsection IV I discusses microcanonical RRMs that use additional metadata on nodes. Subsection IV J finally surveys reference models that are not MRRMs. Namely, *canonical* RRMs, reference models that do not maximize entropy and in particular models based on bootstrapping, and finally reference models that directly randomize summary statistics (i.e. features) but do not take into account the networked structure of the system.

A. Naming convention

Definition IV.1. *Naming convention for MRRMs for temporal networks.* A MRRM is completely defined by the feature(s) it constrains (Def. I.7), so we use a naming

TABLE III: **Effects of MRRMs on features of instant-event temporal networks.** See Tables I and II for symbol definitions and naming. Care should be taken when interpreting results using this table since a feature that is not conserved (–) by a randomization procedure is not necessarily completely randomized either (see discussion at the end of Sec. II). Note that multiple common names for a MRRM may exist; in these cases we have chosen our preferred one.

Canonical name	Common name	Aggregated						Temporal-topological							
		topological			weighted		temp.		node			link			
		G^{stat}	k_i	L	s_i	$w_{(i,j)}$	A^t	Γ^t	Φ_i	$\Delta\alpha_i^m$	d_i^t	$\Theta_{(i,j)}$	$\Delta\tau_{(i,j)}^m$	$t_{(i,j)}^1$	$\mathbf{t}_{(i,j)}^w$
Event shufflings:															
P[E]	*Instant-event shuffling*	–	–	–	μ	–	μ	–	–	–	μ	–	–	–	–
Link shufflings:															
P[$p_{\mathcal{L}}(\mathbf{\Theta})$]	*Link shuffling*	–	μ	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
P[$\mathbb{I}_\lambda, p_{\mathcal{L}}(\mathbf{\Theta})$]		$\mathbb{I}_\lambda$	μ	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
P[$\mathbf{k}, p_{\mathcal{L}}(\mathbf{\Theta})$]	*Maslov-Sneppen*	–	x	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
P[$\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\mathbf{\Theta})$]		$\mathbb{I}_\lambda$	x	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
Timeline shufflings:															
P[$\mathcal{L}$,E]	*Timeline shuffling*	x	x	x	μ	μ	μ	–	–	–	μ	–	–	–	–
P[$\mathbf{w}$]		x	x	x	x	x	μ	–	–	–	μ	–	–	–	–
P[$\mathbf{w}$,$\mathbf{t}^1$,$\mathbf{t}^w$]		x	x	x	x	x	μ	–	–	–	μ	–	$\boldsymbol{\mu}_{\mathcal{L}}$	x	x
P[$\mathbf{w}, \boldsymbol{\pi}_{\mathcal{L}}(\mathbf{\Delta\tau})$]		x	x	x	x	x	μ	–	–	–	μ	–	$\boldsymbol{\pi}_{\mathcal{L}}$	–	–
P[$\mathbf{w}, \boldsymbol{\pi}_{\mathcal{L}}(\mathbf{\Delta\tau})$,$\mathbf{t}^1$]		x	x	x	x	x	μ	–	–	–	μ	–	$\boldsymbol{\pi}_{\mathcal{L}}$	x	x
P[**per**($\mathbf{\Theta}$)]		x	x	x	x	x	μ	–	–	–	μ	–	x	–	–
Sequence shufflings:															
P[$p_{\mathcal{T}}(\mathbf{\Gamma})$]	*Sequence shuffling*	x	x	x	x	x	p	$p_{\mathcal{T}}$	–	–	$p_{\mathcal{T}}$	–	–	–	–
P[$p_{\mathcal{T}}(\mathbf{\Gamma})$,**sgn**($\mathbf{A}$)]		x	x	x	x	x	p, sgn	$p_{\mathcal{T}}$	–	–	$p_{\mathcal{T}}$	–	–	–	–
Snapshot shufflings:															
P[$\mathbf{t}$]	*Snapshot shuffling*	–	–	–	μ	–	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–
P[$\mathbf{t}$,$\mathbf{\Phi}$]		–	–	–	μ	–	x	–	x	x	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–
P[$\mathbf{d}$]		–	–	–	μ	–	x	–	x	x	x	–	–	–	–
P[**iso**($\mathbf{\Gamma}$)]		–	–	–	μ	–	x	$\simeq$	–	–	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–	–
P[**iso**($\mathbf{\Gamma}$),$\mathbf{\Phi}$]		–	–	–	μ	–	x	$\simeq$	x	x	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–	–
Intersections:															
P[$\mathcal{L}$,$\mathbf{t}$]		x	x	x	μ	μ	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–
P[$\mathbf{w}$,$\mathbf{t}$]	*Timestamp shuffling*	x	x	x	x	x	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–
P[$\mathcal{L}$,$p_{\mathcal{L}}(\mathbf{\Theta})$]		x	x	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
P[$\mathbf{w}$,$p_{\mathcal{L}}(\mathbf{\Theta})$]		x	x	x	x	x	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
Compositions:															
P[L, E]		–	μ	x	μ	μ	μ	–	–	–	μ	–	–	–	–
P[$\mathbf{k}, p(\mathbf{w}), \mathbf{t}$]		–	x	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–
P[$\mathbf{k}, \mathbb{I}_\lambda, p(\mathbf{w}), \mathbf{t}$]		$\mathbb{I}_\lambda$	x	x	μ	p	x	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–

convention that lists the corresponding set of features. In particular, if a MRRM constrains the features $\mathbf{x}$ and $\mathbf{y}$, we name it P[$\mathbf{x}, \mathbf{y}$].

Our naming convention is not unique as we may devise different ways to name the same MRRM (see for example the description of the MRRM P[$\mathbf{w}, \mathbf{t}$] in Section IV G). However, and much more importantly, it is unambiguous as a set of features uniquely defines a single MRRM (Def. I.7).

Since MRRMs in practice always constrain the set of nodes $\mathcal{V}$ and the time-interval during which the interactions take place, $[t_{\max}, t_{\min}]$, we do not include these features in the names of MRRMs below in order to avoid clutter.

B. Event shufflings

Here we present general event and instant-event shufflings that do not fall into one of the four restricted

TABLE IV: **Effects of MRRMs on features of temporal networks.** See Tables I and II for symbol definitions and naming. Instant-event shufflings may be used on a temporal network (Def. I.1) with event durations by representing it by a instant-event temporal network (Def. I.2) where each event is represented by a contiguous sequence of instantaneous events. Care should be taken when interpreting results using this table since a feature that is not conserved (–) by a randomization procedure is not necessarily completely randomized either (see discussion at the end of Sec. II). Note that multiple common names for a MRRM may exist; in these cases we have chosen our preferred one.

Canonical name	Common name	Aggregated								Temporal-topological									
		topological			weighted				temp.	node					link				
		G^{stat}	k_i	L	a_i	s_i	$n_{(i,j)}$	$w_{(i,j)}$	A^t	Γ^t	Φ_i	α_i^m	$\Delta\alpha_i^m$	d_i^t	$\Theta_{(i,j)}$	$\tau_{(i,j)}^m$	$\Delta\tau_{(i,j)}^m$	$t_{(i,j)}^1$	$t_{(i,j)}^w$
Event shufflings:																			
$\mathrm{P}[p(\boldsymbol{\tau})]$	*Event shuffling*	–	–	–	–	μ	–	–	μ	–	–	–	–	μ	–	μ	–	–	–
$\mathrm{P}[p(\mathbf{t},\boldsymbol{\tau})]$		–	–	–	–	μ	–	–	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–
Link shufflings:																			
$\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$	*Link shuffling*	–	μ	x	μ	μ	p	p	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
$\mathrm{P}[\mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})]$		$\mathbb{I}_\lambda$	μ	x	μ	μ	p	p	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
$\mathrm{P}[\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$	*Maslov-Sneppen*	–	x	x	μ	μ	p	p	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
$\mathrm{P}[\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})]$		$\mathbb{I}_\lambda$	x	x	μ	μ	p	p	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
Timeline shufflings:																			
$\mathrm{P}[\mathcal{L},p(\boldsymbol{\tau})]$	*Timeline shuffling*	x	x	x	–	μ	μ	μ	μ	–	–	–	–	μ	–	μ	–	–	–
$\mathrm{P}[\mathcal{L},p(\mathbf{t},\boldsymbol{\tau})]$		x	x	x	–	μ	μ	μ	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–
$\mathrm{P}[\mathbf{n},p(\mathbf{t},\boldsymbol{\tau})]$		x	x	x	x	μ	x	μ	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–
$\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	$\boldsymbol{\pi}_{\mathcal{L}}$	–	–	–
$\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}),\mathbf{t}^1,\mathbf{t}^w]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	$\boldsymbol{\pi}_{\mathcal{L}}$	$\boldsymbol{\mu}_{\mathcal{L}}$	x	x
$\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}),\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	$\boldsymbol{\pi}_{\mathcal{L}}$	$\boldsymbol{\pi}_{\mathcal{L}}$	–	–
$\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}),\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}),\mathbf{t}^1]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	$\boldsymbol{\pi}_{\mathcal{L}}$	$\boldsymbol{\pi}_{\mathcal{L}}$	x	x
$\mathrm{P}[\mathbf{per}(\boldsymbol{\Theta})]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	x	x	–	–
$\mathrm{P}[\boldsymbol{\tau},\boldsymbol{\Delta\tau}]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	x	x	–	–
Intersections:																			
$\mathrm{P}[\mathcal{L},p_{\mathcal{L}}(\boldsymbol{\Theta})]$		x	x	x	μ	μ	p	p	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
$\mathrm{P}[\mathbf{w},p_{\mathcal{L}}(\boldsymbol{\Theta})]$		x	x	x	μ	x	p	x	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
$\mathrm{P}[\mathbf{n},p_{\mathcal{L}}(\boldsymbol{\Theta})]$		x	x	x	x	μ	x	p	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
Instant-event shufflings:																			
$\mathrm{P}[E]$	*Instant-event shuffling*	–	–	–	–	μ	–	–	μ	–	–	–	–	μ	–	–	–	–	–
Timeline shufflings:																			
$\mathrm{P}[\mathcal{L},E]$	*Timeline shuffling*	x	x	x	–	μ	–	μ	μ	–	–	–	–	μ	–	–	–	–	–
$\mathrm{P}[\mathbf{w}]$		x	x	x	–	x	–	x	μ	–	–	–	–	μ	–	–	–	–	–
$\mathrm{P}[\mathbf{w},\mathbf{t}^1,\mathbf{t}^w]$		x	x	x	–	x	–	x	μ	–	–	–	–	μ	–	–	–	x	x
$\mathrm{P}[\mathbf{w},\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	–	$\boldsymbol{\pi}_{\mathcal{L}}$	–	–
$\mathrm{P}[\mathbf{w},\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}),\mathbf{t}^1,\mathbf{t}^w]$		x	x	x	x	x	x	x	μ	–	–	–	–	μ	–	$\boldsymbol{\mu}_{\mathcal{L}}$	$\boldsymbol{\pi}_{\mathcal{L}}$	x	x
Sequence shufflings:																			
$\mathrm{P}[p_{\mathcal{T}}(\boldsymbol{\Gamma})]$	*Sequence shuffling*	x	x	x	–	x	–	x	p	$p_{\mathcal{T}}$	–	–	–	$p_{\mathcal{T}}$	–	–	–	–	–
$\mathrm{P}[p_{\mathcal{T}}(\boldsymbol{\Gamma}),\mathbf{sgn}(\mathbf{A})]$		x	x	x	–	x	–	x	p, sgn	$p_{\mathcal{T}}$	–	–	–	$p_{\mathcal{T}}$	–	–	–	–	–
Snapshot shufflings:																			
$\mathrm{P}[\mathbf{t}]$	*Snapshot shuffling*	–	–	–	–	μ	–	–	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–	–
$\mathrm{P}[\mathbf{t},\boldsymbol{\Phi}]$		–	–	–	–	μ	–	–	x	–	x	x	x	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–	–
$\mathrm{P}[\mathbf{d}]$		–	–	–	–	μ	–	–	x	–	x	x	x	x	–	–	–	–	–
$\mathrm{P}[\mathbf{iso}(\boldsymbol{\Gamma})]$		–	–	–	–	μ	–	–	x	$\simeq$	–	–	–	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–	–	–
$\mathrm{P}[\mathbf{iso}(\boldsymbol{\Gamma}),\boldsymbol{\Phi}]$		–	–	–	–	μ	–	–	x	$\simeq$	x	x	x	$\boldsymbol{\pi}_{\mathcal{T}}$	–	–	–	–	–
Intersections:																			
$\mathrm{P}[\mathcal{L},\mathbf{t}]$		x	x	x	–	μ	–	μ	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–	–
$\mathrm{P}[\mathbf{w},\mathbf{t}]$	*Timestamp shuffling*	x	x	x	–	x	–	x	x	–	–	–	–	$\boldsymbol{\mu}_{\mathcal{T}}$	–	–	–	–	–

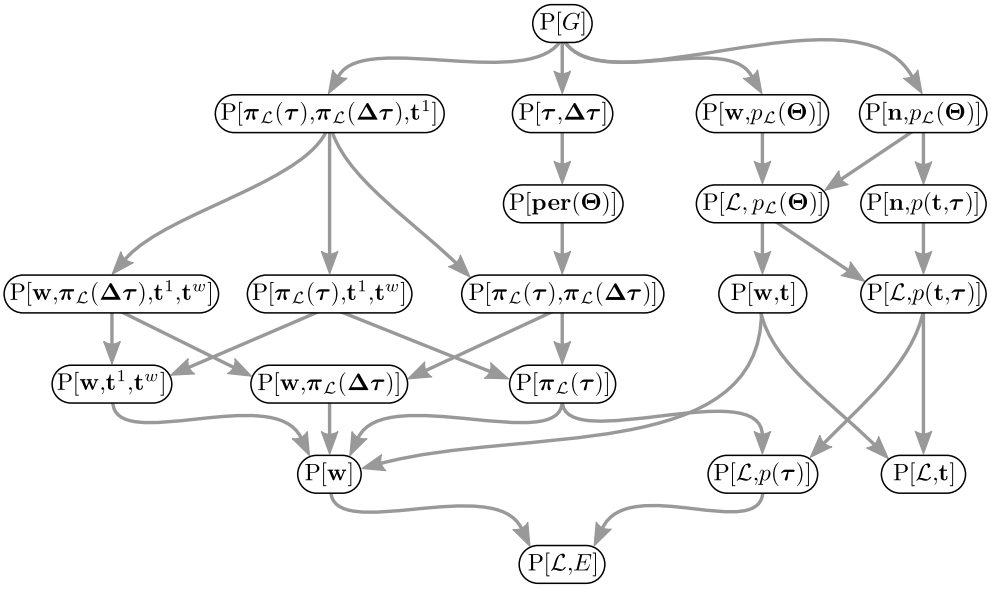


FIG. 9: **Hierarchy of timeline shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features.

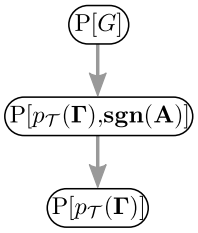


FIG. 10: **Hierarchy of sequence shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features.

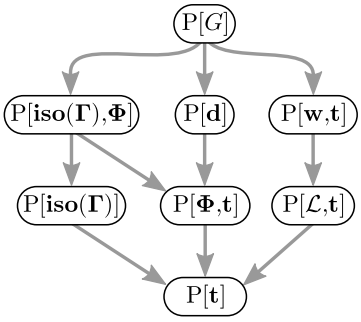


FIG. 11: **Hierarchy of snapshot shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features.

subclasses,i.e. link-, timeline-, sequence-, and snapshot-shuffling (see definitions in Section III).

We present first instant-event shufflings (Def. III.2), which shuffle the reduced instantaneous events (i, j, t) in an instant-event temporal network (Def. I.2), followed by event shufflings (Def. III.1), which shuffle events (i, j, t, τ) in a temporal network with durations (Def. I.1).

1. *Instant-event shufflings*

P[E] permutes the instantanous events at random without any constraints. P[E] is thus the coarsest instant-event shuffling possible. P[E] was employed in Ref. [35].

2. *Event shufflings*

P[$p(\boldsymbol{\tau})$] constrains only the set of event durations but randomizes everything else in the network. P[$p(\boldsymbol{\tau})$] is the coarsest event shuffling.

P[$p(\mathbf{t}, \boldsymbol{\tau})$] randomly permutes the events between links in the network while keeping the time at which the events occur and their duration constrained. P[$p(\mathbf{t}, \boldsymbol{\tau})$] can be seen as a refinement of P[$\mathbf{t}$] (defined in Section IV F below) to preserve event durations. It conserves the distribution of event durations and the times at which they occur.

C. Link shufflings

Link shufflings (Def. III.3) alter the aggregate network topology but conserve temporal structure locally on each link (the nodes' activity timelines generally change though). Formally, we randomize the node pairs (i, j) corresponding to each link in $\mathcal{L}$. Link shufflings always conserve $p_{\mathcal{L}}(\boldsymbol{\Theta})$ and all coarser features (see Fig. 7). Time aggregated network features may be constrained or randomized depending on the link shuffling. Temporal-topological correlations are generally always randomized by link shufflings. Link shufflings are compatible with timeline shufflings (Subsec. IV D) and with sequence shufflings (Subsec. IV E).

All link shuffling are event shufflings since they automatically constrain the contact durations (i.e. $p(\boldsymbol{\tau})$).

P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$] permutes the links and associated timelines between all node pairs (i, j) without any constraints on the static network structure. It is the link-shuffling that maximally randomizes the temporal network. It draws G^{stat} uniformly from the ensemble of all Erdős-Rényi (ER) [50] random graphs with the same nodes $\mathcal{V}$ and number of links L as the original network. P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$] was employed in Ref. [10] where it was referred to as the *Erdős-Rényi model*. Since P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$] is the coarsest possible link shuffling, we may simply refer to it as *link shuffling*.

P[$\mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})$] permutes the links and associated timelines between all node pairs (i, j) with the additional constraint that G^{stat} must be connected. It draws G^{stat} from the ensemble of connected ER graphs with N nodes and L links. P[$\mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})$] was called *rewiring* in Ref. [20] and *random network* in Ref. [30].

P[$\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})$] permutes the links and associated timelines between all node pairs (i, j) while keeping the sequence of degrees, $\mathbf{k}$, constrained. The procedure is typically implemented using the algorithm of Maslov and Sneppen [53]. P[$\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})$] was called *randomized edges* in Refs. [1, 21, 36] *random link shuffling* in Ref. [14], and *randomized structure* in Ref. [33].

P[$\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})$] adds the additional constraint to P[$\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})$] that the static network of the sampled reference networks should be connected. P[$\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})$] was called *configuration model* in Ref. [30].

D. Timeline shufflings

Timeline shufflings (Def. III.4) are permutation methods that randomize timelines without changing the topology of the aggregate network. Thus, they always constrain G^{stat} and all coarser features. They typically randomize temporal-topological features of nodes, while temporal-topological features of links may or may not be randomized depending on the timeline shuffling. Timeline shufflings are compatible with link shufflings (Subsec. IV C).

We list below timeline shufflings, beginning with instant-event shufflings and followed by event shufflings. Most of these are found in the literature, some are refinements of instant-event shufflings to conserve event durations.

1. *Instant-event shufflings*

P[$\mathcal{L}$,E] completely randomizes the events over all the links, while conserving G^{stat}. It was called *random(ized) contacts* in Refs. [1, 21]. It is the coarsest possible timeline shuffling and we may thus refer to it simply as the *timeline shuffling*.

P[$\mathbf{w}$] randomizes the timestamps of the instananeous events on each individual timeline, i.e. it constrains only $\mathbf{w}$. P[$\mathbf{w}$] was called *random time(s)* in Refs. [1, 14, 18, 21], *uniformly random times* in Ref. [30], *temporal mixed edges* in Ref. [4], *poissonized IEIs* in Ref. [20], and *SRan* in Ref. [29]. P[$\mathbf{w}$] was also employed in Ref. [35, 57].

P[$\mathbf{w}$,$\mathbf{t}^1$, $\mathbf{t}^w$] redistributes instantaneous events them at random while keeping their number on each link and the times of the first and last events constrained, $\mathbf{w}$, $\mathbf{t}^1$, and $\mathbf{t}^w$, respectively. It is a maximum entropy version of the *poor man's method* introduced in [31] (see Section IV J 2).

P[$\mathbf{w}$,$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$] permutes the instantaneous events inside each timeline while conserving the set of inter-event durations, $\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$. It is the instant-event shuffling equivalent of P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$,$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$] defined below.

P[$\mathbf{w}$,$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$,$\mathbf{t}^1$,$\mathbf{t}^w$] permutes the inter-event durations between the instantaneous events locally on each link while keeping the times of the first event on each link fixed. P[$\mathbf{w}$, $\boldsymbol{\pi}_{\mathcal{T}}(\boldsymbol{\Delta\tau})$,$\mathbf{t}^1$] was named *shuffled IEIs* in Refs. [20, 22].

2. *Event shufflings*

P[$\mathcal{L}$,$p(\boldsymbol{\tau})$] permutes the events at random between all timelines, while constraining the static network structure G^{stat}. P[$\mathcal{L}$,$p(\boldsymbol{\tau})$] is the coarsest timeline shuffling which constrains event durations, and is the event shuffling equivalent of P[$\mathcal{L}$,E].

P[$\mathcal{L}$,$p(\mathbf{t}, \boldsymbol{\tau})$] randomly permutes the events while constraining their starting times as well as G^{stat}. It can be seen as a refinement of P[$\mathcal{L}$,$\mathbf{t}$] (defined in Section IV G below) to networks with event durations.

P[$\mathbf{n}$,$p(\mathbf{t},\boldsymbol{\tau})$] is an event shuffling variant of P[$\mathbf{w}$,$\mathbf{t}$] (defined in Section IV G below). It conserves the distribution of event durations, their starting times, the cumulative activity timeline $\mathbf{A}$, and the static network topology and contact frequencies $\mathbf{n}$, but not necessarily their cumulative durations $\mathbf{w}$.

P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$] randomizes the events on each link while conserving the total number of events and their durations on each link. The inter-event durations are randomized and asymptotically follow exponential distributions.

P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$,$\mathbf{t}^1$,$\mathbf{t}^w$] redistributes events uniformly inside each timeline, while keeping starting times of the first and last events constrained. P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$, $\mathbf{t}^1$, $\mathbf{t}^w$] is a natural refinement of P[$\mathbf{w}$, $\mathbf{t}^1$, $\mathbf{t}^w$] to event shuffling.

P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$, $\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$] permutes the event and inter-event durations on each link. P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$, $\boldsymbol{\pi}_{\mathcal{L}}\boldsymbol{\Delta\tau}$] is referred to as *interval shuffling* in Ref. [9].

P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$, $\boldsymbol{\pi}_{\mathcal{L}}\boldsymbol{\Delta\tau}$,$\mathbf{t}^1$] adds another constraint to P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$, $\boldsymbol{\pi}_{\mathcal{L}}(\Delta\tau)$] so that it conserves the first event time on each link (it is the event shuffling variant of the instant-event shuffling P[$\boldsymbol{\pi}_{\mathcal{L}}\boldsymbol{\Delta\tau}$,$\mathbf{t}^1$]).

P[$\mathbf{per}(\boldsymbol{\Theta})$] randomly translates the whole sequence of event and inter-event durations on each timeline, i.e it randomizes $\mathbf{t}^1$ for each timeline indivdually using periodic boundary conditions. P[$\mathbf{per}(\boldsymbol{\Theta})$] was named *random offset* in Ref. [2].

P[$\boldsymbol{\tau}$,$\boldsymbol{\Delta\tau}$] randomly translates the whole sequence of event and inter-event durations on each timeline using hard boundary conditions. It is thus more constrained than P[$\mathbf{per}(\boldsymbol{\Theta})$].

E. Sequence shufflings

Sequence shufflings (Def. III.5) are are a particular kind of instant-event shufflings. They randomize the sequence of snapshots while leaving the individual snapshots unchanged in the snapshot-sequence representation. They thus generally destroy temporal correlations in link- and node-activities. Conversely, since they conserve $p_{\mathcal{T}}(\boldsymbol{\Gamma})$ they conserve the weighted aggregated network, $\mathbf{w}$, and consequently G^{stat}, as well as all instantaneous topological correlations inside snapshots. This in particular means that sequence shufflings are also timeline shufflings (see Def. III.5), and are thus compatible with both snapshot shufflings (Subsec. IV F) and timeline shufflings (Subsec. III.4) .

We have identified the following two sequence shufflings in the literature.

P[$p_{\mathcal{T}}(\boldsymbol{\Gamma})$] randomly permutes the timestamps of the snaphots. P[$p_{\mathcal{T}}(\boldsymbol{\Gamma})$] was named *reshuffled sequences* in Ref. [37]; it appears in Refs. [12], in Ref. [4] as *random ordered*, in Ref. [18] as *shuffled times*, and in Ref. [13] as *reshuffle*. Since it is the coarsest possible sequence shuffling, we may name it simply *sequence shuffling*.

P[$p_{\mathcal{T}}(\boldsymbol{\Gamma})$,$\mathbf{sgn}(\mathbf{A})$] permutes the timestamps of the snapshots where at least one event takes place, i.e. where $\text{sgn}(A^t) = 1$. It thus constrains the function $\mathbf{sgn}(\mathbf{A}) = (\text{sgn}(A^t))_{t\in\mathcal{T}}$. It was employed in Ref. [18] with the name of *shuffled times*.

F. Snapshot shufflings

Snapshot shufflings (Def. III.6), as sequence shufflings, are all instant-event shufflings. They randomize the instantaneous snapshot graph corresponding to each time $t \in \mathcal{T}$, while conserving the times t. Snapshot shufflings all constrain $\mathbf{t}$ (and thus $\mathbf{A}$); they generally destroy temporal-topological features of links, and may or may not constrain temporal topological features of nodes (i.e. $\mathbf{d}$ and coarser features) depending on the snapshot shuffling. Snapshot shufflings are compatible with sequence shufflings (Subsec. IV E).

We list below snapshot shufflings found in the literature as well as some new ones.

P[$\mathbf{t}$] resamples the events inside each snapshot while constraining only the number of events at each time $t \in \mathcal{T}$. Each snapshot Γ^t is an instance of an Erdős-Rényi graph with N nodes and $E^t = A^t/2$ edges. P[$\mathbf{t}$] is the coarsest possible snapshot shuffling, and we thus also refer to it simply as *snapshot shuffling*. It was called *random network* in Ref. [18].

P[$\mathbf{t}$,$\boldsymbol{\Phi}$] resamples the events inside snapshots while additionally constraining the set of nodes that are active at t, i.e. it constrains $\boldsymbol{\Phi}$. We have here included P[$\mathbf{t}$,$\boldsymbol{\Phi}$] as it provides a rare MRRM that conserves the nodes' activity and inactivity durations (besides the finer P[$\mathbf{d}$] introduced below).

P[$\mathbf{d}$] resamples the events inside each snapshot while constraining the instantaneous degrees $\mathbf{d}$ of each node. P[$\mathbf{d}$] was called *time ordered and reshuffled networks* in Ref. [4] and *degree preserved network* in Ref. [18], and was also applied in Ref. [40].

P[$\mathbf{iso}(\boldsymbol{\Gamma})$] consists in randomizing the identity of the nodes in each time snapshot. Each snapshot graph in the randomized network $(\Gamma^t)'$ is thus isomorphic to the corresponding Γ^t of the original network, $(\Gamma^t)' \simeq \Gamma^t$, i.e. the shuffling constrains the isomorphism class of all snapshot graphs, $\mathbf{iso}(\boldsymbol{\Gamma}) = (\text{iso}(\Gamma^t))_{m=1}^{T}$. P[$\mathbf{iso}(\boldsymbol{\Gamma})$] was named *anonymize* in Ref. [13].

P[$\mathbf{iso}(\boldsymbol{\Gamma})$,$\boldsymbol{\Phi}$] consists in randomizing the identity of nodes at each time step, but only nodes that are active are permuted. It thus combines P[$\mathbf{iso}(\boldsymbol{\Gamma})$] and P[$\boldsymbol{\Phi}$] by intersection.

G. Intersections of two shufflings

Several permutation methods constrain features corresponding to two classes of the restricted shufflings, and can thus be classified as intersections of the these. We have namely found permutation methods in the literature that are intersections of timeline and snapshot shufflings and of link and timeline shufflings. Intersections of timeline and snapshot shufflings are compatible with both link and sequence shufflings, while intersections of link and timeline shufflings are compatible with both link shufflings , timeline shufflings, and sequence shufflings.

We list first intersections of instant-event shufflings, followed by event shufflings.

1. Instant-event shufflings

The two following instant-event shufflings are intersections of timeline and snapshot shufflings. They are thus compatible with both link shufflings (Subsec. IV C) and sequence shufflings (Subsec. IV E).

P[$\mathcal{L}$,$\mathbf{t}$] resamples the events inside each snapshot while constraining G^{stat}, i.e. assigning the resampled events only to node pairs with at least one event in G. Each snapshot is a subgraph of G^{stat} in which A^t links are randomly chosen.

P[$\mathbf{w}$,$\mathbf{t}$] randomly permutes the timestamps t_q of all events, while keeping i_q and j_q fixed, i.e. it constrains $(\mathbf{i},\mathbf{j}) = (i_q,j_q)_{q=1}^{Q}$ and $p(\mathbf{t}) = \{t_q\}_{q=1}^{Q}$. Completely equivalently, we may define the timestamp shuffling by constraining the timestamps $\mathbf{t}$ and permuting the pairs (i_q,j_q), i.e. constraining $\mathbf{w}$ (see detailed description in the next section). Due to the indistinguishability of networks obtained through permutation of event indices, both are equivalent to conserving $\mathbf{w}$ and $\mathbf{A}$. This *timestamp shuffling* P[$\mathbf{w}$,$\mathbf{t}$] is both a timeline shuffling and a snapshot shuffling, and it is thus compatible with link shufflings and sequence shufflings. P[$\mathbf{w}$,$\mathbf{t}$] is a very popular MRRM. It was named *permuted times* in Ref. [21], and called *time-shuffled* or *time-shuffling* in Refs. [3, 5, 11, 16, 30], *randomly permuted times* in Refs. [1, 19, 23, 36], *random dynamic* in Ref. [27], *random time shuffle* in Ref. [2], *reconfigure* in Ref. [13], and *shuffled time stamps* in Ref. [14]. It was also employed in Refs. [10, 22, 28, 39].

2. Event shufflings

The three following event shufflings are intersections between link shufflings and timeline shufflings. They are thus compatible with link shufflings (Subsec. IV C), timeline shufflings (Subsec. IV D), and sequence shufflings (Subsec. IV E).

P[$\mathcal{L}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] randomly permutes timelines between all links while keeping the static network G^{stat} fixed. P[$\mathcal{L}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] was named *link-sequence shuffled* in Refs. [3, 16, 30], *edge randomization* in Ref. [1], and *link shuffling* in Refs. [9, 17].

P[$\mathbf{w}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] permutes timelines $\Theta_{(i,j)}$ between links with (approximately) the same weight w. P[$\mathbf{w}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] was named *equal-weight link-sequence shuffled* in Refs. [3, 16, 30] and was also called *equal-weight edge randomization (EWER)* in Ref. [1].

P[$\mathbf{n}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] permutes timelines between links with the same contact frequencies $n_{(i,j)}$. P[$\mathbf{n}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] is a natural alternative P[$\mathbf{w}$,$p_{\mathcal{L}}(\mathbf{\Theta})$] to networks with event durations, where $\mathbf{n}$ has a similar role to $\mathbf{w}$ in networks with instantaneous interactions.

H. Compositions of two shufflings

We may combine two compatible MRRMs by composition (i.e. applying the second MRRM to the networks generated by the first) to randomize at different levels at the same time (see Section I C). For example, complete randomization of an instant-event temporal network, while keeping the number of links and events fixed, may be obtained by randomly permuting the links between all pairs of nodes and randomly permuting the instantaneous events on and between the links,. The resulting model is P[L,E], given by the composition of P[$p_{\mathcal{L}}(\mathbf{\Theta})$] and P[$\mathcal{L}$,$E$]. Since compatible MRRMs commute, it does not matter in which order we apply them.

We list only examples we have found in the literature of MRRMs that are compositions of two compatible MRRMs, all of which are compositions of link and timeline shufflings. The number of different MRRMs that may generated by composition is given by the number of combinations of compatible shufflings, and is thus much larger than this.

P[L,E] is generated by the composition of P[$p_{\mathcal{L}}(\mathbf{\Theta})$] and P[$\mathcal{L}$,$E$]. It completely randomizes both the topological and temporal structure of the network while constraining L. P[L,E] was called *all random* in Ref. [21].

P[$\mathbf{k}$,$p(\mathbf{w})$,$\mathbf{t}$] applies P[$p_{\mathcal{L}}(\mathbf{\Theta})$,$\mathbf{k}$] and P[$\mathbf{w}$,$\mathbf{t}$] in composition to the temporal network. The method was called *randomized edges with randomly permuted times* in Refs. [1, 36].

P[$\mathbf{k}$,$\mathbb{I}_{\lambda}$,$p(\mathbf{w})$,$\mathbf{t}$] adds the additional constraint to the above method that the aggregated network of the sampled reference networks should be connected. It is the composition of P[$p_{\mathcal{L}}(\mathbf{\Theta})$,$\mathbb{I}_{\lambda}$,$\mathbf{k}$] and P[$\mathbf{w}$,$\mathbf{t}$] and was called *configuration model* in Refs. [3, 16].

I. Randomization based on metadata

The availability of metadata offers the possibility to impose additional external constraints, allowing the study of effects that are not topological or temporal. For instance, in Ref. [11], the age, gender, and type of subscription of mobile phone users were known; in Ref. [27], the authors used randomization methods respecting the bipartite structure of a sex worker-buyer interaction network, and Ref. [17] used a randomization method that rewired links between each pair of predefined node groups in face-to-face networks.

These *metadata MRRMs* all rely on assigning a *color* to each node, i.e. to which group it belongs among a set of R predefined groups, fixed by the vector of node colors $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_N)$, where $\sigma_i \in \{1, 2, \ldots, R\}$. The $R \times R$ *contact matrix* $\Sigma_{\mathcal{L}}$ fixes the number of links between members of each group ($\Sigma_{\mathcal{L}}$ is necessarily symmetric for undirected networks). These two additional constraints enables us to define MRRMs that impose structure or dynamics determined by the metadata [58]. We may also directly use this to conserve the bipartite structure of a network as in [27] by imposing two groups and a perfectly antidiagonal $\Sigma_{\mathcal{L}}$, with $(\Sigma_{\mathcal{L}})_{11} = (\Sigma_{\mathcal{L}})_{22} = 0$ and $(\Sigma_{\mathcal{L}})_{12} = (\Sigma_{\mathcal{L}})_{21} = L$. We may additionally allow $\boldsymbol{\sigma}$ to vary over time and introduce a contact matrix $\boldsymbol{\Sigma}_{\mathcal{E}}$ that fixes the number of events taking place in each snapshot to capture temporal changes in the group structure. Then $\boldsymbol{\sigma} = (\boldsymbol{\sigma}^m)_{m=1}^T$, with $\boldsymbol{\sigma} = (\sigma_1^m, \sigma_2^m, \ldots, \sigma_N^m)$, and $\boldsymbol{\Sigma}_{\mathcal{E}} = (\Sigma_{\mathcal{E}}^m)_{m=1}^T$, where each $\Sigma_{\mathcal{E}}^m$ gives the number of events taking place between each group in snapshot m.

We classify below and in Table V MRRMs relying on metadata.

P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] permutes the links in the static graph while constraining the group appartenance of each node, $\boldsymbol{\sigma}$, and the number of links between each group, i.e. $\Sigma_{\mathcal{L}}$. It was employed in Ref. [17], where it was called *CM-shuffling*.

P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\mathbf{k}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] randomizes G^{stat} while constraining the group structure as P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] does with the additional constraint that the degrees $\mathbf{k}$ of the nodes be conserved. P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\mathbf{k}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] was used to constrain the bipartite structure of the network and was named *random topological* in Ref. [27].

P[$\mathbf{k}$,$p(\mathbf{w})$,$\mathbf{t}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] is generated by composition of the metadata reliant model P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\mathbf{k}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] with P[$\mathbf{w}$,$\mathbf{t}$]. It was named *random dynamic topological* in Ref. [27].

P[G,$p(\boldsymbol{\sigma})$,$\Sigma_{\mathcal{L}}$] permutes the group appartenances (colors) of the nodes at random (i.e. it randomizes the order of $\boldsymbol{\sigma}$). It thus destroys all correlations between node color and network structure and dynamics. It is equivalent to permuting the links while constraining the static graph to be isometric to the original static graph, and thus could also be named P[iso(G^{stat}),$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$], but we use the above name for conciseness. It was employed in Ref. [11], where it was called *node type shuffled data*.

J. Other reference models

We have above restricted ourselves to microcanonical RRMs as they are the only maximum entropy reference models that can be generated by shuffling elements of an empirical temporal network and they constitute the largest part of RRMs for temporal networks found in the literature.

In this section, we briefly discuss other types of reference models for temporal networks. These models can be divided into three general classes: (1) *canonical* RRMs, which correspond to generalized canonical ensembles of random networks defined by a generative model; (2) data-driven reference models that do not maximize entropy; (3) bootstrap methods, which are a particular, but important, type of reference models that do not maximize entropy.

1. Canonical randomized reference models

Canonical RRMs present alternatives that are very close in spirit to the microcanonical RRMs considered here. They permit to sample canonical ensembles of networks, i.e. ensembles where selected features are constrained only on average, $\langle \mathbf{x}(G) \rangle = \mathbf{x}(G^*)$, instead of exactly, $\mathbf{x}(G) = \mathbf{x}(G^*)$, as is the case for MRRMs. (One often talks of *soft* constraints for the canonical ensemble and *hard* constraints for the microcanonical ensemble). Such canonical generative models are also known as *exponential random graph models* (ERGMs) [51, 59] and allow to model the expected variability between samples (see discussion in [60, Section 4]). They are thus expected to have a lower generalization error than microcanonical RRMs. they may be incorporated in a Bayesian or information theoretic model selection framework [61, 62]. Additionally, due their soft constraints, canonical models are typically more amenable to analytical treatment than their microcanonical counterparts [63].

Conversely, the main advantage of MRRMs is that they are usually defined as data shuffling methods, which are often easier to construct than methods that generate networks from scratch. They are thus generally the only type of models that realistically capture many of the temporal and topological correlations present in empirical networks currently, which explains their popularity for analyzing temporal networks. In particular, it is easy to generate microcanonical RRMs that impose heuristically defined features such as the global activity timeline $\mathbf{A}$ or temporal correlations in individual timelines, which is difficult and often currently impossible to do using a generative model. Perhaps due to the difficulty in defining

TABLE V: **Effects of metadata shufflings on characteristics of temporal networks.** Special metadata symbols are the color (group appartenance) of a node , σ_i, and the contact matrix Σ, see Tables I and II for other symbol definitions. Note that care should be taken when interpreting results using this table since a feature that is not conserved (−) by a randomization procedure is not necessarily completely randomized either (see discussion at the end of Sec. II).

Canonical name	Meta		Aggregated								Temporal-topological									
			topological			weighted				temp.	node					link				
	Σ	g_i	G^{stat}	k_i	L	a_i	s_i	n_ℓ	w_ℓ	A^t	Γ^t	Φ_i	α_i^m	$\Delta\alpha_i^m$	d_i^t	Θ_ℓ	τ_ℓ^m	$\Delta\tau_\ell^m$	t_ℓ^1	t_ℓ^w
P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$]	x	x	−	μ	x	μ	μ	p	p	x	−	−	−	−	$\boldsymbol{\mu_{\mathcal{T}}}$	p	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\mathbf{k}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$]	x	x	−	x	x	μ	μ	p	p	x	−	−	−	−	$\boldsymbol{\mu_{\mathcal{T}}}$	p	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	p	p
P[$\mathbf{k}$,$p(\mathbf{w})$,$\mathbf{t}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$]	x	x	−	x	x	−	μ	−	p	x	−	−	−	−	$\boldsymbol{\mu_{\mathcal{T}}}$	−	−	−	−	−
P[G,$p(\boldsymbol{\sigma})$,$\Sigma_{\mathcal{L}}$]	x	p	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

generative reference models that capture empirical temporal correlations, these are currently almost exclusively defined for static networks or to model either memoryless dynamics [3, 20, 33, 64] or dynamics with limited temporal correlations [65–68] A notable exception is a recent study combining Markov chains with change point detection to model multiscale temporal dynamics [69]. We shall not discuss canonical RRMs in more detail here, but refer to [59] for a recent review of ERGMs for temporal networks and to [60, 70] for recent developments in such models for static networks.

2. *Reference models that do not maximize entropy*

Several reference models exist that impose a constraint that is not justified solely by the data (the empirical temporal network) in conjunction with the maximum entropy principle [42]. Such reference models thus introduce new order that is not found in the original network. Here we discuss different types of such reference models and give examples.

Delta function constraints. Some studies have considered reference models where what we may call a *delta function constraint* was imposed on a set of characteristics of the temporal network. Specifically they constrained all instances of this characteristic to have the same value, i.e. to follow a delta distribution. This is different from (and more constrained than) the maximum entropy distribution. The *SStat* method introduced in Ref. [29] imposes a fixed number of events in each snapshot (equal to the mean number of events per snapshot in the empirical network). Holme [31] introduced three reference models that all three impose a delta-function constraint (referred to as *poor man's reference models* since they do not satisfy the maximum entropy principle and provide only a single reference network instead of an ensemble [14]): equalizing the inter-event durations $\Delta\tau_\ell^m$ while constraining t_ℓ^1, t_ℓ^w and w_ℓ for each link $\ell \in \mathcal{L}$ (a non-MaxEnt version of P[$\mathbf{w}$, $\mathbf{t}^1$, $\mathbf{t}^w$]); shifting the whole sequence of events (sequences of event and inter-event times) on each link in order to make $t_\ell^1 = t_{\min}$ or to make $t_\ell^w = t_{\max}$ for all $\ell \in \mathcal{L}$.

Biased sampling. Kovanen et al. [5] proposed a biased version of P[$\mathbf{w}$,$\mathbf{t}$], where instead of swapping timestamps of events at random, for each event (i, j, t) they drew m other events at random from set of events $\mathcal{E}$ and swapped the timestamps of (i, j, t) and the other event (i', j', t') among the m drawn for which t' was closest to t. This reference model thus retains some temporal correlations due to the biased sampling, where the parameter m controls the force of this bias and thus of temporal correlations (for $m = 1$ the reference model is equal to P[$\mathbf{w}$,$\mathbf{t}$]). The same method was also employed in Refs. [8, 11]. Valdano et al. [13] considered a heuristic variant of P[$p_{\mathcal{T}}(\mathbf{g})$] (called *reshuffle-social*, where they only permuted snapshots inside intervals where nodes showed the same median *social strategy* [71], where the social strategy of a node i is defined as the ratio $\gamma_i^t = k_i^{\delta,t}/s_i^{\delta,t}$ of its degree $k_i^{\delta,t}$ and its strength $s_i^{\delta,t}$ in a network aggregated over $\delta = 20$ consecutive snapshots from $t - \delta\Delta t$ to t. The empirical temporal network that they investigated showed very clear spikes in γ_i^t separated by low-γ_i^t intervals, referred to as γ-slices, which allowed them to permute snapshots within each γ-slice only.

Time reversal. A quick but informal to gain insight into the role of causality in the contact dynamics is to simply reverse time [1, 5, 14, 36], i.e. reversing the order of the snapshots. This method obviously does not increase the entropy of the network as the time-reversed network is unique, but it may be used as a simple way to study the importance of causality in the temporal network. A *time-reversal* MRRM may in principle be defined as one that return an input temporal network and its time-reversed version with equal probability.

3. *Bootstrap methods*

Bootstrap methods are based on sampling with replacement, whereas microcanonical RRMs are based on sampling without replacement (i.e. permutation). Resampling with replacement means that network features are not constrained exactly as for permutation meth-

ods. The hope when using bootstrapping can thus be to capture some of the out-of-sample variability. However, the set of states that may be generated is strongly constrained by the particular dataset. So bootstrapping does not generate a maximum entropy model. Though it may be seen as a means to approximate one, it does not come with the same statistical guarantees as microcanonical RRMs and generative models do. So the nice theoretical results and guarantees that exist for microcanonical RRMs (see Sec. I) do not hold for bootstrapping, and additional care is advised when analyzing results obtained using bootstrapping.

Two bootstrap methods used in the literature are described below. The method called *time shuffling* in Ref. [9] constrains the number of events per link $\mathbf{n}$ exactly and resamples the event durations τ from the global distribution $p(\boldsymbol{\tau})$ with replacement. The method called *time shuffling* in Ref. [17] constrains the static network G^{stat} and bootstraps n_ℓ, t^1_ℓ for all links from the global distributions $p(\mathbf{n})$ and $p(\mathbf{t}^1)$, respectively, and then bootstraps the n_ℓ event durations τ^m_ℓ and $(n_\ell - 1)$ of inter-event durations $\Delta\tau^m_\ell$ for each link ℓ from the global distributions $p(\boldsymbol{\tau})$ and $p(\boldsymbol{\Delta\tau})$, respectively.

V. ANALYSIS AND HYPOTHESIS TESTING USING RANDOMIZED REFERENCE MODELS

In this section we outline a general procedure for using MRRMs in statistical analysis and hypothesis testing, and we provide a walk-through example of the use of series of comparable and compatible MRRMs to analyze an empirical temporal network dataset of face-to-face interactions in a primary school.

MRRMs permit us to perform null-hypothesis testing for temporal networks. Loosely speaking, they permit us to answer the question: can a given set of features alone explain the phenomenon we observe in the original temporal network? Furthermore, by using a nested series of (*comparable* and/or *compatible*) MRRMs, we may answer the questions: what is the effect of individual features and which feature is most important for the phenomenon we observe? For example, are heterogeneous distributions of inter-event durations and link weights enough to explain how a dynamical process propagates through a network and is the result significantly different from reference networks where only one of the distributions is heterogeneous? [3] Alternatively: is the original network significantly different from a random network with the same link weights and overall activity patterns? [11]

Practically speaking, hypothesis testing using a given MRRM, $\mathrm{P}[\mathbf{x}]$, builds on the same three general steps as used in permutation tests in classical statistics:

1. Calculate some summary statistic of the original (input) temporal network, $\mathbf{y}(G^*)$, e.g., a given feature of the network such as a marginal distribution of characteristics or the frequencies of given network motifs [11], or some statistic of a dynamical processes on the networks such as the distribution of arrival times of a simulated spreading process at each node [9] .

2. Apply a permutation method corresponding to $\mathrm{P}[\mathbf{x}]$, and calculate for the resulting randomized network G' the value $\mathbf{y}(G')$ of the same summary statistic as above. Repeat this step many times in order to sample $\mathcal{G}_{\mathbf{x}(G^*)}$ and obtain a *null distribution* of $\mathbf{y}$.

3. Compare the values of the summary statistic of the original network G^* to its null distribution. If the two do not agree, the reference model is highly unlikely as the sole explanation of the observed data and we may conclude that the constrained features are not enough to explain what we see in the original network. Conversely, if the two do agree, we may conclude that the constrained features can alone explain what we see in the original network.

The above procedure may also be used to compare two ensembles of randomized networks corresponding to different MRRMs in order to pinpoint the individual effects of the different features that are conserved/destroyed by the different models. More, generally, we may want to apply a series of hierarchically nested MRRMs and compare them in order to discern the effects of a range of network features.

Note finally that p-values are generally not very informative here. Due to the large size of typical datasets and many non-random features of empirical networks, p-values will typically be small and thus statistically significant even for the most constrained MRRMs. One is instead most often interested in computing and comparing effect sizes (e.g. how much faster/slower does a contagion process spread in the randomized networks?) and qualitative comparisions (e.g. does the distribution of inter-event durations have a broad tail or not?). Thus, we do not generally seek to reject or confirm null hypotheses about data, but rather to use MRRMs as an investigative tool to first flesh out qualitative effects linked to different features and second investigate quantitative effect sizes.

A. Walktrough example: analyzing face-to-face interactions using MRRMs

We analyze below a SocioPatterns dataset of face-to-face interactions in a primary school [72, 73], which is freely available at `www.sociopatterns.org/datasets`. Using the hierarchies of methods described in Figs. 8 and 9, we show how MRRMs can be used sequentially to destroy correlations in a systematic and ordered way, in order to investigate features of the temporal network.

We consider two classes of shufflings that are compatible: timeline shufflings which conserve the underlying

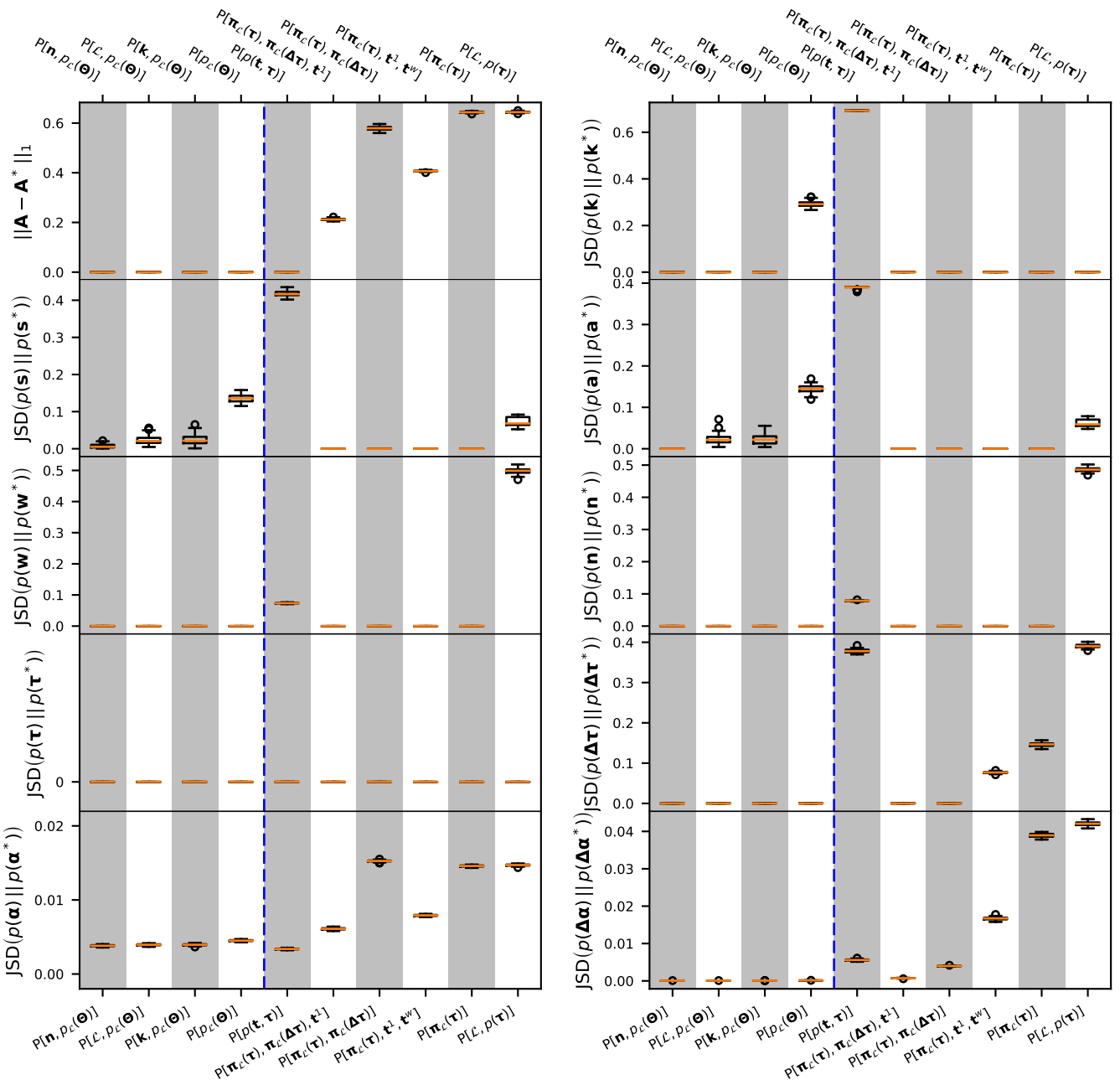


FIG. 12: Effects of the MRRMs on different features of the temporal network of interactions in a primary school. Each panel shows the difference between the value of the feature in the empirical network and its null distribution under each model. For the activity timeline $\mathbf{A}$, the difference is quantified as the L1-distance between the activity at each time. For all other features, the difference is quantified as the Jensen-Shannon divergence between the global distributions of the values of individual characteristics. The boxplots show the distribution of the difference over 100 different randomized networks generated by the MRRM in question.

static graph and shuffle the events in the timelines, and link shufflings which conserve the individual timelines of activity but shuffle the structure of the static graph. We present a selection of methods for each class, listed from the most to the least constrained (i.e. from finest to coarsest). We finally end with an example of a event shuffling which randomizes both the static topology and temporal dynamics of the network. This walk-through illustrates how a methodical exploration of the features of a temporal network using MRRMs can be used to build a statistical portrait of a temporal network.

The MRRMs employed here are:

a. Intersections of link and timeline shufflings:

- $\mathrm{P}[\mathbf{n}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$
- $\mathrm{P}[\mathcal{L}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$

See Section IV G for theoretical definitions and Appendix B for algorithmic implementations.

b. Link shufflings:

- $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta}), \mathbf{k}]$
- $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$

See Section IV C for theoretical definitions and Appendix B for algorithmic implementations.

c. Timeline shufflings:

- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}), \mathbf{t}^1]$
- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$
- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$
- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$
- $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$

See Section IV D for theoretical definitions and Appendix B for algorithmic implementations.

d. Event shuffling:

- $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$

See Section IV B for a theoretical definition and Appendix B for algorithmic implementation.

Figure 12 resumes our statistical portrait of the temporal network. It quantifies how each MRRM changes the distributions of a selection of temporal network characteristics as measured by the Jensen-Shannon divergence [74] between the null distributions and their distribution in the empirical network (For the activity timeline $\mathbf{A}$, the difference is quantified by the L1 distance). (The values of the features for each MRRM is shown in Figs. 13 and 14.)

We note first that the distribution of event durations is conserved by construction by all MRRMs since they all are event shufflings. Furthermore, most of the MRRMs conserve the distributions of cumulative durations and event frequencies on the links, $p(\mathbf{w})$ and $p(\mathbf{n})$, respectively, with the expected exceptions of $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$ (which conserves the static structure, but not the heterogeneity in the number and durations of events in timelines) and $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$ (which does not even conserve the number of links in G^{stat}).

The last link feature we investigate is the distribution of inter-event durations $p(\boldsymbol{\Delta\tau})$. Comparison of the effects of $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$ and $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$ demonstrates that the timing of the first and last events in the timelines constrain the inter-event durations to some degree in the network (see also Fig. 14).

The results become more interesting when we look at the effects on node characteristics. The effects on the degree distribution of nodes in the static network, $p(\mathbf{k})$, are the least surprising as $p(\mathbf{k})$ is constrained by most of the MRRMs, with the exception of $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$ (which draws G^{stat} from an Erdős-Rényi model) and $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$ (which does not conserve the number of links in G^{stat}).

As was the case for $p(\mathbf{w})$ and $p(\mathbf{n})$, the effects of the MRRMs on the distributions of node strengths and activities, $p(\mathbf{s})$ and $p(\mathbf{a})$, are very similar, so we take $p(\mathbf{a})$ as example and note that everything we discuss also applies to $p(\mathbf{s})$. The distribution of a_i generated by $\mathrm{P}[\mathcal{L}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$ and $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta}), \mathbf{k}]$ are almost indistinguishable, which can be explained by k_i and $n_{(i,j)}$ being largely uncorrelated. Furthermore, comparison of the effects of $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$ and $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$ shows that $p(\mathbf{k})$ (constrained by $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$) is more important than $p(\mathbf{n})$ (constrained by $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$) for the shape of $p(\mathbf{a})$ in the empirical network.

None of the MRRMs completely constrain the distributions of the nodes activity and inactivity durations, $p(\boldsymbol{\alpha})$ and $p(\boldsymbol{\Delta\alpha})$. However, all link shufflings produce null distributions that are very close to the empirical ones, indicating that the temporal correlations of the individual links' activity strongly constrain the nodes' activity. More surprisingly, the small difference observed in $p(\boldsymbol{\alpha})$ between the link shufflings and $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$ as compared to the other MRRMs points at the global timing of the events as the most important temporal feature in determining the node activity durations in the network; even more important than the distributions of event and inter-event durations on the links together, and even though $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$ does not even conserve the number of links in the static graph of the network. Conversely, we see that the distribution of inter-event durations, $\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$, is the most important temporal feature in determining the nodes' inactivity durations, $p(\boldsymbol{\Delta\alpha})$, while the timing of the events comes in second order (compare $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}), \mathbf{t}^1]$ and $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$ to $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$ and $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$).

Finally, the activity timeline $\mathbf{A}$ is by construction completely constrained by all link shufflings and the snapshot shuffling $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$, while at the opposite end it is completely randomized by $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$ and $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$ (see Fig. 14). Comparison between the two latter and $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$ shows that the distribution of inter-event durations does not significantly constrain $\mathbf{A}$. Comparing this with $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$ shows that the timing of the first and last events on each link does on the other hand have a significant effect on $\mathbf{A}$ in the network. Finally, comparison with $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}), \mathbf{t}^1]$ shows that constraining both $\mathbf{t}^1$ and $\mathbf{t}^w$ together with $\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$ imposes an even stronger constraint on the activity timeline $\mathbf{A}$ (see also Figure 14).

As seen in Figs. 13 and 14, the distributions of the different characteristics obtained from a single randomized network generally vary little around their median, even though the empirical network is of relatively modest size (242 nodes and 77 521 events were recorded). Fur-

thermore, the median distributions converge rapidly to their expectation with the number of generated samples (Figs. 15 and 16).

VI. APPLICATIONS OF RANDOMIZED REFERENCE MODELS

The applications of MRRMs for temporal networks are manifold, but all follow two main directions: (i) studying how the network and ongoing dynamical processes are controlled by the effects of temporal and structural correlations that characterize empirical temporal networks, and (ii) highlighting statistically significant interaction motifs in temporal networks.

(i) Dynamical processes have been studied by using data-driven models, where temporal interactions are obtained from real data, while the ongoing dynamical process is modeled by using any conventional process definition [1, 76] and typically simulated numerically on the empirical and randomized temporal networks [76, 77]. One common assumption in all these models is that information can flow between interacting entities only during their interactions. This way the direction, temporal, and structural position, duration, and the order of interactions become utmost important from the point of view of the dynamical process. MRRMs provide a way to systematically eliminate the effects of these features and to study their influence on the ongoing dynamical process. This methodology has recently shown to be successful in indicating the importance of temporality, bursty dynamics, community structure, weight-topology correlations, and higher-order temporal correlations on the evolution of dynamical processes, just to mention a few examples.

(ii) MRRMs have also commonly been used as null models to find statistically significant interaction motifs in temporal networks or correlations between the network dynamics and node attributes. This approach is conceptually the same as using the Maslov-Sneppen model to detect frequent motifs in static networks [78, 79]. The difference here is that the studied networks vary in time, which induces further challenges.

In the first three subsections (Subsecs. VI A–VI C), we review studies applying MRRMs to study various dynamical processes in empirical temporal networks. In the fourth part of this section (Subsec. VI D) applications to inferring statistically significant motifs and correlations in network dynamics will be discussed briefly. Finally, in the last subsection (Subsec. VI E) we discuss a pair of recent papers that have applied MRRMs to study temporal network controllability. We will in the following include data-driven reference models that are not maximum entropy models (i.e. the reference models discussed in Subsecs. IV J 2–IV J 3). While they do not come with the same formal statistical guarantees as maximum entropy RRMs do, they have nevertheless been useful in identifying important features in temporal networks.

A. Contagion processes

Contagion phenomena is the family of dynamical processes that has been studied the most using MRRMs. Since epidemics, information, or influence are all transmitted by person-to-person interactions (to a large extent), the approximation provided by contact-data-driven simulations are indeed closer to reality than other conventional methods based solely on analytical models. MRRMs became important in this case to help understand which temporal or structural features of real temporal networks control the speed, size, or the critical threshold of the outbreak of any kind of epidemic process. In the following we will address various types of contagion dynamics ranging from simple to complex spreading processes, focusing on findings that are due to MRRMs. For detailed definitions and characteristics of the different contagion processes we refer readers to the recent review by Pastor-Satorras et al. on the topic [76].

1. SI process

The susceptible-infected (SI) process is the simplest possible contagion model. Here nodes can be in two mutually exclusive states: susceptible (S) or infectious (I). Susceptible nodes (initially everyone except an initial seed node) become infected with rate β when in contact with an infected node. This process does not display a phase transition since each node (belonging to the seeded component of the network) becomes infected in the end (almost surely), i.e. the fraction of infected nodes $\langle I(t)\rangle/N$ tends to 1. The single parameter β controls the speed of saturation, thus by considering the limit $\beta \to \infty$ one can simulate the fastest possible contagion dynamics on a given network. In this case the infection times correspond to the temporal distances between the seed and the nodes that get infected. This can be seen as a "light-cone" defining the horizon of propagation in the temporal network [21].

Early motivation to use RRMs of temporal networks was to understand why models of information diffusion unfold extremely slowly in communication networks even when modeled by the fastest possible spreading model, i.e. an SI process with $\beta \to \infty$ [3]. In this study two mobile communication networks and an email network were taken as temporal networks to study this phenomenon [75]. The study introduced four MRRMs and measured the average fraction of infected nodes to study the early and late time behavior of the spreading process. As compared to the diffusion on the original sequence (which takes about 700 days for full penetration) the fastest null model was P[$\mathbb{I}_\lambda$,$\mathbf{k}$,$p(\mathbf{w})$,$\mathbf{t}$], removing all structural and temporal correlations while keeping only the empirical heterogeneities in the node degrees, $\mathbf{k}$, and link weights, $p(\mathbf{w})$, as well as the cumulative activity over time, $\mathbf{A}$). At the same time, the largest contribution to the overall acceleration effect appeared once applying the

P[$\mathbf{w}$, $\mathbf{t}$] and P[$\mathcal{L}$,$p_{\mathcal{L}}(\boldsymbol{\Theta})$] models. This lead to the conclusions that the bursty interaction dynamics and the Granovetterian weight-topology correlations [81] are dominantly responsible for the slow spreading of information in these systems. On the other hand the P[$\mathbf{w}$,$p_{\mathcal{L}}(\boldsymbol{\Theta})$] model, which eliminates all causal correlations between events taking place on adjacent links but conserves the weighted network structure and temporal correlations in individual timelines, was observed to slightly slow down the process during the early phase, while accelerating it in the long run. As an alternative possible explanation, effects of circadian fluctuations, were also studied here via two generative reference models where interaction times were generated by either a homogeneous or an inhomogeneous Poisson process (with the rate of creation of new events set equal to either the mean (time-averaged) or the instantaneous rate of event creation, respectively). These models thus conserved $\langle w_{(i,j)} \rangle$ and $(\langle w_{(i,j)} \rangle, \langle A^t \rangle)$, respectively. These generative models demonstrated that although circadian fluctuations may cause short term fluctuations in the overall speed of the spreading process, on the longer temporal scale they have negligible effects on the spreading dynamics.

Three other null models were introduced by Kivelä et al. [30] to study the same process on the same empirical networks. They applied the MRRM P[$\mathbf{w}$] to randomize all temporal correlations while conserving the aggregated structure, and the MRRMs P[$\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})$] and P[$\mathbb{I}_\lambda, p_{\mathcal{L}}(\boldsymbol{\Theta})$] to randomize the static network topology while conserving all temporal correlations in individual timelines. They concluded that, while temporal correlations have much stronger effects on the dynamics, the heterogeneous degree distribution of the underlying social structure initially accelerates the spreading while slowing it down on the long run. Additionally, their main conclusion was that the slow diffusion can be partially explained by the timings of individual call sequences. The spreading is strongly constrained by the frequency of interactions, i.e. the high variance in the inter-event durations causes the residual waiting (relay times) times to be large, and makes the spreading slower than the Poissonian case.

Another study by Gauvin et al. [9] analyzed face-to-face interaction sequences and employed MRRMs to identify the effective dynamical features, responsible for driving the diffusion of epidemics in local settings like schools, hospitals, or scientific conferences. To understand the dominant temporal factors driving the epidemics in these cases, they took both a bottom-up approach by using generative network models, and a top-down approach by employing two shuffling methods (corresponding to MRRMs) and a bootstrap method. They shuffled event and inter-event durations on individual links using P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$,$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$], they shuffled the timelines between existing links using P[$\mathcal{L}$,$p_{\mathcal{L}}(\boldsymbol{\Theta})$], and they finally bootstrapped the global distribution of event durations $p(\boldsymbol{\tau})$ while keeping the number of events $\mathbf{n}$ on each link fixed. In this study, time was not taken as a global measure but interpreted to be node specific. Each node was assigned with an *activity clock* measuring the time that a node spent in interaction with others. This way, for an SI process, which was initiated from a seed node i at time t and which reached node j at time t_j, the arrival time of node j was not defined as $t_j - t$ but the cumulative duration of all of j's events during the period from t to t_j. Simulating the SI process this way, they measured the distribution of arrival *activity* times, defined as the time it takes for the epidemic, started from a seed, to reach a node in the network (measured using the nodes activity clock). To compare different null models they calculated Kullback-Leibler divergences between the corresponding arrival-time distributions. From these measurements they concluded that the bursty nature of interaction dynamics has the strongest effect on the speed of spreading, while the heterogeneity in the number of events per link and the synchronized contact patterns (typical in a school during breaks) also have a strong effect on the contagion dynamics.

Perotti et al. [57] studied the effect of temporal sparsity, an entropy based measure quantifying temporal heterogeneities on the empirical scale of average inter-event durations $\langle \Delta\tau^m_{(i,j)} \rangle$. As a reference model the authors used P[$\mathbf{w}$]. They showed via the numerical analysis of several temporal datasets and using analytical calculations that there is a linear correspondence between the temporal sparsity of a temporal network and the slowing down of a simulated SI process.

A unique temporal interaction dataset was studied by Rocha et al. [27], which recorded the interaction events of sex sellers and buyers in Brazil. The system is a temporal bipartite network where connections may only exist between sellers and buyers. Using this dataset the authors studied, among other questions, the effects of temporal and structural correlations on simulated SI (and SIR) processes. They introduced three different MRRMs imposing a bipartite network structure, which may be obtained using metadata MRRMs (Section IV I) with two groups and a perfectly anti-diagonal contact matrix between the groups. Their first model, P[$\mathbf{k}$,$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$], was used to destroy any structural correlations in the bipartite structure while keeping temporal heterogeneities unchanged (except for link-link temporal correlations). Conversely, their second null model, P[$\mathbf{w}$,$\mathbf{t}$], destroyed all the temporal structure except global activity patterns, but kept the weighted (bipartite) network structure unchanged. Their third model, P[$\mathbf{k}$,$p(\mathbf{w})$,$\mathbf{t}$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] was generated as the composition of the two others. Interestingly, they observed that bursty patterns accelerate the spreading dynamics, contrary to other studies [3, 9, 28, 30, 57]. At the same time they showed that structural correlations slow down the dynamics in the long run, and by applying the two reference models at the same time, that bursty temporal patterns and structural correlations together slows spreading initially and speeds it up for later times. The authors arrived at the same conclusion using SIR model dynamics (for definition

see Section VI A 2 below), with the additional observation that temporal effects cause relatively high epidemic thresholds as compared to degree-heterogeneous static networks, where thresholds are vanishing [76]. Note that the accelerating effect of burstiness in this case was explained later by the non-stationarity of the temporal network [31, 35]

2. SIR and SIS processes

The Susceptible-Infected-Recovered (SIR) and Susceptible-Infected-Susceptible (SIS) processes are two other dynamical processes that have been widely studied on temporal networks using MRRMs. In addition to the S→I transition of the SI process, in the SIR (SIS) process infected nodes transition spontaneously to a recovered, R (susceptible, S), state with rate γ (or after a fixed time θ), after which they cannot (can) be re-infected again. These processes are characterized by the basic reproduction number $R_0 = \beta/\gamma$ and display a phase-transition between a non-endemic and an endemic phase. An analogy with information diffusion can easily be drawn, where the infection is associated to the exposure to a given information, while spontaneous recovery mimics that the agent later forgets the given information.

One of the first studies addressing SIR dynamics using MRRMs was published by Miritello et al. [28] and investigated mobile phone communication networks. In their model they used β as the control parameter while letting the recovery time θ for each node be constant. They used two reference models. The first was the P[**w**,**t**] model, used to study the effects of bursty interaction dynamics on global information spreading by shuffling event timestamps. Their second null model applied a local shuffling scheme that cannot evidently be interpreted as a MRRM for networks since it considers only local information and not the whole network. It investigates the effects of group conversations on local information spreading. In this case they considered an event $(i \to j)$ and its preceding one $(* \to i)$ reaching the node i. To eliminate local causal correlations between the two events they randomized relay times by selecting randomly a time for the $(* \to i)$ event from the times of any event observed in the dataset. Both reference models preserve the link weights **w**, the duration of interactions, and also the circadian rhythms of human communications. As their first conclusion, they realized that relay times depend on two competing properties of communication. While burstiness induces large transmission times, thus hindering any possible infection, casual interaction patterns translate into an abundance of short relay times, favoring the probability of propagation. They also showed that the outbreak size of the simulated SIR process depends counterintuitively on β. More precisely, if β is small the spreading is faster and reach a larger fraction of nodes in the original temporal network than in the P[**w**,**t**] model. While on the contrary, if β is large the process evolves slower and unfolds in smaller cascades on the original data relative to the P[**w**,**t**] model. If β is large, the information propagates easily but its is affected strongly by large inter-event durations and local correlations, while if β is small the propagation is more successful at the local scale thus reaching a larger fraction of nodes in the original temporal network even if temporal correlations are present. To quantitatively explain these effects the authors introduce the *dynamical tie strength* to represent the network as static and show that the phenomena can be explained by the competition of heterogeneous interaction patterns and local causal correlations.

Génois et al. studied the effects of sampling of face-to-face interaction data on data-driven simulations of SIR (and SIS) processes [17], and proposed an algorithm for compensating for the sampling effect by reconstructing surrogate versions of the missing contacts from the incomplete data, taking into account the network group structure and heterogeneous distributions of $n_{(i,j)}$, $\tau^m_{(i,j)}$, and $\Delta\tau^m_{(i,j)}$. Using the reconstructed data instead of the sampled data allowed to trade in a large underestimation of the epidemic risk by a small overestimation; here the epidemic risk was quantified by the fraction of recovered (susceptible) nodes in the stationary state and the probability that the epidemics reached at least 20% of the population. They used MRRMs to investigate and explain the reasons for the small overestimation of the epidemic risk when using the reconstructed networks. They applied following reference models: the time shuffling P[$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})$,$\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})$]; the metadata link shuffling P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$]; a bootstrap method, resampling $p(\mathbf{n})$, $p(\boldsymbol{\tau})$, $p(\boldsymbol{\Delta\tau})$, and $p(\mathbf{t}^1)$; and finally applied the metadata MRRM P[$p_{\mathcal{L}}(\boldsymbol{\Theta})$,$\boldsymbol{\sigma}$,$\Sigma_{\mathcal{L}}$] in composition with the bootstrap method. This allowed them to conclude that the overestimation was due to higher order temporal and structural correlations in the empirical temporal networks, which however are notoriously hard to quantify and to model.

The effect of birth and death of links on epidemic spreading was demonstrated by Holme and Liljeros [31] using twelve empirical temporal networks. They investigated an ongoing link picture where the lifetime of social ties is irrelevant as links are assumed to be created and end before and after the observation period; and a link turnover picture where social links are assigned with a lifetime being created and dissolved during the observation. To understand which case is more relevant for modeling epidemic spreading, they defined three deterministic *poor man's reference models* [14] (see Sec. IV J 2). Their first reference model conserved $\mathbf{t}^1$, $\mathbf{t}^w$, and **w**, and equalized all inter-event durations in the timelines, eliminating the effects of heterogeneous inter-event durations. Their second and third models aimed to neutralize the effects of the beginning and ending times of active intervals, thus they shifted the active periods of each link either to the beginning or to the end of the observation period, i.e. they set $t^1_{(i,j)} = t_{\min}$ or $t^w_{(i,j)} = t_{\max}$

for all $(i,j) \in \mathcal{L}$, respectively, while keeping the original sequence of inter-event durations, $\boldsymbol{\Delta\tau}$, on the links. The authors presented an exhaustive analysis by simulating SIR and SIS processes on each dataset using the original event sequences, and each reference model. They explored the entire phase space in each case. They concluded that for both processes, while shifting activity periods (either way) induce large differences in the final fraction of infected nodes, equalizing the inter-event durations while keeping the times of the first and last events on each link only marginally changes the outcome. This indicates that it is enough to consider only the observed lifetime of links while their fine-grained dynamics is less relevant in terms of modeling spreading processes.

Valdano et al. [13] proposed an infection propagator approach to compute the epidemic threshold of discrete time SIS (and SIRS) processes on temporal networks. Their aim was to account for more realistic effects, namely a varying force of infection per contact, the possibility of waning immunity, and limited time resolution of the temporal network. To better understand the effects of temporal aggregation and correlations on the estimation of the epidemic threshold they used three different MRRMs, as well as a heuristic reference model, and applied them on face-to-face interaction datasets recorded in school settings. The three MRRMs being: P[$p_{\mathcal{T}}(\mathbf{\Gamma})$], P[$\mathbf{w}$,$\mathbf{t}$], and P[$\mathbf{iso}(\mathbf{\Gamma})$]. They measured, for different recovery rates, how the epidemic threshold changed as a function of the aggregation time window relative to the case with the highest temporal resolution. They considered two different aggregation strategies: where the link weights were (i) or were not (ii) considered. They showed that the obtained thresholds were mostly independent of the cumulative activity of the network, and more related to specific time-evolving topological structures. Finally, they considered a fourth heuristic reference model, which shuffled the snapshot order, but only within a given number of slices, this way keeping control on the length of temporal correlations destroyed (see description in Sec. IV J 2). They showed that long range temporal correlations, which in turn lead to repeated interactions and strong weight-topology correlations, must be considered to provide a good approximation of the epidemic threshold on short temporal scales and for slow epidemics.

Finally, there has been a single study using MRRMs with rumor spreading dynamics [32]. It considered the Daley-Kendall model, which is very similar to the SIR model with the exception that nodes do not recover spontaneously but via interactions with other infected or recovered (stifler) nodes. The aim of this study was to understand the effects of memory processes, inducing repeated interactions between people, on the global mitigation of rumors in large social networks. Using a mobile phone communication dataset they utilized a directed temporal network MRRM, P[$\mathbf{d}_{\rightarrow}$], which constrained the instantaneous out-degree $d^m_{i\rightarrow}$ of each node in each snapshot (see Appendix A). In practice this amounted to randomizing the called person for each event in order to eliminate the effects of repeated interactions over the same link. This MRRM randomized the topological and temporal correlations in the network, destroyed link weights, and increased the static node degrees considerably. Results were confronted with corresponding model simulations, which verified that memory effects play the same role in data-driven models as was observed in the case of synthetic model processes, namely they keep rumors local due to repeated interactions over strong ties.

3. Threshold models

A third family of spreading processes are *complex contagion* processes, which are often used to model social contagion. These models capture the effects of social influence, which is considered via a non-linear mechanism for contaminating neighboring nodes (typically a threshold mechanism). In the conventional definition of threshold models [82] nodes can be in two mutually exclusive states, non-adopter (i.e. susceptible) – initially all but one node – and adopter (i.e. infectious) – initially a randomly selected *seed* node – and each node i is assigned a threshold ϕ_i defining the necessary number k_i^I of adopter neighbors, or fraction k_i^I/k_i, in order to make the node (with degree k_i) adopt; we refer to the first variant as the Watts threshold model with *absolute* thresholds, and the second as the Watts threshold model with *relative* thresholds. The central question here is the condition needed to induce a large adoption cascade that spreads all around the network. These models are highly constrained by the network structure and dynamics as the distribution of individual thresholds determine the conditions for global cascades. This is fundamentally different from SIR type of dynamics (called *simple contagion processes*) which are highly stochastic, driven by random infection and recovery. In the latter case, transmission of infection is not fully determined by structural properties but possible even via a single stimuli coming from an infected neighbor. The conventional threshold model introduced by Watts [82], and other related dynamical processes have been thoroughly studied on static networks, however their behavior on temporal networks has been addressed only recently by studies using RRMs.

Karimi and Holme [10] studied two different threshold models on six empirical datasets of time-resolved human interactions. They studied both the relative threshold and the absolute threshold Watts models [82], where the numbers of adopted neighbors $k_i^I(\delta)$ and all neighbors $k_i(\delta)$ were calculated over a given *memory* time window δ. Their main goal was to identify the effect of temporal and structural correlations on the size of the emerging cascade as function of the threshold ϕ (chosen to be equal for every node) and the memory time window size. For the model with fractional thresholds they observed that the cascade size decreased with ϕ and with the time window length, while for the absolute threshold

model it increased with longer time windows. They furthermore employed two MRRMs: P[**w**,**t**] and P[$p_{\mathcal{L}}(\mathbf{\Theta})$]. They found that in the fractional threshold model temporal correlations (burstiness) allowed smaller cascades to evolve in most of the cases, while in the absolute threshold model the effect was the contrary. An exception they found was a conference setting, where temporal correlations increased the cascade size, while structural correlations slightly decreased it. As they explained, this may be due to specific constraints in this setting as bursty interaction patterns appeared synchronously during the conference breaks where also a large number of simultaneous interactions appeared between people discussing in groups.

Backlund et al. [2] also studied the effects of temporal correlations on cascades in threshold models on temporal networks. In their study, they introduced a stochastic and a deterministic threshold model. Their stochastic model is a linear threshold model where the probability of adoption increases linearly with the fraction of adopting neighbors observed in a finite time window prior to the actual interaction. Note that in this case, rapidly repeated interactions with an adopted neighbor does not increase the adoption probability per interaction. However, since adoption potentially occurs after every interaction, bursty interaction patterns evidently affects the adoption process. Conversely, in their deterministic model they employed the conventional deterministic threshold rule, thus assigning a relative threshold to a node (the same for each of them, as in Ref. [10]), which then certainly adopts after this fraction of adopted neighbors has been reached within a finite observation window. Note that in each model when calculating the actual threshold of a node they considered the static degree k_i, aggregated over the whole observation period, in the denumerator, and not the degree $k_i(\delta)$, aggregated over the time window δ only as in [10]. They applied two MRRMs to four different temporal interaction datasets. They used the P[**w**,**t**] model to destroy all temporal correlations while keeping circadian fluctuations, and introduced another model, P[**per**($\mathbf{\Theta}$)], that randomly shifts each individual timeline using periodic boundary conditions to keep all temporal correlations inside each timeline and destroy correlations between events on adjacent links as well as circadian fluctuations. After simulating both models, they found that increasing the memory length (time window size) facilitates spreading, and so does the removal of temporal correlations using P[**w**,**t**]. This way they concluded that burstiness negatively affects the size of the emerging cascades. At the same time, they found that higher order temporal-structural correlations, removed by P[**per**($\mathbf{\Theta}$)], facilitate the emergence of large cascades. In addition, they observed that for the deterministic model, high degree nodes tend to block the spreading process, contrary to the case of simple contagion. For complex contagion, hubs are unlikely to interact with enough adopters to reach their adoption threshold.

A somewhat different picture was proposed by Takaguchi et al. [19], where the authors used a threshold model denoted *history dependent contagion*. This model is an extension of an SI process with threshold mechanisms. Here each node has an internal variable measuring the concentration of pathogen and is increased by unity after a stimuli arrived via temporal interactions with infected neighbors. However, this concentration decays exponentially as function of time in the absence of interaction with infected nodes. A node becomes infected if its actual concentration reaches a given threshold, after which it remains in the infected state. They simulated this model on two different temporal interaction networks and measured the fraction of adopters as function of time. In order to identify the effects of bursty interaction patterns they used the P[**w**,**t**] model, which lead to a slower spreading dynamics. From this they argued that burstiness facilitates the speed of spreading in both datasets. Furthermore, they showed through the analysis of single link dynamics, that this acceleration was mostly due to the bursty patterns on separate links and not due to correlations between bursty events on adjacent links or to the overall structure of the network.

B. Random walks

Random walks are some of the simplest and most studied dynamical processes on networks. On a temporal network, a random walk is defined by a walker, which is located at a node at time t, and can be re-located to the node's current neighbors in each timestep. The walker choses the neighbor to which it jumps either at random or with a probability proportional some link weight. A central measure is here the *mean-first passage time* (MFPT), defined as the average time taken by the random walker to arrive for the first time at a given node, starting from some initial position in the network. Another important measure is the *coverage* defined as the number of different vertices that have been visited by the walker up to time t.

Starnini et al. [29] studied stationary properties of random walks on temporal networks, and used reference models to define ways to synthetically extend their temporal face-to-face interactions datasets with limited observation length. They assumed periodic temporal boundary conditions on their empirical temporal network (their first model), with weak induced biases as discussed in an earlier paper [83]. Their second model, P[**w**], kept all weighted characteristics of the aggregated network, but destroyed all temporal correlations and induced Poissonian interaction dynamics. Finally, they introduced a third heuristic reference model in which they impose a delta function constraint on the number of events starting at each time step (see Sec IV J 2), randomly drawing the pairs of nodes that interact in order to approximately conserve **n** and finally bootstrap the event durations from $p(\boldsymbol{\tau})$. This approximately conserves certain important statistical properties of the empirical event se-

quence, namely $p(\mathbf{n})$ and $p(\boldsymbol{\tau})$, but not $\mathbf{A}$ and $p(\boldsymbol{\Delta\tau})$. After providing a mean field solution, they measured the MFPT and coverage on the original and synthetic sequences. They found that the results for empirical sequences deviated systematically from the mean field prediction and from the results for the reference models, inducing a slowdown in coverage and MFPT. They concluded that this slowdown is not due to the heterogeneity of the durations of conversations, but uniquely due to what they term *temporal correlations* (which, given the reference models they tested, encompasses the time-varying cumulative activity, the broad distribution of inter-event durations, and higher-order temporal correlations between different events).

Delvenne et al. [33] also addressed random walks on temporal networks. They used MRRMs in order to understand which factor is dominant in determining the relaxation time of linear dynamical processes to their stationary state. They introduced a general formalism for linear dynamics on temporal networks, and showed that the asymptotic dynamics is determined by the competition between three factors: a structural factor (communities) associated with the spectral properties of the Laplacian of the static network, and two temporal factors associated to the shape of the waiting-time distribution, namely its burstiness coefficient (defined in [84]) and the decay rate of its tail. They demonstrated their methodology on six empirical temporal interaction networks and used two RRMs. The link shuffling $\mathrm{P}[\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$ aimed to remove the effects of the structural correlations. In this case they found that in sparse networks, structure remains the dominant determinant of the dynamics as sparsity results in the inevitable creation of bottlenecks for diffusion even in a random network. On the other hand, in denser structures the removal of communities leads to the dominance of temporal features. The other null model, a generative reference model using a homogeneous Poisson process to generate events and constraining only G^{stat} and the mean number of events $\langle E \rangle$ (see Sec. IV J 1), destroyed all temporal and weight correlations while conserving the static network structure, leading to the evident dominance of the network structure in regulating the convergence to stationarity.

A greedy random walk process and a non-backtracking random walk process were studied by Saramäki and Holme on eight different human interaction datasets in Ref. [34]. A greedy random walker always moves from the occupied node to one of its neighbors whenever possible. Thus its dynamics is more sensitive to local temporal correlations in the network. A non-backtracking greedy random walker is additionally forbidden to return to its previous position. Thus, it is forced to move to a new neighbor or wait until the next event which moves it to a new neighbor. The authors studied what types of temporal correlations are determinant during these dynamics by using a the $\mathrm{P}[\mathbf{w},\mathbf{t}]$ model and measuring the coverage of the walker after a fixed number of moves. They found that after removing temporal correlations using $\mathrm{P}[\mathbf{w},\mathbf{t}]$, the walker reached considerably more nodes. They explained this observation by the dominant effects of bursty (ping-pong) event trains on single links which trap the walker for a long time going back and forth between two nodes. In addition, they also indicated somewhat weaker effects of larger temporal motifs such as triangles. They finally traced the entropy of the greedy walkers and concluded that, on average, the entropy production rates measured in the original event sequences were lower than for randomized data, indicating more predictable node sequences of visited nodes in the empirical case.

C. Evolutionary games

Evolutionary games [85] define another set of dynamical processes which have historically been studied on networks. They are analogous to several social dilemmas where the balance of local and global payoffs drive the decision of interacting agents. Any agent may choose between two strategies (Cooperation or Defection) and can receive four different payoffs (Reward, Punishment, Sucker, or Temptation). The relative values of Temptation and Sucker determines the game, where players update their strategy depending on the state of their neighbors with a given frequency and tend to find an optimal strategy to maximise their benefits.

Cardillo et al. [12] studied various evolutionary games on temporal networks and asked two questions: "Does the interplay between the time scale associated with graph evolution and that corresponding to strategy updates affect the classical results about the enhancement of cooperation driven by network reciprocity?" and "what is the role of the temporal correlations of network dynamics in the evolution of cooperation?". They analyzed two human interaction sequences, and for comparison they applied the snapshot shuffling $\mathrm{P}[p_{\mathcal{T}}(\boldsymbol{\Gamma})]$, and the activity-driven generative network model [64] (see Sec. IV J 1). As a parameter to control the time-scale of the network, they varied the integration time window size defining a single snapshot of the temporal networks and measured the fraction of cooperators after the simulated dynamics reached equilibrium. They showed for all social dilemmas studied that cooperation is seriously hindered when agent strategies are updated too frequently relative to the typical time scale of interactions, and that temporal correlations between links are present and lead to relatively small giant components of the graphs obtained at small aggregation intervals. However, when one use randomized or synthetic time-varying networks preserving the original activity potentials but destroying temporal correlations, the structural patterns on the reference networks change dramatically. Effects of the temporal resolution on cooperation are smoothed out, and due to the lack of temporal and structural correlations, cooperation may persistently evolve even for moderately small time periods.

D. Temporal motifs and networks with attributes

Another direction of application of RRMs is to highlight significant temporal correlations of motifs in interaction signals or when interactions may correlate with additional node attributes.

For directed temporal networks, one simple application of MRRMs was introduced by Karsai et al. [24], who analyzed the correlated activity patterns of individuals, which induced bursty event trains. They found that the number of consecutive events arriving in clusters are distributed as a power-law. To identify the reason behind this observation they used a very simple MRRM that shuffled the inter-event durations between consecutive event pairs, $\mathrm{P}[\mathbf{s}_{\rightarrow}, p(\boldsymbol{\Delta\tau})]$ (see Appendix A for definitions of features of directed temporal networks). They found that in the shuffled signal, bursty event trains were exponentially distributed, which evidently indicated that bursty trains were induced by intrinsic correlations in the original system and were not simply due to the broad distribution of inter-event durations.

In another study, Karsai et al. [25] also applied this framework to identify whether correlated bursty trains of individuals is a property of nodes or links. Using a large mobile phone call interaction dataset, the observation was made that bursty train size distributions were almost the same for nodes and links. This suggests that such correlated event trains were mostly induced by conversations by single peers rather than by group conversations. To further verify this picture, the fraction of bursty trains of a given size emerging between a varying number of individuals were calculated in the empirical event sequence and in a directed network MRRM, $\mathrm{P}[\mathbf{w},p(\boldsymbol{\Delta\tau})]$, where the receivers of calls were shuffled between calls of the actual caller. This reference model leaves the timing of each event unchanged, thus leading to the observation of the same bursty trains, and keeps the instantaneous and static out-degrees of individuals. However, since the receivers were shuffled, potential correlations that induce bursty trains on single links were eliminated. Results show that the fraction of single link bursty trains drops from $\sim 80\%$ to $\sim 20\%$ after shuffling in call and SMS sequences. This supports the hypothesis that single link bursty trains are significantly more frequent than one would expect from the null hypothesis, which is then rejected.

However, real temporal networks commonly reveal more complicated temporal motifs, whose detection was first addressed by Kovanen et al. [5]. They proposed a method to identify mesoscale causal temporal patterns in interaction sequences where events of nodes do not overlap in time. This framework can be used to identify overrepresented patterns, called temporal motifs which are not only similar topologically but also in the temporal order of the events. They propose different RRMs to quantify the significance of different temporal motifs. They used $\mathrm{P}[\mathbf{w},\mathbf{t}]$, and they introduced a non-MaxEnt reference model which biases the sampling of the ensemble of temporal networks defined by $\mathrm{P}[\mathbf{w},\mathbf{t}]$ in order to keep some temporal correlations in the sequence (see Sec. IV J 2). To do so, they selected randomly for each event in a motif m other events from the sequence and chose the one which was the closest in time to the original event in focus. If $m = 1$ the model is identical to $\mathrm{P}[\mathbf{w},\mathbf{t}]$, while the larger m is the more candidate events there are, thus the more likely it is to find one close to the original event. They furthermore suggested that to remove causal correlations from the sequence, one may simply reverse the interaction sequence and repeat the motif detection procedure (see Sec. IV J 2). They used these reference models in the same spirit as configuration model networks were used to identify static network motifs, which are typically overrepresented in real networks as compared to random structures [78, 79]. Here applying the complete and partial shuffling as null models on motifs consisting of three events, they found that motifs between two nodes, i.e. bursty link trains, are the most frequent, and motifs which consist of potentially casually correlated events are more common.

In another study by the same authors, the same methodology was used [11] to identify motifs in temporal networks where nodes (individuals) were assigned with metadata attributes like gender, age, and mobile subscription types. Beyond the $\mathrm{P}[\mathbf{w},\mathbf{t}]$ model, the authors introduced the metadata MRRM $\mathrm{P}[G, p(\boldsymbol{\sigma}), \Sigma_{\mathcal{L}}]$, which shuffled single attributes between nodes. In addition, they applied the biased version of $\mathrm{P}[\mathbf{w},\mathbf{t}]$ introduced in [5] (see Sec. IV J 2), which accounts for the frequency of motif emergence in the corresponding static weighted network without considering node attributes. Using this non-MaxEnt reference model and the two MRRMs they found gender-related differences in communication patterns and showed the existence of temporal homophily, the tendency of similar individuals to participate in communication patterns beyond what would be expected on the basis of their average interaction frequencies.

The dynamics of egocentric network evolution was studied by Kikas et al. [86], where they used a large evolving online social network to analyse bursty link creation patterns. First of all they realized that link creation dynamics evolve through correlated bursty trains. They verified this observation by comparing the distribution of inter-event durations (measured between consecutive link creation events) to those generated by the MRRM $\mathrm{P}[\mathbf{s}_{\rightarrow}, p(\boldsymbol{\Delta\tau})]$, where inter-event durations were randomly shuffled. In addition, they classified users based on their link creation activity signals (where activity was measured as the number of new link added within a given month). They showed that bursty periods of link creations are likely to appear shortly after a user account creation, or when a users actively use free or payed services provided by the online social service. In order to verify these correlations they used a RRM where they shuffled link creation activity values between the active months of a given user and found considerably weaker correlations between the randomized link creation

activity signals and service usage activity signals of people.

Finally, in a different framework, a special kind of metadata reference model was also used by Karsai et al. [87] to demonstrate whether the effect of social influence or homophily is dominating during the adoption dynamics of online services on static networks. This reference model did not consider randomizing the temporal networks, but rather node attributes linked to the dynamics of the game (i.e. a purely metadata MRRM – Sec. IV I – coupled with a dynamical process on the network); we include it in this survey to demonstrate the scope of maximum entropy permutation methods beyond randomizing network features solely. In case of real adoption cascades, these two mechanisms may lead to similar collective adoption patters at the system level. In reality, influence-driven adoption of an ego can happen once one or more of its neighbors have adopted, since their actions may then influence the decision of the central ego. Consequently, the time-ordering of adoptions of the ego and its neighbors matters in this case. Homophily-driven adoption is, however, different. Homophily drives social tie formation such that similar people tend to be connected in the social structure. In this case connected people may adopt because they have similar interests, but the time ordering of their adoptions would not matter. Therefore, it is valid to assume that adoption could evolve in clusters due to homophily, but these adoptions would appear in a random order. In order to demonstrate these differences, the authors used a reference model where they shuffled all adoption times between adopted nodes and confronted the emerging adoption rates of innovator, vulnerable, and stable adopters (for definitions see [82, 87]) to the adoption rates observed in the empirical system. They found that after shuffling the rate of innovators considerably increased, while the rate of influence driven (vulnerable and stable) adoptions dropped. This verified that adoption times matters during real adoption dynamics, thus the social spreading process was dominantly driven by social influence. Note that in this case the network was static and shuffling was applied on the observed dynamical process. We mention this example here to demonstrate the potential of MRRMs in other settings.

E. Network controllability

We finally mention two recent studies of the controllability of temporal networks that have leveraged MRRMs. Control of a dynamical system aims at guiding a system to a desired state by designing the inputs to the system [88]. Although control theory has a long history as a branch of engineering applied to diverse subjects, it was only recently that we saw a general theory of the controllability of the systems in which elements interact in a networked manner [89]. The key finding was that sparse networks require more driver nodes (i.e. the nodes receiving the designed input) than dense networks, and that the driver nodes are not necessarily hubs in general [89]. An algorithm to approximate the minimal set of driver nodes was also proposed in [89], based on finding the maximum matching in the network.

It is natural to think of extending the theory for static networks to temporal networks. Pósfai and Hövel made the first study in this direction, in which they considered a discrete-time linear dynamical system with time-varying interactions [18]. The focal measure of controllability is the size of the structural controllable subspace. The structural controllable subspace is defined by the subset of nodes satisfying that any of their final states at time t is realizable from any initial state in at most a number τ of time-steps by appropriately tuning the non-zero elements of the adjacency and input matrices as well as the input signals. First, they proved a theorem stating that a node subset is a structural controllable subspace if and only if any node in the subset are connected to disjoint time-respecting paths from the nodes receiving the input signals. This theorem implies that, keeping the same average instantaneous degree, the temporal network with a heavy-tailed $\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d})$ is more difficult to control than a network with a homogeneous $\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d})$ because the presence of hubs in snapshots decreases the number of disjoint time-respecting paths. They examine this theoretical argument by comparing the structural controllable subspace for an empirical temporal network to the ensembles produced by following the four MRRMs: P[$\mathbf{w}$,$\mathbf{t}$], P[$p_{\mathcal{T}}(\boldsymbol{\Gamma})$,$\mathbf{sgn}(\mathbf{t})$], P[$\mathbf{t}$], and P[$\mathbf{d}$]. The sizes of the maximum structural controllable subspace for P[$\mathbf{w}$,$\mathbf{t}$] and P[$\mathbf{t}$] were generally larger than that for the original network. This result suggests that the homogenization of $\boldsymbol{\pi}_{\mathcal{T}}(\mathbf{d})$ and thus the elimination of hubs in snapshots increases the controllability of networks, which is consistent with the theoretical argument. For the other two MRRMs, the controllability of P[$\mathbf{d}$] is almost the same as the original network and P[$p_{\mathcal{T}}(\boldsymbol{\Gamma})$,$\mathbf{sgn}(\mathbf{A})$] has a slightly lower controllability than the original network. These results implies that the higher-order structural correlations in snapshots have little effect on network controllability and that the temporal correlations over successive snapshot present in the original network contribute to enhance the controllability to some extent.

Recently, Li et al. [36] showed that temporal networks have a fundamental advantage in controllability compared to their static network counterparts. They compared the time and energy required to achieve the full controllability of the network, when driving nodes in the sequence of snapshots or the single aggregated network. The numerical experiments on multiple empirical networks demonstrated that temporal networks can be fully controlled more efficiently in terms of both time and energy than their static counterparts. They argued that this advantage comes from temporality itself, but not from particular temporal features, by showing that a set of different MRRMs, namely P[$\mathbf{w}$,$\mathbf{t}$], P[$\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})$], and P[$\mathbf{k}$,$\mathbf{w}$,$\mathbf{t}$], as well as the time reversal reference model (Sec. IV J 2), achieve more efficient controllability than

their aggregated counterparts.

VII. CONCLUSION

We have here introduced a fundamental framework for microcanonical randomized reference models (MRRMs). This enables consistent naming, analysis, and classification of MRRMs for temporal networks. We have used this framework to describe numerical shuffling procedures found in the literature rigorously in terms of microcanonical RRMs, built a taxonomy of these RRMs, and surveyed their applications to the study of temporal networks. This framework also allowed us to define conditions for when we may combine two MRRMs in a composition to generate a new MRRM and to derive which features it inherits from them. Such compositions of compatible MRRMs makes it possible to easily generate hundreds of new MRRMs from the dozens existing ones.

Many MRRMs are easily implemented by numerical shuffling methods. As such MRRMs provide an outstanding and very generally applicable toolbox for the analysis of dynamical networked systems. Their main advantages being their wide applicability and relative ease of implementation: they only require the definition of a corresponding unbiased permutation method. This means that they can be used to test the importance of any given feature, provided a corresponding permutation method, and may in principle be used to generate model networks that are arbitrarily close to empirical ones.

Shuffling methods provide an interesting alternative to more elaborate generative models, and can be seen as a *top-down* approach to modeling by progressively randomizing features of an empirical network, as opposed to the *bottom-up* approach of generative models. Each approach has its strengths and weaknesses (as discussed in Sec. IV J 1). We believe that shuffling methods are best used as exploratory tools to identify important qualitative characteristics and effects. Generative models can then be used to explore them quantitatively and to perform model selection in order to identify potential underlying generative mechanisms.

We have focused on undirected and unweighted temporal networks, but the extension of the basic microcanonical RRM framework introduced in Section I to any other type of network is trivial as it can be applied to any finite structures. Extending the rest of the framework may require defining new ways of representing the structure and defining appropriate features. This is straightforward for directed (see Appendix A) and edge-valued temporal networks. Temporally varying multilayer networks [90] require the definition of three-level nested features. Further, an MRRM-based framework can be developed for any other types of multilayer networks, such as multiplex networks or networks of networks. Finally, it should be helpful to define a similar framework for canonical reference models (see Section IV J 1) as more of such models are emerging.

It is our intention that this framework and collection of MRRMs will serve as a reference for researchers who want to employ MRRMs to analyze temporal networks and dynamical processes taking place on them. Notable important challenges remain which can now be addressed using the solid foundations posed here. For example, how to automatize the definition and classification of new MRRMs, which would allow a user to simply state the set of features she wants to constrain to generate a corresponding MRRM ensemble of networks. How to automatize the choice of MRRMs in order to most efficiently infer which features of an empirical temporal network control a given dynamical phenomenon, i.e. identifying which models best divide the space of network features (corresponding to cuts in the dependency diagram, Fig. 7). How to characterize automatically the effect of a MRRM on temporal network features that are not comparable to nor independent of the features constrained by the MRRM.

We note finally that it is a notoriously difficult problem to design unbiased permutation methods for many microcanonical RRMs for networks [26, 91], especially MRRMs that take higher order topological correlations into account, which may put natural barriers on the possible resolution of exact MRRMs. Instead, approximate procedures for generating such MRRMs would have to be considered, and their accuracy may be gauged by how closely they reproduce features which we know should be constrained by the MRRM.

ACKNOWLEDGEMENTS

L.G. acknowledges support from the Lagrange Laboratory of the ISI Foundation funded by the CRT Foundation. M.G. was supported by the French ANR HARMSflu, ANR-12-MONU-0018. M. Karsai acknowledges support from the DylNet (ANR-16-CE28-0013) and SoSweet (ANR-15-CE38-0011) ANR projects and the MOTIf STIC-AmSud project. T.T. was supported by the JST ERATO Grant Number JPMJER1201, Japan. C.L.V. was supported by the EU FET project MULTIPLEX 317532. M.G, T.T., and C.L.V. acknowledge financial support through the Bilateral Joint Research Program between MAEDI, France, and JSPS, Japan (SAKURA Program).

AUTHOR CONTRIBUTIONS

C.L.V. conceived and directed the study. All authors contributed to the definition of the classification system, to classifying existing MRRMs found in the literature, and to writing. M. Kivelä, T.T., and C.L.V. performed theoretical calculations. M.G. and M. Kivelä wrote the Python/C++ software packages. M.G., M. Karsai, M. Kivelä, T.T., and C.L.V. drafted the final manuscript.

Appendix A: Randomized reference models for directed temporal networks

Two of the studies in the survey above (Sec. VI) considered MRRMs specifically defined for directed temporal networks, namely P[$\mathbf{d}_{\rightarrow}$] and P[$\mathbf{s}_{\rightarrow}, p(\boldsymbol{\Delta\tau})$]. The inclusion of MRRMs for directed networks is straightforward as it simply requires defining the appropriate directed features (and characteristics). For characteristics of links, no generalizations are necessary since they all generalize automatically to directed networks by using the convention that (i, j) designates an interaction from i to j. However, since in directed networks a link from i to j does not imply the presence of the reciprocal link from j to i, the interpretation of link features may change. For each characteristic of nodes, three generalizations typically exist: an outgoing version, e.g., the out-strength $s_{i\rightarrow}$, an ingoing version, e.g., the in-strength $s_{i\leftarrow}$, and a combined version, e.g., the total strength $s_i = s_{i\rightarrow} + s_{i\leftarrow}$. We list some generalizations of node characteristics to directed temporal networks in Table VI.

Appendix B: Implementation of algorithms and pseudocode

In this appendix we describe the implementation of a selected list of randomization methods, along with their name in the Python library and plots showing their effect on different statistics of the temporal network. The methods are separated into three classes: link, timeline, and event shufflings.

In the figures, features are plot in black and circles for the original network, and in color and triangles for the randomized network. The color indicates whether the feature is constrained by the method (blue) and thus kept identical, or not (red). Note that the fact that a feature is not constrained does not necessarily mean that it will be completely different: correlations between features might act as effective constraints.

In the pseudocode, the original temporal network is designated as `DATA`, lists have an `L_` prefix and dictionaries a `D_` prefix. If needed, the value of the duration of one time step is given by the variable `dt`, and the boundaries of the time window of the `DATA` is given by `ti` and `tf`.

1. Link shufflings

- P[$\mathbf{n}, p_{\mathcal{L}}(\boldsymbol{\Theta})$]

Description: The set of timelines $\boldsymbol{\Theta}$ is extracted and each timeline $\Theta_{(i,j)}$ is associated with the number of events $n_{(i,j)}$ it contains. Pools P_n of timelines with the same n are built, and then for each link (i, j) a timeline Θ is drawn from the corresponding pool $P_{n_{(i,j)}}$.
Library function: `P__n_pTheta()`
Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the timeline of each link
D_timeline = timelines(DATA)
#extraction of the number of events of each
  link
D_n = number_of_events(DATA)
#construction of the pools
D_pool = {n:[]}
for link in L_links:
  n = D_n[link]
  timeline = D_timeline[link]
  append timeline to D_pool[n]
#shuffling
for link in L_links:
  n = D_n[link]
  new_timeline = choice(D_pool[n])
  remove new_timeline from D_pool[n]
  update DATA[link] with new_timeline
return DATA
```

- P[$\mathcal{L}, p_{\mathcal{L}}(\boldsymbol{\Theta})$]

Description: The set of timelines $\boldsymbol{\Theta}$ is extracted. Each timeline $\Theta_{(i,j)}$ is then randomly associated with a link $(i, j) \in \mathcal{L}$ in the static graph G^{stat}.
Library function: `P__L_pTheta()`
Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the list of timelines
L_timelines = list_timelines(DATA)
#shuffling
for link in L_links:
  new_timeline = choice(L_timelines)
  remove new_timeline from L_timelines
  update DATA[link] with new_timeline
return DATA
```

- P[$\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})$]

Description: The set of timelines $\boldsymbol{\Theta}$ is extracted. The links are then randomized while keeping the aggregated degree $\mathbf{k}$ of each node unchanged using a configuration model technique. Timelines $\Theta_{(i,j)}$ are finally randomly assigned to the new links.
Library function: `P__k_pTheta()`
Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the list of timelines
L_timelines = list_timelines(DATA)
#extraction of the list of nodes
L_nodes = list_nodes(DATA)
#calculation of the degree of each node
D_degree = degree(List_links)
#sorting the nodes according to their degree
  in decreasing order
sort(L_nodes, D_degree, reverse=True)
#variable to handle unsolvable finalisations
  of the shuffling
redo = True
while redo:
  initialize NEW_DATA
```

TABLE VI: **Additional features of directed temporal networks.** Generalizations of node features in Table I to directed networks. Link characteristics are the same as for undirected networks. Below, $(\cdot)$ denotes an ordered sequence, $\{\cdot\}$ denotes an unordered set, $|\cdot|$ denotes the cardinality of a set, : means "for which" or "such that".

Symbol	*Meaning of symbol*	*Definition*
Topological-temporal characteristics		
$\mathcal{V}_{i\rightarrow}$	Outgoing neighborhood of node.	$\mathcal{V}_{i\rightarrow} = \{j : (i,j) \in \mathcal{L}\}$
$\mathcal{V}_{i\leftarrow}$	Incoming neighborhood of node.	$\mathcal{V}_{i\leftarrow} = \{j : (j,i) \in \mathcal{L}\}$
$\mathcal{V}_i$	Neighborhood of node.	$\mathcal{V}_i = \{j : (i,j) \in \mathcal{L} \text{ or } (j,i) \in \mathcal{L}\}$
Topological-temporal characteristics		
$d^t_{i\rightarrow}$	Instantaneous out-degree.	$d^t_{i\rightarrow} = \lvert\{j : (i,j) \in \mathcal{E}^t\}\rvert$
$d^t_{i\leftarrow}$	Instantaneous in-degree.	$d^t_{i\leftarrow} = \lvert\{j : (j,i) \in \mathcal{E}^t\}\rvert$
d^t_i	Instantaneous (total) degree	$d^t_i = \lvert\{j : (i,j) \in \mathcal{E}^t \text{ or } (j,i) \in \mathcal{E}^t\}\rvert$
$\Phi_{i\rightarrow}$	Node activity timeline.	$\Phi_{i\rightarrow} = \left((v^1_{i\rightarrow}, \alpha^1_{i\rightarrow}), (v^2_{i\rightarrow}, \alpha^2_{i\rightarrow}), \ldots, (v^{a_{i\rightarrow}}_{i\rightarrow}, \alpha^{a_{i\rightarrow}}_{i\rightarrow})\right)$
$\alpha^m_{i\rightarrow}$	Activity duration.	Consecutive interval during which i has at least one outgoing contact.
$\Delta\alpha^m_{i\rightarrow}$	Inactivity duration.	$\Delta\alpha^m_{i\rightarrow} = v^{m+1}_{i\rightarrow} - (v^m_{i\rightarrow} + \alpha^m_{i\rightarrow})$
Aggregated characteristics		
$a_{i\rightarrow}$	Outgoing node activity.	$a_{i\rightarrow} = \sum_{j\in\mathcal{V}_{i\rightarrow}} n_{(i,j)}$
$a_{i\leftarrow}$	Intgoing node activity.	$a_{i\leftarrow} = \sum_{j\in\mathcal{V}_{i\leftarrow}} n_{(i,j)}$
a_i	(Total) node activity.	$a_i = \sum_{\ell\in\mathcal{L}_i} n_{(i,j)}$
$s_{i\rightarrow}$	Node out-strength.	$s_{i\rightarrow} = \sum_{\ell\in\mathcal{L}_{i\rightarrow}} w_{(i,j)}$
$s_{i\leftarrow}$	Node in-strength.	$s_{i\leftarrow} = \sum_{\ell\in\mathcal{L}_{i\leftarrow}} w_{(i,j)}$
s_i	Node (total) strength.	$s_i = \sum_{\ell\in\mathcal{L}_i} w_{(i,j)}$
$k_{i\rightarrow}$	Node out-degree.	$k_{i\rightarrow} = \lvert\mathcal{V}_{i\rightarrow}\rvert$
$k_{i\leftarrow}$	Node in-degree.	$k_{i\leftarrow} = \lvert\mathcal{V}_{i\leftarrow}\rvert$
k_i	Node (total) degree.	$k_i = \lvert\mathcal{V}_i\rvert$

```
    redo = False
    #local copies in case of redo
    L_loc_nodes = copy(L_nodes)
    L_loc_timelines = copy(L_timelines)
    D_loc_degree = copy(D_degree)
    #loop on all the nodes
    while L_loc_nodes is not empty:
      #choice of the node with higher degree
      n = L_loc_nodes[0]
      remove n from L_loc_nodes
      k = D_loc_degree[n]
      #test of solvable reconstruction
      if len(L_loc_nodes) >= k:
        D_loc_degree[n] = 0
        #sampling of the new neighbors
        neighbors = sample(L_loc_nodes,k)
        #creation of the new links
        for p in neighbors:
          link = (n,p)
          timeline = choice(L_timelines)
          remove timeline from L_timelines
          update NEW_DATA[link] with timeline
          #update of the unattributed half-links
          D_loc_degree[p] = D_loc_degree[p] - 1
        #update of the list of available nodes
        remove all n from L_loc nodes if
    D_loc_degree[n] = 0
        sort(L_loc_nodes, D_loc_degree, reverse=
    True)
      #case when the reconstruction cannot be
    done
      else:
        L_loc_nodes = []
        redo = True
  return NEW_DATA
```

- $\mathrm{P}[p_{\mathcal{L}}(\boldsymbol{\Theta})]$

Description: All links are shuffled. Timelines $\Theta_{(i,j)}$ are then randomly assigned to the new links.

Library function: `P__pTheta()`

Pseudocode:

```
  #extraction of the list of links
  L_links = list_links(DATA)
  #extraction of the list of timelines
  L_timelines = list_timelines(DATA)
  #extraction of the list of nodes
  L_nodes = list_nodes(DATA)
  #shuffling
  L_new_links = []
  initialize NEW_DATA
  for link in L_links:
    n,p = sample(L_nodes,2)
    new_link = (n,p)
    while new_link in L_new_links:
      n,p = sample(L_nodes,2)
      new_link = (n,p)
    new_timeline = choice(L_timelines)
    remove new_timeline from L_timelines
    update NEW_DATA[new_link] with new_timeline
  return NEW_DATA
```

2. Timeline shufflings

- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}), \mathbf{t}^1]$

Description: Event durations τ and inter-event durations $\Delta\tau$ are shuffled in place on each timeline $\Theta_{(i,j)}$ while preserving the starting time t^1 of the first event on the timeline.

Effet:

Library function: `P__pitau_pidtau_t1()`

Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the events
D_events = events(DATA)
#extraction of the inter-event durations
D_events = interevents(DATA)
#shuffling
for link in L_links:
  #local lists of events and inter-event
  durations
  L_tau = D_events[link]
  L_dtau = D_interevents[link]
  #starting time of the first event
  t0 = DATA[link][0]
  #construction of the new timeline
  L_new_tau = shuffle(L_tau)
  L_new_dtau = shuffle(L_dtau)
  t = t0
  tau = L_new_tau.pop()
  new_timeline = [(t,tau)]
  t = t + tau
  while L_new_tau <> []:
    dtau = L_new_dtau.pop()
    t = t + dtau
    tau = L_new_tau.pop()
    append (t,tau) to new_timeline
    t = t + tau
  update DATA[link] with new_timeline
return DATA
```

- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$

Description: Event durations τ and inter-event durations $\Delta\tau$ are shuffled in place on each timeline $\Theta_{(i,j)}$. The starting time t^1 is randomly set so that the timeline fits within the window of the `DATA`.

Effet:

Library function: `P__pitau_pidtau()`

Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the events
D_events = events(DATA)
#extraction of the inter-event durations
D_events = interevents(DATA)
#shuffling
for link in L_links:
  #local lists of events and inter-event
  durations
  L_tau = D_events[link]
  L_dtau = D_interevents[link]
  #remaining usable time
  tu = tf - ti - sum(L_tau) - sum(l_dtau)
  #new initial time
  t0 = randint(0,tu/dt)*dt
  #construction of the new timeline
  L_new_tau = shuffle(L_tau)
  L_new_dtau = shuffle(L_dtau)
  t = t0
  tau = L_new_tau.pop()
  new_timeline = [(t,tau)]
  t = t + tau
  while L_new_tau <> []:
    dtau = L_new_dtau.pop()
    t = t + dtau
    tau = L_new_tau.pop()
    append (t,tau) to new_timeline
    t = t + tau
  update DATA[link] with new_timeline
return DATA
```

- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, t^w]$

Description: Event durations τ are shuffled in place on each timeline Θ while preserving the starting time t^1 of the first event and the ending time t^w of the last event.

Effet:

Library function: `P__pitau_t1_tw()`

Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the events
D_events = events(DATA)
#shuffling
for link in L_links:
  #local lists of events
  L_tau = D_events[link]
  nC = len(L_tau)
  #temporal parameters
  t1 = DATA[link][0]
  tw = DATA[link][-1] + L_tau[-1]
  #remaining usable time
  tu = tw - t1 - sum(L_tau)
  #construction of the new timeline
  L_new_tau = shuffle(L_tau)
  #first event
  tau1 = L_new_tau.pop()
  new_timeline = [(t1,tau1)]
  #cas plus d'un event
  if nC > 1:
    #last event
    tauw = L_new_tau.pop()
    append (tw-tau_w,tauw) to new_timeline
    #other events
    L_new_t = [randint(1,tu/dt)*dt for tau in
  L_new_tau]
    sort L_new_t
    #variable for the offset of the new times
    delta = t1 + tau1
    for tau in L_new_tau:
      t = L_new_t.pop(0) + delta
      append (t,tau) to new_timeline
      delta = delta + tau
  update DATA[link] with new_timeline
return DATA
```

- $\mathrm{P}[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$

Description: Event durations τ are shuffled in place on each timeline Θ.

Effet:

Library function: `P__pitau()`

Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the events
D_events = events(DATA)
#shuffling
for link in L_links:
  #local lists of events
  L_tau = D_events[link]
  nC = len(L_tau)
  #temporal parameters
  t1 = DATA[link][0]
  tw = DATA[link][-1] + L_tau[-1]
  #remaining usable time
  tu = tf - ti - sum(L_tau)
  #construction of the new timeline
  L_new_tau = shuffle(L_tau)
  #new time stamps
  L_new_t = [randint(0,tu/dt)*dt for tau in
  L_new_tau]
  sort L_new_t
  #variable for the offset of the new time
  stamps
  delta = 0
  for tau in L_new_tau:
    t = L_new_t.pop(0) + delta
    append (t,tau) to new_timeline
    delta = delta + tau
  update DATA[link] with new_timeline
return DATA
```

- $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$

Description: All events are shuffled among existing links, while retaining their duration τ. The events are set with random starting times, with a non overlapping condition.

Effet:

Library function: `P__L_ptau()`

Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the list of events
L_events = list_events(DATA)
#shuffling
initialize NEW_DATA
D_test_t = {}
for c in L_events:
  new_link = choice(L_links)
  t0 = randint(ti/dt,tf/dt)*dt
  t1 = t0 + c.duration
  #extended event to test for overlap
  loc_c = range(ti-dt,tf+dt,dt)
  test = [is t in D_test_t[new_link], for all
  t in loc_c]
  while test contains at least one True:
    new_link = choice(L_links)
    t0 = randint(ti/dt,tf/dt)*dt
    t1 = t0 + c.duration
    loc_c = range(ti-dt,tf+dt,dt)
    test = [is t in D_test_t[new_link], for
  all t in loc_c]
  update NEW_DATA[new_link] with c
  append loc_c to D_test_t[new_link]
return NEW_DATA
```

3. Event shufflings

- $\mathrm{P}[p(\mathbf{t}, \boldsymbol{\tau})]$

Description: All events are shuffled among all the possible links, while retaining there starting time t and duration τ. We implement a condition of non-overlapping between the events on the same link.

Effet:

Library function: `P__pttau()`

Pseudocode:

```
#extraction of the list of links
L_links = list_links(DATA)
#extraction of the list of nodes
L_nodes = list_nodes(DATA)
#extraction of the list of events
L_events = list_events(DATA)
#shuffling
initialize NEW_DATA
D_test_t = {}
for c in L_events:
  n,p = sample(L_node,2)
  new_link = (n,p)
  t0 = c.time
  t1 = c.time + c.duration
  #extended event to test for overlap
  loc_c = range(t0-dt,t1+dt,dt)
  if new_link in NEW_DATA:
    test = [is t in D_test_t[new_link], for
  all t in loc_c]
    while test contains at least one True:
      n,p = sample(L_node,2)
      new_link = (n,p)
      t0 = c.time
      t1 = c.time + c.duration
      loc_c = range(t0-dt,t1+dt,dt)
      if new_link in NEW_DATA:
        test = [is t in D_test_t[new_link],
  for all t in loc_c]
      else:
        test = [False]
    if new_link in NEW_DATA:
      update NEW_DATA[new_link] with c
    else
      D_test_t[new_link] = []
      add new_link,c to NEW_DATA
  else
    D_test_t[new_link] = []
    add new_link,c to NEW_DATA
  append loc_c to D_test_t[new_link]
return NEW_DATA
```

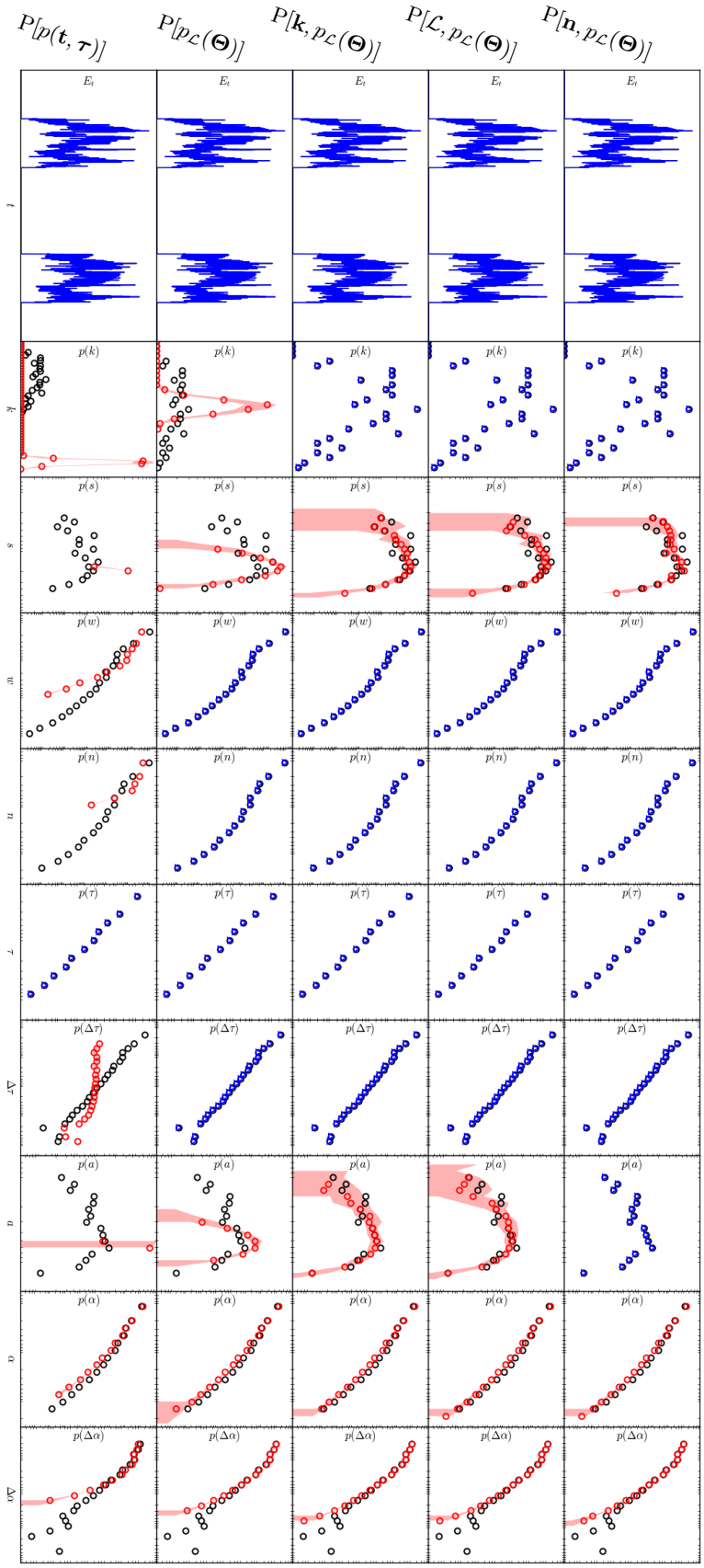


FIG. 13: **Effects of different link and event shufflings on temporal network features.** Values of a selection of features in the empirical face-to-face interaction network considered in Section V A and in randomized networks generated from it. Original data is in black. Randomized data is in blue if constrained, in red if not. Red lines are medians over 100 randomizations, red areas show 90 % confidence intervals.

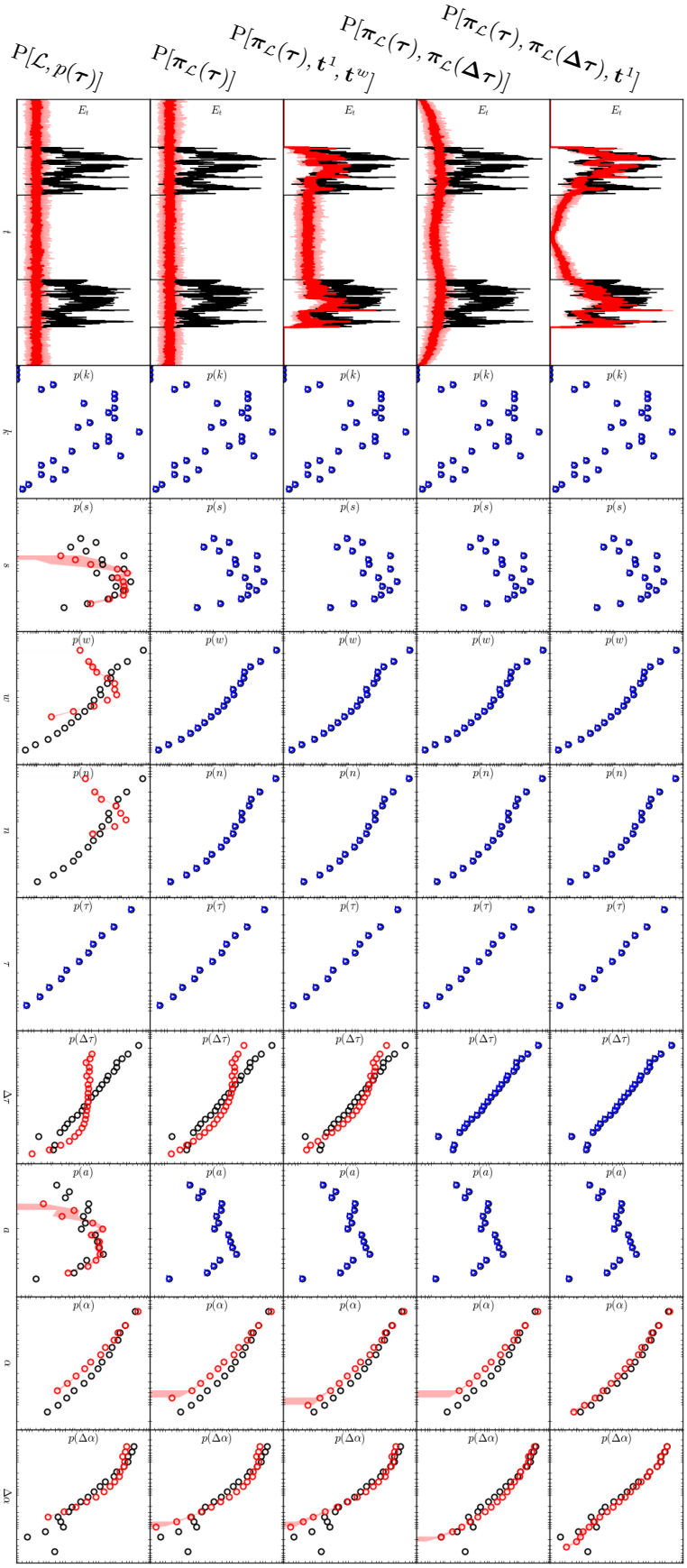


FIG. 14: **Effects of different timeline shufflings on temporal network features.** Values of a selection of features in the empirical face-to-face interaction network considered in Section V A and in randomized networks generated from it. Original data is in black. Randomized data is in blue if constrained, in red if not. Red lines are medians over 100 randomizations, red areas show 90 % confidence intervals.

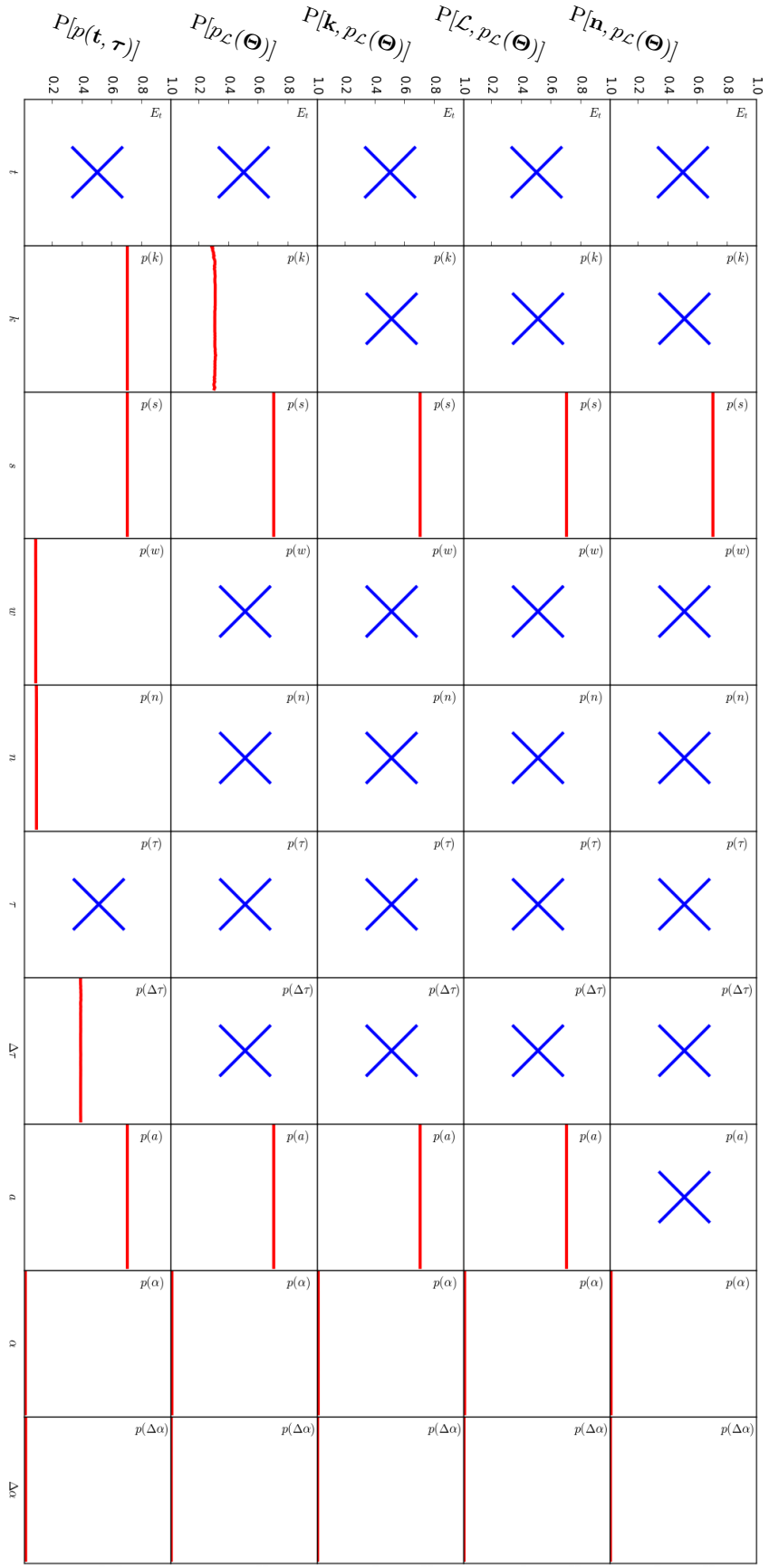


FIG. 15: **Convergence of the distributions generated by link and event shufflings and shown in Fig. 13.** We realized 100 randomizations of the network, and compute the evolution of the distance (as the Jensen-Shannon divergence) between the measure on the data and the median measure on the randomizations. Blue crosses mark the features that are constrained by the randomization method.

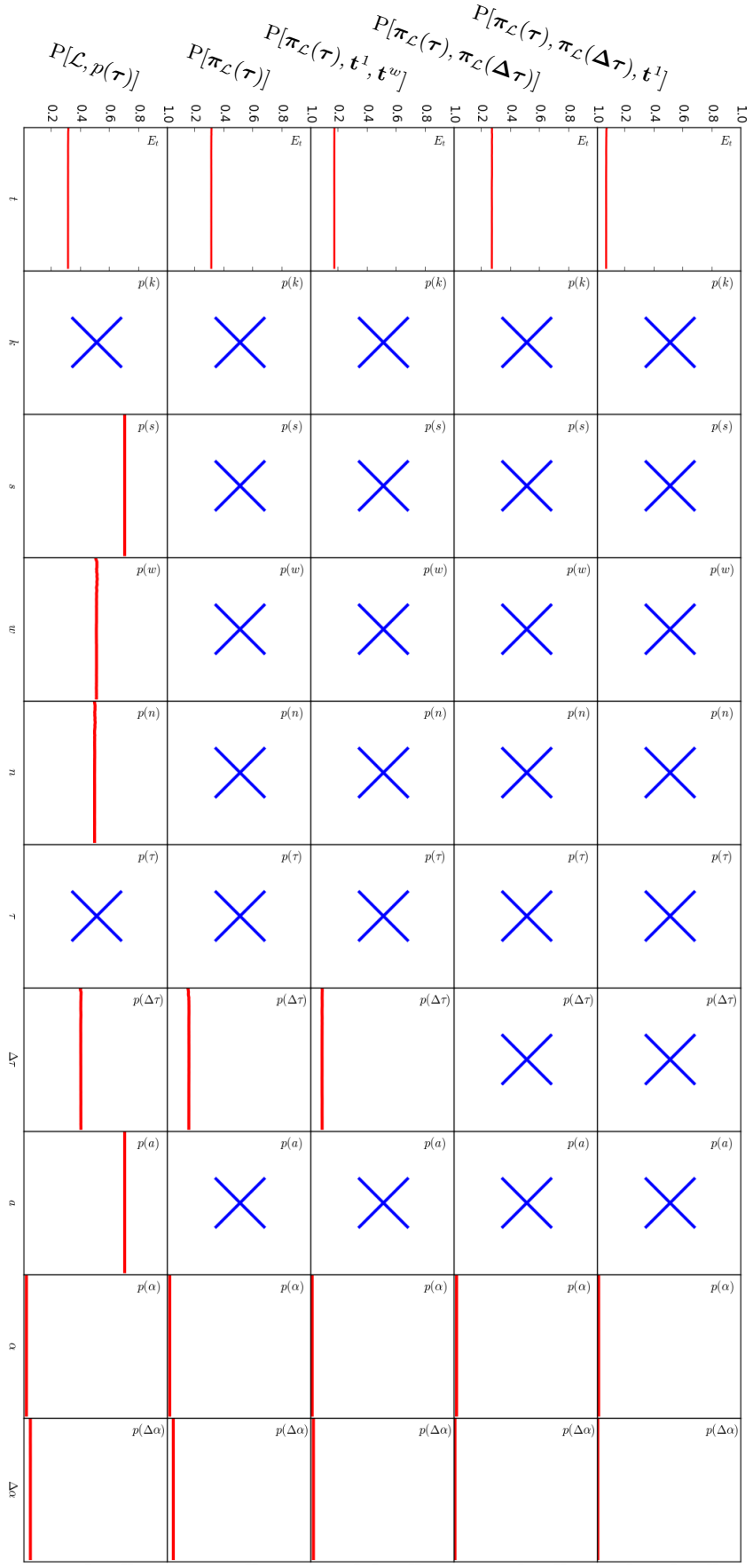


FIG. 16: **Convergence of the distributions generated by link and event shufflings and shown in Fig. 14.** We realize 100 randomizations of the network, and compute the evolution of the distance (as the Jensen-Shannon divergence) between the measure on the data and the median measure on the randomizations. Blue crosses mark the features that are constrained by the randomization method.

[1] Holme, P. & Saramäki, J. Temporal networks. *Physics Reports* **519**, 97–125 (2012). 1108.1780.

[2] Backlund, V.-P., Saramäki, J. & Pan, R. K. Effects of temporal correlations on cascades: Threshold models on temporal networks. *Physical Review E* **89**, 062815 (2014).

[3] Karsai, M. *et al.* Small but slow world: How network topology and burstiness slow down spreading. *Physical Review E* **83**, 025102(R) (2011).

[4] Bajardi, P., Barrat, A., Natale, F., Savini, L. & Colizza, V. Dynamical patterns of cattle trade movements. *PLoS ONE* **6**, e19869 (2011).

[5] Kovanen, L., Karsai, M., Kaski, K., Kertész, J. & Saramäki, J. Temporal motifs in time-dependent networks. *Journal of Statistical Mechanics: Theory and Experiment* **2011**, P11005 (2011).

[6] Squartini, T., Fagiolo, G. & Garlaschelli, D. Randomizing world trade. i. a binary network analysis. *Physical Review E* **84**, 046117 (2011).

[7] Squartini, T., Fagiolo, G. & Garlaschelli, D. Randomizing world trade. ii. a weighted network analysis. *Physical Review E* **84**, 046118 (2011).

[8] Jurgens, D. & Lu, T.-c. Temporal Motifs Reveal the Dynamics of Editor Interactions in Wikipedia. In *Proceedings of the Sixth International AAAI Conference on Weblogs and Social Media*, 1, 162–169 (2012).

[9] Gauvin, L., Panisson, A., Cattuto, C. & Barrat, A. Activity clocks: spreading dynamics on temporal networks of human contact. *Scientific Reports* **3**, 3099 (2013). 1306.4626.

[10] Karimi, F. & Holme, P. Threshold model of cascades in empirical temporal networks. *Physica A: Statistical Mechanics and its Applications* **392**, 3476–3483 (2013).

[11] Kovanen, L., Kaski, K., Kertész, J. & Saramäki, J. Temporal motifs reveal homophily, gender-specific patterns, and group talk in call sequences. *Proceedings of the National Academy of Sciences of the United States of America* **110**, 18070–5 (2013).

[12] Cardillo, A. *et al.* Evolutionary dynamics of time-resolved social interactions. *Physical Review E* **90**, 052825 (2014).

[13] Valdano, E., Poletto, C. & Colizza, V. Infection propagator approach to compute epidemic thresholds on temporal networks: impact of immunity and of limited temporal resolution. *The European Physical Journal B* **88**, 341 (2015).

[14] Holme, P. Modern temporal network theory: a colloquium. *The European Physical Journal B* **88**, 1–30 (2015).

[15] Saracco, F., Clemente, R. D., Gabrielli, A. & Squartini, T. Detecting early signs of the 2007-2008 crisis in the world trade. *Scientific Reports* **30286** (2016).

[16] Thomas, B., Jurdak, R., Zhao, K. & Atkinson, I. Diffusion in colocation contact networks: the impact of nodal spatiotemporal dynamics. *PLoS ONE* **11**, e0152624 (2015).

[17] Génois, M., Vestergaard, C. L., Cattuto, C. & Barrat, A. Compensating for population sampling in simulations of epidemic spread on temporal contact networks. *Nature Communications* **6**, 8860 (2015).

[18] Pósfai, M. & Hövel, P. Structural controllability of temporal networks. *New Journal of Physics* **16**, 123055 (2014).

[19] Takaguchi, T., Masuda, N. & Holme, P. Bursty Communication Patterns Facilitate Spreading in a Threshold-Based Epidemic Dynamics. *PLoS ONE* **8** (2013). 1206.2097.

[20] Takaguchi, T., Sato, N., Yano, K. & Masuda, N. Importance of individual events in temporal networks. *New Journal of Physics* **14**, 093003 (2012).

[21] Holme, P. Network reachability of real-world contact sequences. *Physical Review E* **71**, 046119 (2005).

[22] Takaguchi, T., Sato, N., Yano, K. & Masuda, N. Inferring directed static networks of influence from undirected temporal networks. In *Proceedings of IEEE 37th Annual Computer Software and Applications Conference*, 155–156 (Ieee, Kyoto, 2013).

[23] Takaguchi, T., Yano, Y. & Yoshida, Y. Coverage centralities for temporal networks. *The European Physical Journal B* **89**, 35 (2016).

[24] Karsai, M., Kaski, K., Barabási, A.-L. & Kertész, J. Universal features of correlated bursty behaviour. *Scientific Reports* **2**, 397 (2012).

[25] Karsai, M., Kaski, K. & Kertész, J. Correlated dynamics in egocentric communication networks. *PLoS one* **7**, e40612 (2012).

[26] Orsini, C. *et al.* Quantifying randomness in real networks. *Nature Communications* **6**, 8627 (2015).

[27] Rocha, L. E. C., Liljeros, F. & Holme, P. Simulated Epidemics in an Empirical Spatiotemporal Network of 50,185 Sexual Contacts. *PLoS Computational Biology* **7**, e1001109 (2011).

[28] Miritello, G., Moro, E. & Lara, R. Dynamical strength of social ties in information spreading. *Physical Review E* **83**, 045102(R) (2011).

[29] Starnini, M., Baronchelli, A., Barrat, A. & Pastor-Satorras, R. Random walks on temporal networks. *Physical Review E* **85**, 056115 (2012).

[30] Kivelä, M. *et al.* Multiscale analysis of spreading in a large communication network. *Journal of Statistical Mechanics: Theory and Experiment* **2012**, P03005 (2012). 1112.4312.

[31] Holme, P. & Liljeros, F. Birth and death of links control disease spreading in empirical contact networks. *Scientific Reports* **4**, 4999 (2014).

[32] Karsai, M., Perra, N. & Vespignani, A. Time varying networks and the weakness of strong ties. *Scientific Reports* **4**, 4001 (2014).

[33] Delvenne, J.-C., Lambiotte, R. & Rocha, L. E. C. Diffusion on networked systems is a question of time or structure. *Nature communications* **6**, 7366 (2015).

[34] Saramäki, J. & Holme, P. Exploring temporal networks with greedy walks. *The European Physical Journal B* **88**, 1–8 (2015).

[35] Holme, P. Temporal network structures controlling disease spreading. *Phys. Rev. E* **94**, 022305 (2016).

[36] Li, A., Cornelius, S. P., Liu, Y.-Y., Wang, L. & Barabási, A.-L. The fundamental advantages of temporal networks. *Science* **358**, 1042–1046 (2017).

[37] Tang, J., Scellato, S., Musolesi, M., Mascolo, C. & Latora, V. Small-world behavior in time-varying graphs. *Physical Review E* **81**, 055101 (2010).

[38] Alessandretti, L., Sapiezynski, P., Lehmann, S. &

Baronchelli, A. Evidence for a conserved quantity in human mobility. *arXiv:1609.03526v1* (2016).

[39] Redmond, U. & Cunningham, P. Identifying over-represented temporal processes in complex networks. In *Proceedings of the 2nd Workshop on Dynamic Networks and Knowledge Discovery co-located with ECML PKDD*, vol. 1229, 61–72 (2014).

[40] Sun, K., Baronchelli, A. & Perra, N. Contrasting effects of strong ties on SIR and SIS processes in temporal networks. *European Physical Journal B* **88**, 326 (2015).

[41] Jaynes, E. T. Information theory and statistical mechanics. *Physical Review* **106**, 620–630 (1957).

[42] Presse, S., Ghosh, K., Lee, J. & Dill, K. A. Principles of maximum entropy and maximum caliber in statistical physics. *Reviews of Modern Physics* **85**, 1115–1141 (2013).

[43] Good, P. *Permutation, Parametric and Bootstrap Tests of Hypotheses (3rd ed.),* (Springer, 2005).

[44] Katz, L. & Powell, J. H. Probability distributions of random variables associated with a structure of the sample space of sociometric investigations. *Annals of Mathematical Statistics* **28**, 442 (1957).

[45] Snijders, T. Enumeration and simulation methods for 0–1 matrices with given marginals. *Psychometrika* **56**, 397–417 (1991).

[46] Fournet, J. & Barrat, A. Contact patterns among high school students. *PLoS ONE* **9**, e107878 (2014).

[47] Stopczynski, A. *et al.* Measuring large-scale social networks with high resolution. *PLoS ONE* **9**, e95978 (2014).

[48] A link-timeline network is equivalent to the *interval graph* defined in [1]; it is furthermore equivalent to the link stream defined in [92], but where the set of possible event times is defined explicitly there. If we set $\tau_m = 0$, i.e. in the instantaneous event limit, we obtain what is termed *contact sequences* in [1].

[49] Note that from a practical point of view it is enough for our purposes to consider only finite state spaces. Considering all possible temporal networks would make the state space at least countably infinite, but as all of the reference models encountered in the literature keep the system size (measured in the number of nodes) fixed, we can (implicitly) fix each full state space to contain only networks of fixed size. Further, for RRMs the time can be considered finite as the observation windows and measurement resolutions are finite.

[50] Erdos, P. & Rényi, A. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences* **5**, 17–61 (1960).

[51] Newman, M. E. The structure and function of complex networks. *SIAM Review* **45**, 167–256 (2003).

[52] Fosdick, B. K., Larremore, D. B., Nishimura, J. & Ugander, J. Configuring random graph models with fixed degree sequences. *SIAM Review* **60**, 315–355 (2018).

[53] Maslov, S. & Sneppen, K. Specificity and Stability in Topology of Protein Networks. *Science* **296**, 910–913 (2002).

[54] Hrbacek, K. & Jech, T. *Introduction to Set Theory, Revised and Expanded* (Crc Press, 1999).

[55] Stanley, R. P. Enumerative combinatorics (volume 1 second edition). *Cambridge studies in advanced mathematics* (2011).

[56] Berend, D. & Tassa, T. Improved bounds on bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics* **30**, 185–205 (2010).

[57] Perotti, J. I., Jo, H.-H., Holme, P. & Saramäki, J. Temporal network sparsity and the slowing down of spreading. *arXiv:1411.5553* (2014).

[58] These node-grouped MRRMs can be seen as microcanonical variants of the stochastic block model [93]. However, typical stochastic block models found in the literature assign either the links at random inside each block (i.e. equivalent to $\mathrm{P}[\boldsymbol{\sigma}, \Sigma_{\mathcal{L}}]$) or while constraining the degree sequence (equivalent to $\mathrm{P}[\mathbf{k},\boldsymbol{\sigma}, \Sigma_{\mathcal{L}}]$) [94], while the metadata MRRMs we consider here may be any refinement of these.

[59] Zhang, X., Moore, C. & Newman, M. E. Random graph models for dynamic networks. *The European Physical Journal B* **90**, 200 (2017).

[60] Squartini, T., Mastrandrea, R. & Garlaschelli, D. Unbiased sampling of network ensembles. *New Journal of Physics* **17**, 023052 (2015).

[61] Peixoto, T. P. Hierarchical block structures and high-resolution model selection in large networks. *Physical Review X* **4**, 011047 (2014).

[62] Peixoto, T. P. Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Physical Review E* **92**, 042807 (2015).

[63] Squartini, T. & Garlaschelli, D. Analytical maximum-likelihood method to detect patterns in real networks. *New Journal of Physics* **13**, 083001 (2011).

[64] Perra, N., Gonçalves, B., Pastor-Satorras, R. & Vespignani, A. Activity driven modeling of time varying networks. *Scientific Reports* **2**, 469 (2012).

[65] Pfitzner, R., Scholtes, I., Garas, A., Tessone, C. J. & Schweitzer, F. Betweenness preference: Quantifying correlations in the topological dynamics of temporal networks. *Physical Review Letters* **110**, 198701 (2013). 1208.0588.

[66] Rosvall, M., Esquivel, A. V., Lancichinetti, A., West, J. D. & Lambiotte, R. Memory in network flows and its effects on spreading dynamics and community detection. *Nature Communications* **5**, 4630 (2014).

[67] Scholtes, I. *et al.* Causality-driven slow-down and speed-up of diffusion in non-Markovian temporal networks. *Nature Communications* **5**, 5024 (2014).

[68] Peixoto, T. P. & Rosvall, M. Modeling sequences and temporal networks with dynamic community structures. *Nature Communications* **8**, 582 (2017).

[69] Peixoto, T. P. & Gauvin, L. Change points, memory and epidemic spreading in temporal networks. *arXiv preprint arXiv:1712.08948* (2017).

[70] Casiraghi, G., Nanumyan, V., Scholtes, I. & Schweitzer, F. Generalized hypergeometric ensembles: Statistical hypothesis testing in complex networks. *arXiv:1607.02441* (2016).

[71] Miritello, G., Lara, R., Cebrian, M. & Moro, E. Limited communication capacity unveils strategies for human interaction. *Scientific Reports* **3**, 1950 (2013).

[72] Stehlé, J. *et al.* High-resolution measurements of face-to-face contact patterns in a primary school. *PloS ONE* **6**, e23176 (2011).

[73] Gemmetto, V., Barrat, A. & Cattuto, C. Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC infectious diseases* **14**, 695 (2014).

[74] Lin, J. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information theory* **37**, 145–151 (1991).

[75] Note that since $L \geq p(\boldsymbol{\Delta\tau}) \geq p_{\mathcal{L}}(\boldsymbol{\Theta})$ and $p_{\mathcal{L}}(\boldsymbol{\Theta})$ and $\mathcal{L}$

are conditionally independent on the common coarsening L, then also $p(\boldsymbol{\Delta\tau})$ and $\mathcal{L}$ are and the constraint of $\mathcal{L}$ in $\mathrm{P}[\mathcal{L}, p(\boldsymbol{\tau})]$ does not influence the values that $p(\boldsymbol{\Delta\tau})$ may take.

[76] Pastor-Satorras, R., Castellano, C., Mieghem, P. V. & Vespignani, A. Epidemic processes in complex networks. *Reviews of Modern Physics* **87**, 925 (2015).

[77] Vestergaard, C. L. & Génois, M. Temporal gillespie algorithm: Fast simulation of contagion processes on time-varying networks. *PLoS computational biology* **11**, e1004579 (2015).

[78] Alon, U. Network motifs: theory and experimental approaches. *Nature Reviews Genetics* **8**, 450–461 (2007).

[79] Shen-Orr, S. S., Milo, R., Mangan, S. & Alon, U. Network motifs in the transcriptional regulation network of escherichia coli. *Nature genetics* **31**, 64 (2002).

[80] Since the available datasets were not long enough to simulate full penetration of the dynamical process, periodic temporal boundary conditions were applied letting the process to continue from the beginning of the event sequence once it reached the last event. Note that this condition introduces some biases, by generating non-existing time-respecting paths [83], but with a minor effect which does not qualitatively change the results.

[81] Granovetter, M. S. The strength of weak ties. *American journal of sociology* **78**, 1360–1380 (1973).

[82] Watts, D. J. A simple model of global cascades on random networks. *Proceedings of the National Academy of Sciences* **99**, 5766–5771 (2002).

[83] Pan, R. K. & Saramäki, J. Path lengths, correlations, and centrality in temporal networks. *Physical Review E* **81**, 016105 (2011).

[84] Goh, K. I. & Barabási, A.-L. Burstiness and memory in complex systems. *Europhysics Letters* **81**, 48002 (2008).

[85] Nowak, M. A. *Evolutionary Dynamics: Exploring the Equations of Life* (Harvard University Press, 2006).

[86] Kikas, R., Dumas, M. & Karsai, M. Bursty egocentric network evolution in Skype. *Social Network Analysis and Mining* **3**, 1393–1401 (2013).

[87] Karsai, M., IÃśiguez, G., Kikas, R., Kaski, K. & KertÃľsz, J. Local cascades induced global contagion: How heterogeneous thresholds, exogenous effects, and unconcerned behaviour govern online adoption spreading. *Scientific reports* **6** (2016).

[88] Kirk, D. E. *Optimal Control Theory: An Introduction* (Prentice-Hall, Inc, 1971).

[89] Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Controllability of complex networks. *Nature* **473**, 167–173 (2011).

[90] Kivelä, M. *et al.* Multilayer networks. *Journal of complex networks* **2**, 203–271 (2014).

[91] Jerrum, M. R., Valiant, L. G. & Vazirani, V. V. Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science* **43**, 169–188 (1986).

[92] Latapy, M., Viard, T. & Magnien, C. Stream graphs and link streams for the modeling of interactions over time. *arXiv preprint arXiv:1710.04073* (2017).

[93] Holland, P. W., Laskey, K. B. & Leinhardt, S. Stochastic blockmodels: First steps. *Social networks* **5**, 109–137 (1983).

[94] Karrer, B. & Newman, M. E. Stochastic blockmodels and community structure in networks. *Physical review E* **83**, 016107 (2011).