



**HAL**  
open science

## Randomized reference models for temporal networks

Laetitia Gauvin, Mathieu Génois, Márton Karsai, Mikko Kivelä, Taro Takaguchi, Eugenio Valdano, Christian L. Vestergaard

► **To cite this version:**

Laetitia Gauvin, Mathieu Génois, Márton Karsai, Mikko Kivelä, Taro Takaguchi, et al.. Randomized reference models for temporal networks. 2019. hal-01817633v2

**HAL Id: hal-01817633**

**<https://hal.science/hal-01817633v2>**

Preprint submitted on 12 Mar 2019 (v2), last revised 24 Nov 2021 (v4)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Randomized reference models for temporal networks

L. Gauvin,<sup>1</sup> M. Génois,<sup>2</sup> M. Karsai,<sup>3</sup> M. Kivela,<sup>4</sup> T. Takaguchi,<sup>5</sup> E. Valdano,<sup>6</sup> and C. L. Vestergaard<sup>2,7,8,\*</sup>

<sup>1</sup>*Data Science Lab, ISI Foundation, Turin, Italy.*

<sup>2</sup>*Aix Marseille Univ., Univ. Toulon, CNRS, CPT, Marseille, France.*

<sup>3</sup>*Université de Lyon, ENS de Lyon, INRIA, CNRS,*

*LIP-UMR 5668, IXXI, 69364, Cedex 07 Lyon, France.*

<sup>4</sup>*Department of Computer Science, Aalto University School of Science, P.O. Box 12200, FI-00076 Aalto, Finland.*

<sup>5</sup>*National Institute of Information and Communications Technology,*

*4-2-1 Nukui-Kitamachi, Koganei, Tokyo 184-8795, Japan.*

<sup>6</sup>*Center for Biomedical Modeling, The Semel Institute for Neuroscience and Human Behavior,*

*David Geffen School of Medicine, 760 Westwood Plaza,*

*University of California Los Angeles, Los Angeles, CA 90024 USA.*

<sup>7</sup>*Decision and Bayesian Computation, Pasteur Institute,*

*CNRS UMR 3571, 25-28 rue du Dr Roux, 75015 Paris, France.*

<sup>8</sup>*Bioinformatics and Biostatistics Hub, C3BI, Pasteur Institute,*

*CNRS USR 3756, 25-28 rue du Dr Roux, 75015 Paris, France.*

(Dated: February 7, 2019)

Many dynamical systems can be successfully analyzed by representing them as networks. Empirically measured networks and dynamic processes that take place in these situations show heterogeneous, non-Markovian, and intrinsically correlated topologies and dynamics. This makes their analysis particularly challenging. Randomized reference models (RRMs) have emerged as a general and 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 for example be used to identify important features of empirical networks and their effects on dynamical processes unfolding in the network. RRMs are typically implemented as procedures that reshuffle an empirical network, making them very generally applicable. However, the effects of most shuffling procedures on network features remain poorly understood, rendering their use non-trivial and susceptible to misinterpretation. Here we propose a unified framework for classifying and understanding micro-canonical RRMs (MRRMs). Focusing on temporal networks, we use this framework to build a taxonomy of MRRMs that proposes a canonical naming convention, classifies them, and deduces their effects on a range of important network features. We furthermore show that certain classes of *compatible* MRRMs may be applied in sequential composition to generate over a hundred new MRRMs from the existing ones surveyed in this article. We provide two tutorials showing applications of the MRRM framework to empirical temporal networks: 1) to analyze how different features of a network affect other features and 2) to analyze how such features affect a dynamic process in the network. We finally survey applications of MRRMs found in literature. Our taxonomy provides a reference for the use of MRRMs, and the theoretical foundations laid here may further serve as a base for the development of a principled and automatized way to generate and apply randomized reference models for the study of networked systems.

---

\* cvestergaard@gmail.com

## CONTENTS

I. Introduction	3
A. Road map to this article	4
II. Fundamental definitions and general results	4
A. Temporal network	5
B. Randomized reference model	7
C. Combining microcanonical randomized reference models	10
III. Features of temporal networks	14
A. Notation for features of instant-event networks	21
IV. Shuffling methods	21
A. Link and timeline shufflings	22
B. Sequence and snapshot shufflings	23
V. Classifying randomized reference models	23
A. Naming convention	24
B. Applying instant-event shufflings to temporal networks with event durations	24
C. The basic instant-event and event shufflings	26
D. Link shufflings	27
E. Timeline shufflings	27
F. Sequence shufflings	28
G. Snapshot shufflings	28
H. Intersections of two shufflings	29
I. Compositions of two shufflings	29
J. Randomization based on metadata	30
K. Other reference models	30
VI. Analysis and hypothesis testing using randomized reference models	32
A. Walkthrough example: analyzing face-to-face interactions using MRRMs	33
B. Walkthrough example: analyzing temporal distances in a communication network using MRRMs	35
VII. Applications of randomized reference models	36
A. Contagion processes	37
B. Random walks	41
C. Evolutionary games	42
D. Temporal motifs and networks with attributes	42
E. Network controllability	44
VIII. Conclusion	45
A. Randomized reference models for directed temporal networks	46
References	46

## I. INTRODUCTION

Random network models are responsible of major parts of our theoretical understanding of networked systems and practical knowledge extracted from networked data. Well-known examples of such models include the Erdős-Rényi model [1] – fixing the total number of links – and the configuration model [2, 3] – fixing the degree sequence. A main application of these models is as null models for hypothesis testing, but their use go beyond this. To underline their general scope, we here call these type of models *randomized reference models* (RRMs). Much of what we know of the behavior of dynamic processes in networks is based on these [4, 5], and they stand behind many prominent results on network science such as the absence of epidemic threshold [6], the vulnerability to attacks [7–10], and the robustness to failures [10–12] in certain types of networks. These models are also integral parts of many methods for network data analysis, such as popular network clustering methods [13, 14], network motif analysis [15, 16], and the analysis of structural correlations [17, 18]. The above is just a small selection of applications, but the examples are legion.

As network science has matured there has been an increasing need to go beyond the simple graph representation for networks, and at the same time repeat the success of RRMs for these new types of networks. An important extension to simple graphs are temporal networks, which allow the networks’ topology to evolve in time. RRMs [19] have also emerged as a powerful toolbox for the study of the dynamics on and of temporal networks [20, 21] and have been applied to complex systems in a broad range of fields, including sociology, epidemiology, infrastructure, economics, and biology. They have been used to study how given network features affect other node- or interaction-level features [22–26], and how the features affect dynamical processes unfolding in the network [27–44] as well as the network’s controllability [45, 46]. Systems studied using temporal network RRMs include: human face-to-face interactions and physical proximity [30, 32–34, 37–40, 43, 44, 47–50]; prostitution networks [28, 33, 40, 44]; brain function [50]; human mobility [51]; livestock transport [52]; mobile phone calls and text messages [25, 27, 29, 31, 37]; email correspondences [27, 33, 34, 37, 40, 45, 49, 53]; online communities [33, 40, 50, 53–55]; editing of Wikipedia pages [56]; and world trade [57–59].

The popularity of RRMs for the study of complex networks may be explained by the fact that they may often be defined simply as numerical procedures that generate random networks by shuffling the original data, thus avoiding the need to specify a complete generative model. The resulting ensemble of randomized networks typically serves as a null reference, which is compared to the original temporal network, or may be compared to a second ensemble generated by another RRM. For example, by comparing how a given dynamic process evolves on the randomized networks with how it evolves on the original

network, we may identify how the randomized features affect the dynamic process.

The algorithmic definition of RRMs as shuffling methods makes them simple to apply in very general settings and with little domain-specific tweaking needed. However, an important downside to the algorithmic representation is that the effects of RRMs on 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. This makes it difficult to compare the methods used in different studies and has lead to a situation where the algorithms producing equivalent RRMs are given a multitude of different names, and possibly worse, where multiple algorithms producing different RRMs are given the same names. Second, researchers are confronted with the problems of how to choose and develop randomization techniques, in which order they should be applied, and how to interpret the results. These are crucial choices in order to be able to identify important features for each given dynamical phenomenon and for each temporal network under study (problems that are non-trivial even for simple graphs [26]).

We review temporal network RRMs used in the literature and find that most of them fall into a class of methods that gives a uniform probability of sampling all networks with a given set of features constrained to the same value as that of the original data [31]. Inspired by the concept of microcanonical ensembles in statistical physics [60, 61], 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. Furthermore, this framework enables us to build a taxonomy of existing MRRMs, which lists their effects on important temporal network features and 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 previously implemented algorithms one after another.

Reference models, which keep parts of the features of original data and shuffle the rest, are clearly widely applicable outside of temporal networks. For example, MRRMs are closely related to exact (permutation) tests of classical statistics [62] and to conditionally uniform graph tests (CUGTs) found in the sociology literature [53, 63, 64]. Furthermore, even though we are here mostly concerned with temporal networks, our framework of MRRMs is directly applicable to a far more general class of systems, which can be considered as a re-

alization of a state in a predefined microcanonical state space. In particular, it may be applied e.g. to correlation matrices or to more general types of relational data such as multilayer networks or hypergraphs.

It is our aim that, in addition to categorizing previous RRM 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; furthermore, 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.

### A. Road map 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 V 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 VI, while a literature review of applications of MRRMs to temporal network data is given in Section VII.

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

In Section III we list a number of important features of temporal networks, which are sufficient to fully describe the MRRMs found in our literature review, and we establish their hierarchy.

In Section IV 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 II. Furthermore, we see how the compositions of MRRMs discussed earlier in theory can be used in practice.

In Section V 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 Section VI we give a procedure for statistical analysis using MRRMs, and we provide two different walk-through examples of how to apply them to analyze empirical temporal networks. Namely, we give 1) an analysis of how different features of a face-to-face interaction network constrain and affect other features of the network, and 2) a characterization of how different network features influence the speed of spreading in an email communication network.

In Section VII we finally give a review of applications of MRRMs found in the literature.

An appendix lists additional features for describing MRRMs for directed temporal networks (Appendix A).

## II. FUNDAMENTAL DEFINITIONS AND GENERAL RESULTS

In this section we define fundamental concepts and derive general results for temporal networks and for microcanonical randomized reference models (MRRMs). We focus on microcanonical RRM as these represent the class of maximum entropy models that can be obtained directly by randomly sampling constrained permutations of an empirically observed network (i.e. by shuffling the network). We note in passing that, while we here apply the formalism to the study of temporal networks, the results derived in subsections IIB and IIC apply more generally to MRRMs for any system with a discrete state space [65].

Subsection IIA 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 III) and the implementation of many MRRMs (Section IV). Subsection IIB provides a rigorous definition of a MRRM and introduces its 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 V). These hierarchies are also useful in the practical employment of MRRMs (Section VI).

Finally, Subsection IIC 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 the theory needed to show that two MRRMs are compatible and to identify the MRRMs resulting from their composition. The theorems developed in this subsection are instrumental for defining the important classes of compatible MRRMs which are employed in practice in Section IV.

## 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 II.1.** *Temporal network.* A temporal network  $G = (\mathcal{V}, \mathcal{C})$  is defined by the set of nodes  $\mathcal{V} = \{1, 2, \dots, N\}$  and a set of events  $\mathcal{C} = \{c_1, c_2, \dots, c_C\}$ , 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 II.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), but does not consider possible weights of the events. 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  $(i_q, j_q, t_q, \tau_q)$  implies the symmetric interaction  $(j_q, i_q, t_q, \tau_q)$ , and in practice we may impose  $i_q < j_q$  for efficient data storage.

For simplicity, we consider only undirected temporal networks in the main text, and in the same spirit of simplicity we have not considered possible weights of events in the above definition. However, all models and methods may be applied directly to temporal networks with directed and/or weighted events. Furthermore, they may easily be extended to explicitly take into account the directionality or weights of interactions by defining the appropriate features (see Appendix A and Section VII).

For some systems, e.g., email communications or instant messaging [27, 33, 34, 37, 40, 45, 49, 53], 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 [27, 31]. 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 II.2.** *Instant-event temporal network.* An instant-event temporal network  $G = (\mathcal{V}, \mathcal{E})$  is defined by the set of nodes  $\mathcal{V} = \{1, 2, \dots, N\}$  and a set of reduced events  $\mathcal{E} = \{e_1, e_2, \dots, e_E\}$ , 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 [66, 67], are recorded with a coarse time resolution at evenly spaced points in time,  $t = t_{\min}, t_{\min} + \Delta t, t_{\min} + 2\Delta t, \dots, t_{\max} - \Delta t$ . 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  $\tau$  indicating the length of the event. Both representations will come in handy for defining different types of MRRMs.

**Definition II.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  $\mathbf{x} : \mathcal{G} \rightarrow \mathcal{X}$ , 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 functions, 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 II.4.** *Static graph.* The static graph,  $G^{\text{stat}}$ , 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 the set of *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 Def. II.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 [20]. 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 III).

### 1. Two-level temporal network representations

Sometimes it is useful to separate the static structure and the temporal aspect in the definition of the temporal network as opposed to having them mixed together like in definitions II.1 and II.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. III), and for designing and implementing MRRMs for temporal networks (Sec. IV).

**Definition II.5.** *Link-timeline network.* A link-timeline network  $G_{\mathcal{L}} = (G^{\text{stat}}, \Theta)$  represents a temporal network by using the static graph  $G^{\text{stat}} = (\mathcal{V}, \mathcal{L})$  to indicate the

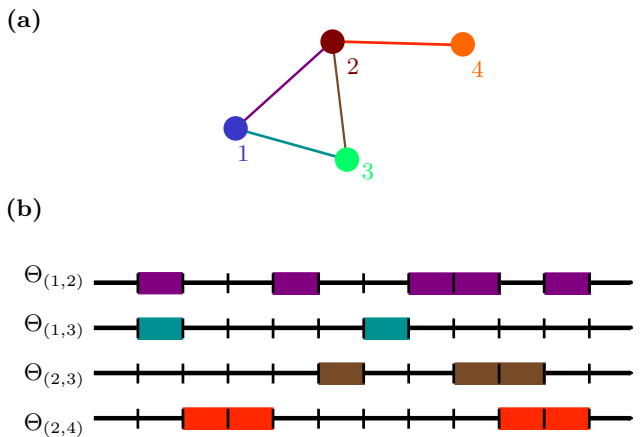


FIG. 1: **Link-timeline network.** Graphical representation of the link-timeline network  $G_{\mathcal{L}}$  of an illustrative temporal network consisting of four nodes and recorded in discrete time. (a) Static graph  $G^{\text{stat}}$  showing links between nodes. Links are drawn between pairs  $(i,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

pairs of nodes that interact at least once during the observation period (Def. II.4). To each link  $(i, j) \in \mathcal{L}$  we associate a *timeline*  $\Theta_{(i,j)} \in \Theta$ , which indicates when the corresponding nodes interact. Each timeline is given by a sequence

$$\Theta_{(i,j)} = \left( (t_{(i,j)}^1, \tau_{(i,j)}^1), (t_{(i,j)}^2, \tau_{(i,j)}^2), \dots, (t_{(i,j)}^{n_{(i,j)}}, \tau_{(i,j)}^{n_{(i,j)}}) \right), \quad (1)$$

where  $t_{(i,j)}^m$  is the start of the  $m$ th event on link  $(i, j)$ , with  $\tau_{(i,j)}^m$  its duration, and  $n_{(i,j)}$  is the total number of events taking place over the link [68].

**Example II.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  $G^{\text{stat}}$ , while panel (b) shows the link timelines  $\Theta$ .

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:

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

$$\Gamma = (\Gamma^1, \Gamma^2, \dots, \Gamma^T), \quad (2)$$

where for each  $m = 1, 2, \dots, T$ ,  $\Gamma^m \in \Gamma$  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

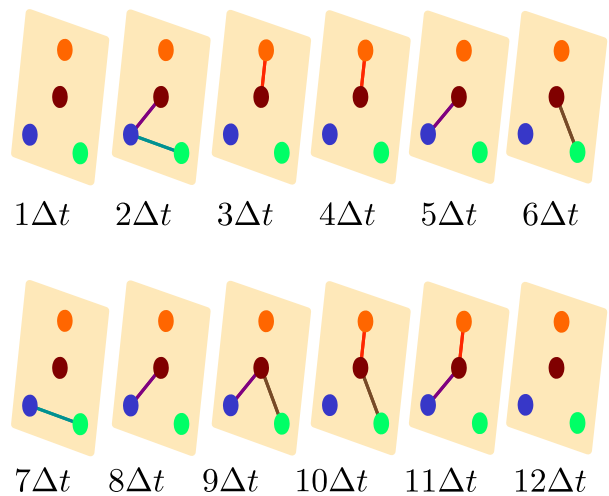


FIG. 2: **Snapshot-graph sequence.** Sequence of snapshot graphs of the temporal network shown in Fig. 1.

set of edges for which there is an event taking place at time  $t$ ,

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

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 sequence of times becomes  $\mathcal{T} = (\Delta t, 2\Delta t, \dots, T)$ ) and if the time resolution is coarse enough so that the individual snapshot graphs do not become too sparse. We use the shorthand  $\Gamma^t$  to refer to the snapshot graph associated with the time  $t \in \mathcal{T}$  (i.e.  $\Gamma^{t_m}$  for  $m$  such that  $t_m = t$ ).

**Example II.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. They are simply alternative ways of representing them because any temporal network can be uniquely 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 types 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. Furthermore, 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 acquaintance 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. Furthermore, 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, for networks in which the structure changes on 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 where 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 IV and see that many RRM found in the literature are implemented this way in Section V.

## 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 IV and V.

### 1. Basic definitions

Consider a predetermined finite [69] Many procedures leading to a sampling from a conditional probability distribution  $P(G|G^*)$  defined on  $\mathcal{G}$  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 RRM include Erdős-Renyi (ER) models [1, 2], configuration models [2, 3], and exponential random graph models [2].

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 static 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 a randomly selected graph with degree sequence  $\mathbf{k} = (k_i)_{i \in \mathcal{V}}$ , also known as the *Maslov-Sneppen* model [15], are MRRMs.

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

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

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

where  $\delta$  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 sometimes 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. II.11 below).

**Example II.3.** The Maslov-Sneppen model [15] is an example of a MRRM for static graphs. In the state space of static graphs it is defined as  $P[\mathbf{k}]$ . It maps an input graph  $G^*$  to a microcanonical ensemble of graphs 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 II.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 can represent a MRRM by a transition matrix  $\mathbf{P}^{\mathbf{x}}$  with elements

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

$\mathbf{P}^{\mathbf{x}}$  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 [70]: 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 which  $G^* \in \mathcal{G}$  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. *Shuffling method.* An algorithm that transforms  $G^*$  into  $G$  according to Def. II.7. These algorithms often shuffle some elements of  $G^*$ . Note that multiple algorithms or shuffling procedures might correspond to the same MRRM and in this case these are considered here to be the same shuffling method.

**Example II.4.** To illustrate the different MRRM representations, we consider the state space  $\mathcal{G}$  of all static graphs with 3 nodes and the MRRM  $P[L]$ , defined by the feature  $L$  which returns the number of edges in the network, corresponding to the Erdős-Rényi random graph model  $ER(3, L)$ . We number the 8 states in  $\mathcal{G}$  such that  $L(G_1) = 0$ ,  $L(G_i) = 1$  for  $i = 2, 3, 4$ ,  $L(G_i) = 2$  for  $i = 5, 6, 7$ , and  $L(G_8) = 3$  [Fig. 3(a)], and we take as input state the graph  $G^* = G_5$ . Figure 3 shows graphically the four different representations of  $P[L]$  for the state space  $\mathcal{G}$  and the input graph  $G^*$ .

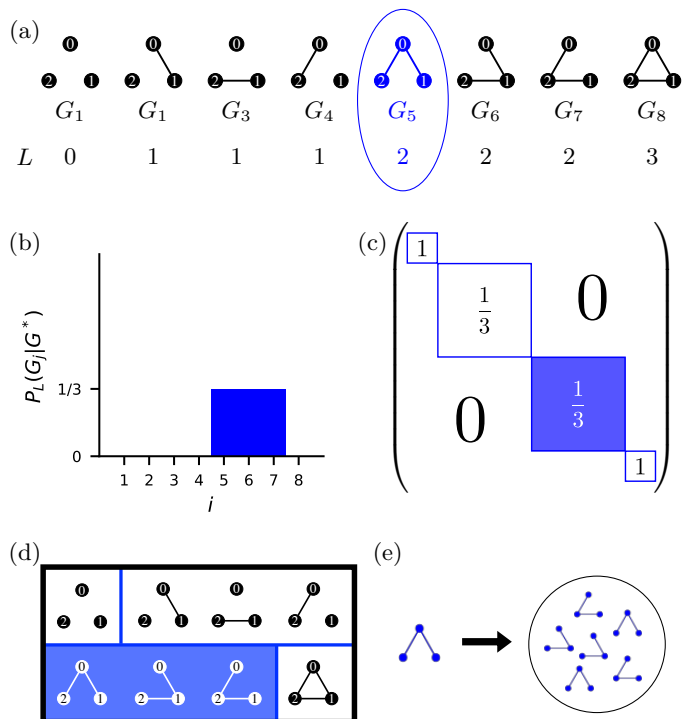
All of these representations are equivalent in a sense that they completely and uniquely specify a MRRM (in the case of shuffling methods this is by definition). For example, each  $P[\mathbf{x}]$  defines exactly one partition and each partition defines exactly one  $P[\mathbf{x}]$ . Note that two feature functions  $\mathbf{x}$  and  $\mathbf{y}$  might correspond to the same partition, transition matrix, or shuffling method, but in this case the two functions also give the same conditional probabilities in Def. II.7 and we say that  $P[\mathbf{x}] = P[\mathbf{y}]$ . The power of the equivalence between the different representations is that any result proven for one representation automatically carries over to the others. We will in the following text switch between the representations to use the one that is most convenient in each context.

## 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. Furthermore, sequences of MRRMs where each one shuffles slightly more structure than the previous one are often used in practice [26, 31]. A central concept is that of *comparability* of MRRMs.

**Definition II.9.** *Comparability.* We will write  $P[\mathbf{x}] \leq P[\mathbf{y}]$  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  $P[\mathbf{x}] \leq P[\mathbf{y}]$  or  $P[\mathbf{y}] \leq P[\mathbf{x}]$ .

The intuition behind this notation is that when  $P[\mathbf{x}] \leq P[\mathbf{y}]$ , then  $P[\mathbf{y}]$  shuffles more of the given structure than



**FIG. 3: Equivalent MRRM representations.** We consider the state space of all simple graphs with three nodes and the MRRM  $P[L]$  which constrains the number of edges  $L$  in a graph. The state space  $\mathcal{G}$  is ordered as shown in (a). The input network is taken to be  $G^* = G_5$  (highlighted in blue). Panels (b)-(e) illustrate graphically the four equivalent representations (Defs. II.7 and II.8) of  $P[L]$  when applied to  $G^*$ : (b) The conditional probability  $P_L(G_j|G^*)$  gives the probability to generate each state  $G_j$  using  $P[L]$  with  $G^*$  as the input state. (c) The block-diagonal stochastic matrix  $\mathbf{P}^L$  gives the probability to generate any output state  $G_j$  from any given input state  $G_i$  (the block corresponding to the input state  $G^*$  is marked in blue). (d) The partition of  $\mathcal{G}$  induced by  $P[L]$  consists of the four distinct sets:  $\mathcal{G}_0 = \{G_1\}$ ,  $\mathcal{G}_1 = \{G_2, G_3, G_4\}$ ,  $\mathcal{G}_2 = \{G_5, G_6, G_7\}$  (marked in blue), and  $\mathcal{G}_3 = \{G_8\}$ . (e) A shuffling method corresponding to  $P[L]$  samples graphs  $G_j$  from  $\mathcal{G}$  according to  $P_L(G_j|G^*)$ . (Note that in a real application the number of states in the ensemble generated by a MRRM is typically very large, so almost all states are sampled at most once in practice.)

$P[\mathbf{x}]$ . Due to the 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 II.5.** In the space of all static graphs with  $N$  nodes, one can define the MRRMs corresponding to the Erdős-Rényi random graph model [1],  $P[L]$ , and the Maslov-Sneppen model [15],  $P[\mathbf{k}]$ . We have  $L = \sum_i k_i/2$ , so  $L$  is a function of  $\mathbf{k}$  and  $P[L] \geq P[\mathbf{k}]$ . Conversely,  $\mathbf{k}$

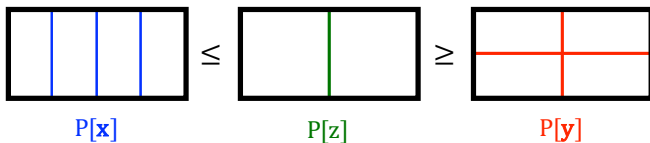


FIG. 4: **Comparability.**  $P[z]$  is coarser than both  $P[x]$  and  $P[y]$  since each set in the partition induced by  $P[x]$  ( $P[y]$ ) is contained in a set of the partition induced by  $P[z]$ . Conversely,  $P[x]$  and  $P[y]$  are not comparable.

is not a function of  $L$ , so  $P[L] \not\leq P[k]$ , as networks with different degree sequences can have the same number of links.

Representing MRRMs as partitions (see Def. II.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 II.1.** *Equivalence with partition refinements.*  $P[x] \leq P[y]$  if and only if the partition  $\mathcal{G}_x$  is finer than  $\mathcal{G}_y$ .

*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  $\mathcal{G}_{x^*}$  (Def. II.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  $\mathcal{G}_{y^*}$  may correspond to different values of  $\mathbf{x}$ . Thus  $\mathcal{G}_{x(G^*)} \subseteq \mathcal{G}_{y(G^*)}$  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  $\mathbf{y}$ .  $\square$

Borrowing the terminology from the theory of set partitions, we say for  $P[x] \leq P[y]$  that  $P[x]$  is *finer* than  $P[y]$  and equivalently that  $P[y]$  is *coarser* than  $P[x]$ . We will also refer to  $P[x]$  as a *refinement* of  $P[y]$  and to  $P[y]$  as a *coarsening* of  $P[x]$ . Figure 4 illustrates the concept of comparability in terms of partitions.

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 appropriate as the relation it denotes is indeed a partial order:

**Corollary II.1.** *Comparability induces a partial order.* The relation  $\geq$  is a partial order over the space of MRRMs.

The above corollary follows immediately from the fact that partition refinement 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 V).

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 II.10.** *Zero and unity elements.* The *zero element*,  $P[0] = P[G]$ , 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*,  $P[1]$ , 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:  $P[0] \leq P[x] \leq P[1]$  for any  $x$ .

**Example II.6.** We continue from Example II.5, limiting the state space to the set of simple graphs consisting of 3 nodes,  $\mathcal{V} = \{1, 2, 3\}$ , and 2 links. There are three such graphs:  $\{G_1, G_2, G_3\}$ , with  $\mathcal{L}(G_1) = ((1, 2); (1, 3))$ ,  $\mathcal{L}(G_2) = ((1, 2); (2, 3))$ ,  $\mathcal{L}(G_3) = ((1, 3); (2, 3))$ . 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  $\mathbf{k}(G_3) = (1, 1, 2)$ , so the partition related to the Maslov-Sneppen model separates all networks  $\mathcal{G}_k = \{\{G_1\}, \{G_2\}, \{G_3\}\}$ . The partition  $\mathcal{G}_k$  is a refinement of  $\mathcal{G}_L$  and thus  $P[k] \leq P[L]$ . Note that for this state space the Maslov-Sneppen model is the zero element  $P[k] = P[0]$  and the ER model is the unity element  $P[L] = P[1]$ .

The interesting hierarchical structure is found between the zero and unity elements, and as we will see, the structure can be 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  $\mathcal{G}$  is the same as the number of possible partitions of the space. This is given by the Bell number  $B_\Omega$  [71], which grows faster than exponentially with the state space size  $\Omega = |\mathcal{G}|$  [72]. 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  $P[x_1] \geq P[x_2] \geq P[x_3] \dots$ ) is the maximum chain length in the set of partitions of the state space, which is equal to  $\Omega + 1$ . 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,  $\Omega$ , typically is extremely large. A point of practical concern is that we are often interested in studying the effects of both MRRMs that randomize temporal features and MRRMs that randomize topological features of temporal

networks. Since such MRRMs are generally not comparable (see Section V), we find ourselves constrained to use and study sets of MRRMs for temporal networks that do not satisfy a linear order [31].

### 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.

We use these results in Section IV to show that MRRMs which shuffle the links are compatible with MRRMs which randomize the timelines in the link-timeline representation of a temporal network (Def. II.5), and that MRRMs shuffling the order of snapshots are compatible with MRRMs randomizing the individual snapshots in the snapshot graph sequence representation (Def. II.6).

#### 1. Intersections and compositions

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

**Definition II.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  $P[\mathbf{x}, \mathbf{y}] = P[(\mathbf{x}, \mathbf{y})]$

Note that the intersection by definition gives another MRRM. In terms of conditional probabilities the intersection becomes

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

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

The partition of  $\mathcal{G}$  induced by  $P[\mathbf{x}, \mathbf{y}]$  is trivially given by the set of pairwise intersections between the

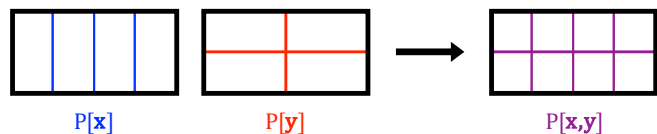


FIG. 5: **Intersection of MRRMs.** The intersection  $P[\mathbf{x}, \mathbf{y}]$  of two MRRMs,  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ , shuffles less than either of the two. In terms of partitions,  $P[\mathbf{x}, \mathbf{y}]$  produces the greatest lower bound of  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ .

$\mathbf{x}$ -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  $P[\mathbf{x}, \mathbf{y}]$  shuffles less (or equally) than  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$  (Fig. 5).

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  $P[\mathbf{x}, 1] = P[\mathbf{x}]$ , 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  $P[\mathbf{x}, 0] = P[0]$ , 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 [70].

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

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

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

For a given indexing of the state space  $\mathcal{G}$ , Eqs. (5) and (7) show that the transition matrix for the composition of  $P[\mathbf{y}]$  on  $P[\mathbf{x}]$  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 II.13.** *Compatibility.* We say that two MRRMs  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$  are *compatible* if their composition  $P[\mathbf{y}]P[\mathbf{x}]$  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 II.2.** *Compatible randomized reference models commute.* If two MRRMs,  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ , are compatible then  $P[\mathbf{x}]P[\mathbf{y}] = P[\mathbf{y}]P[\mathbf{x}]$ .

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

$$\mathbf{P}^y \mathbf{P}^x = (\mathbf{P}^y \mathbf{P}^x)^\top = (\mathbf{P}^x)^\top (\mathbf{P}^y)^\top = \mathbf{P}^x \mathbf{P}^y . \quad (8)$$

□

Proposition II.2 means that it does not matter in which order we apply two compatible MRRMs in the composition, and consequently that  $P[\mathbf{y}]P[\mathbf{x}]$  and  $P[\mathbf{x}]P[\mathbf{y}]$  define the same MRRM if  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$  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 II.7.** Let the state space  $\mathcal{G}$  be all static graphs with 3 nodes. As in Example II.4, 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  $P[L]$ , defined by the number of links  $L$  in the graph, 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  $P[\delta_{G,G_4}]$  keeps the state  $G_4$  and shuffles all the others. It is defined by the feature  $\delta_{G,G_4}$  which returns 1 if the state is  $G_4$  and 0 for all other states. This 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  $P[L]P[\delta_{G,G_4}] \neq P[\delta_{G,G_4}]P[L]$ ; for example, for  $G_4$  the target space of  $P[L]P[\delta_{G,G_4}]$  is  $\{G_2, G_3, G_4\}$ , while the target space of  $P[\delta_{G,G_4}]P[L]$  is the entire  $\mathcal{G}$ . So the two MRRMs do not commute and are thus not compatible. Consequently, the ensembles obtained by composition of  $P[\delta_{G,G_4}]$  and  $P[L]$  are not microcanonical. It is in the above example also easy to verify that the states generated by  $P[\delta_{G,G_4}]P[L]$  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 II.3.** *Comparable microcanonical randomized reference models are compatible.* Let  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$

be two MRRMs and  $P[\mathbf{x}] \leq P[\mathbf{y}]$ . Then they are compatible and their composition gives  $P[\mathbf{y}]P[\mathbf{x}] = P[\mathbf{y}]$ .

*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 is equal to  $\mathbf{y}^* = \mathbf{y}(G^*)$ . This means that  $P_{\mathbf{y}}(G|G') = P_{\mathbf{y}}(G|G^*)$  for all  $G' \in \mathcal{G}_{\mathbf{x}^*}$ , and Eq. (7) reduces to:

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

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

**Example II.8.** Consider again the MRRMs  $P[L]$  and  $P[\mathbf{k}]$  from Example II.5. Since they are comparable, they are compatible according to Proposition II.3. Consequently they commute (Proposition II.2) and  $P[\mathbf{k}]P[L] = P[L]P[\mathbf{k}] = P[L]$ .

We can use Proposition II.2 to show that the composition of two compatible MRRMs randomizes more than either of the individual MRRMs.

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

*Proof.* Since  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$  are compatible,  $P[\mathbf{y}]P[\mathbf{x}]$  is a MRRM by the definition of compatibility (Def. II.13). Since  $G$  itself is by definition in the target set of any MRRM applied to  $G$ , the target set of  $P[\mathbf{y}]P[\mathbf{x}]$  can never be smaller than the target set of  $P[\mathbf{x}]$ . Thus,  $P[\mathbf{y}]P[\mathbf{x}] \geq P[\mathbf{x}]$ . By the same reasoning we obtain that  $P[\mathbf{x}]P[\mathbf{y}] \geq P[\mathbf{y}]$ , and since  $P[\mathbf{x}]P[\mathbf{y}] = P[\mathbf{y}]P[\mathbf{x}]$  (Proposition II.2), that  $P[\mathbf{y}]P[\mathbf{x}] \geq P[\mathbf{y}]$ . □

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 II.3) for which zero is the neutral element  $P[\mathbf{x}]P[0] = P[\mathbf{x}]$  and unity is the absorbing element  $P[1]P[\mathbf{x}] = P[1]$ . Furthermore, by Proposition II.4, the composition gives an upper bound for the two MRRMs (Fig. 6). In fact, the bound is the least upper bound, and any set of compatible MRRMs forms a *lattice* [71], but this connection to the theory of partially ordered sets is not pursued further here.

## 3. Conditional independence and compatibility

Our aim in this section is to be able to compose MRRMs to produce new ones, and even though comparable MRRMs are always compatible they are not useful

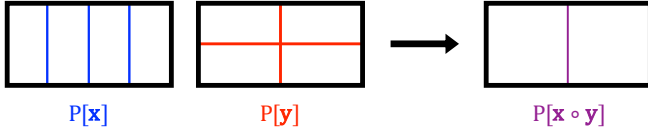


FIG. 6: **Composition of MRRMs.** The composition  $P[\mathbf{x} \circ \mathbf{y}]$  of two compatible MRRMs,  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ , shuffles more than (or as much as) either of the two. In terms of partitions,  $P[\mathbf{x} \circ \mathbf{y}]$  produces the least upper bound of  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ .

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 in Theorem 1 that it is equivalent to compatibility. We next show in Theorem 2 that certain refinements of compatible MRRMs (termed *adapted* refinements) are themselves compatible. Theorem 2 furthermore shows which features their composition inherits from the original MRRMs.

The theorems will be used in Section IV to show that certain important classes of MRRMs are compatible. In particular, we will show that *link shufflings*, which shuffle the links in the link-timeline representation of a temporal network (Def. II.5), are compatible with *timeline shufflings*, which randomize the individual timelines. This will be done by first showing that the coarsest possible link and timeline shufflings are compatible using Theorem 1. Second, noting that all link and timeline shufflings are adapted refinements of these, Theorem 2 then gives us that they are compatible, as well as telling us which features their composition constrains. The same reasoning will be applied to show that *sequence shufflings*, which shuffle the order of snapshots in the snapshot-graph sequence (Def. II.6), and *snapshot shufflings*, which randomize events inside snapshots, are compatible and to derive which features their compositions constrain.

Before we can define the concept of conditional independence given a common coarsening, we first need to define the concepts of conditional probability and conditional independence between features.

**Definition II.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

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

The conditional probability of a feature satisfies all properties of usual conditional probabilities. We may now

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 below). It also allows us to define the conditional independence in the usual sense as when  $P_{\mathbf{y}|\mathbf{x}, \mathbf{z}}(\mathbf{y}^\dagger|\mathbf{x}^\dagger, \mathbf{z}^*) = P_{\mathbf{y}|\mathbf{z}}(\mathbf{y}^\dagger|\mathbf{z}^*)$  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 II.15.** *Conditional independence given a common coarsening.* If there exist a feature  $\mathbf{z}$  that is a common coarsening of both  $\mathbf{x}$  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 in the following proposition.

**Proposition II.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 in the same manner 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.  $P_{\mathbf{x}|\mathbf{z}}(\mathbf{x}^\dagger|\mathbf{z}^*) = P_{\mathbf{x}|\mathbf{y}, \mathbf{z}}(\mathbf{x}^\dagger|\mathbf{y}^\dagger, \mathbf{z}^*)$ . To show that this implies the symmetric relation, we write out the relation using Eq. (10):

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

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

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 characterization of compatibility. The following theorem proves this.

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

*Proof.* To show that conditional independence given a common coarsening is equivalent to compatibility, 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  $\mathbf{y}(G')$ , as well as  $\mathbf{y}''$  for  $\mathbf{y}(G'')$  and  $\mathbf{x}''$  for  $\mathbf{x}(G'')$ .

**Conditional independence given a common coarsening implies compatibility.** We will use the law of total probability and Proposition II.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:

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

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. II.7) and  $P_{\mathbf{y}|\mathbf{x}}(\mathbf{y}^\dagger|\mathbf{x}^*)$  (Def. II.14). Using the law of total probability, we now expand  $P_{\mathbf{y} \circ \mathbf{x}}(G|\mathbf{x}^*)$  to get:

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

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

**Compatibility implies conditional independence given a common coarsening.** Because  $\mathbf{x}$  and  $\mathbf{y}$  are compatible their composition is a 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:

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

In the third 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  $\Omega_{\mathbf{y}''}$ .  $\square$

As discussed above, conditional independence is 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 II A 1. The simple example below illustrates the concepts of conditional independence and compatibility.

**Example II.9.** Consider a state space with 9 states,  $\mathcal{G} = \{G_1, \dots, G_9\}$ , 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,  $P[f_r]$  and  $P[f_c]$ , are compatible and their composition is the unity element  $P[f_r]P[f_c] = P[f_c]P[f_r] = P[1]$ , 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  $G^{\text{stat}}$  returning the static network and  $p_{\mathcal{L}}(\Theta)$  returning the number events on each link and their timings. If we let  $f_r = G^{\text{stat}}$  and  $f_c = p_{\mathcal{L}}(\Theta)$ , 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. To do so we define a special type of *adapted refinements* of compatible MRRMs which we can show are also compatible.

**Definition II.16.** *Adapted refinement.* Consider two compatible MRRMs,  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ . Any refinement of  $P[\mathbf{x}]$  of the form  $P[\mathbf{x}, \mathbf{f}(\mathbf{y})]$ , where  $\mathbf{f}$  is any function of  $\mathbf{y}$ , is said to be *adapted* to  $P[\mathbf{y}]$ . We will refer to  $P[\mathbf{x}, \mathbf{f}(\mathbf{y})]$  as an *adapted refinement* of  $P[\mathbf{x}]$  with respect to  $P[\mathbf{y}]$ .

In the following theorem we will now demonstrate that all adapted refinements of compatible MRRMs are themselves compatible, as well as showing which features the composition of such MRRMs inherits from the individual MRRMs. This theorem will be very useful in practice since if we can show that a given pair of MRRMs are compatible (i.e. using Theorem 1), we get for free that a whole class of MRRMs, consisting of all adapted refinements of the original MRRMs, are also compatible.

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

*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  $P[\mathbf{y}, \mathbf{f}(\mathbf{x})]$  and  $P[\mathbf{x}, \mathbf{g}(\mathbf{y})]$  are compatible with  $P[\mathbf{y}, \mathbf{f}(\mathbf{x})]P[\mathbf{x}, \mathbf{g}(\mathbf{y})] = P[\mathbf{y} \circ \mathbf{x}, \mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})]$ . To develop the proof, we consider the conditional probability of  $\mathbf{x}$  given  $(\mathbf{y}, \mathbf{f}(\mathbf{x}))$ :

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

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,

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

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:

$$P_{\mathbf{f}(\mathbf{x})|\mathbf{y}}(\mathbf{f}(\mathbf{x}^*) | \mathbf{y}^\dagger) = \sum_{\mathbf{x}^\dagger} P_{\mathbf{f}(\mathbf{x})|\mathbf{x}}(\mathbf{f}(\mathbf{x}^*) | \mathbf{x}^\dagger) P_{\mathbf{x}|\mathbf{y}}(\mathbf{x}^\dagger | \mathbf{y}^\dagger)$$

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

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

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

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.  $\square$

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 II.17.** *Independent MRRMs.*  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$  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 II.2.** *Combining refinements of independent MRRMs.* Consider two independent MRRMs  $P[\mathbf{x}]$  and  $P[\mathbf{y}]$ , and two MRRMs,  $P[\mathbf{y}, \mathbf{f}(\mathbf{x})]$  and  $P[\mathbf{x}, \mathbf{g}(\mathbf{y})]$ , that shuffle less. Then  $P[\mathbf{y}, \mathbf{f}(\mathbf{x})]P[\mathbf{x}, \mathbf{g}(\mathbf{y})] = P[\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{y})]$ .

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

While it is always possible to construct partitions of a given state space that are completely independent, it turns out to be difficult to define independent features of a temporal network that are useful in practice. We have found no examples of independent pairs of temporal network MRRMs in the literature (except for the trivial case where one MRRM is the unit element  $P[1]$ ). So, while fully independent MRRMs are interesting due to their attractive theoretical properties, we will in the following sections only encounter conditionally independent ones.

### III. FEATURES OF TEMPORAL NETWORKS

A typical goal when employing MRRMs is to investigate how given predefined features of a 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 presented in Section V. Table I lists basic features of temporal networks, often describing single elements of a network such as nodes or events, and Table II lists different general ways to construct features describing the whole temporal network by using the basic features as building blocks.

A feature of a temporal network is any function that takes a network as an input (Def. II.3). Clearly, there is a very large number of such functions which could be defined [73], 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 II and the concept of order of features given by Def. II.9 and Proposition II.1.

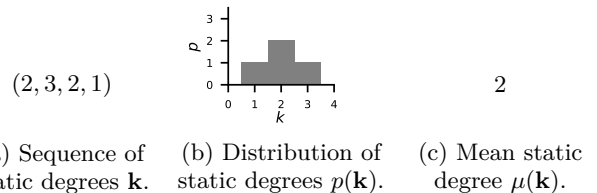
For definiteness we will here consider features of temporal networks with event durations. In order to make our description of MRRMs consistent for both networks with and without event durations, we shall need to modify the definitions of some of these features for instant-event networks. We do this in Subsection III A at the end of this section.

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 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 III.1.** *Sequence of features.* A sequence of features is a tuple  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$  of individual features ordered according to an arbitrary but fixed index  $q \in \mathcal{Q}$ .

The individual features  $\mathbf{x}_q$  may be any functions, e.g. scalar functions, sequences of other features, or graphs. Typically, each  $\mathbf{x}_q$  depends on a different part of the temporal network such as a node  $i$ , a link  $(i, j)$ , or a time  $t$ . We shall in the following use a subscript to index individual topological features (e.g.  $\mathbf{x}_i$  or  $\mathbf{x}_{(i,j)}$ ) for a feature of a single node or link, respectively) and superscript for



**FIG. 7: 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.

temporal ones (e.g.  $\mathbf{x}^t$  for a feature of a single snapshot). Individual features that depend both on topology and time are given both a subscript and a superscript index (e.g.  $x_i^m$  or  $x_{(i,j)}^m$ , where  $m$  refers to a temporal ordering).

The basic building blocks of many features constructed as sequences and used in MRRMs are scalar features describing single elements of the network (Table I). We will in the following mainly consider two types of sequences: *one-level* sequences of scalar features and *two-level* sequences of sequences of scalar features.

**Definition III.2.** *One-level and two-level sequences.*

1. *One-level sequence.* We refer to a non-nested sequence of scalar features,  $\mathbf{x} = (x_q)_{q \in \mathcal{Q}}$ , as a *one-level sequence*.
2. *Two-level sequence.* We refer to a nested sequence of features that are themselves one-level sequences,  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$  with  $\mathbf{x}_q = (x_q^r)_{r \in \mathcal{R}_q}$ , as a *two-level sequence*. We refer to the  $\mathbf{x}_q$  as *local sequences*.

Detailed definitions of particular types of one- and two-level sequences are given in Table II (symbol:  $\mathbf{x}$ ).

One-level sequences are typically used to represent features that are aggregated over the temporal or topological dimension of the temporal network, while two-level sequences are composed of features that depend both on topology and time. The following examples illustrate this.

**Example III.1.** *One-level sequence of static degrees.* A well known example of an aggregated graph feature is the node degree,  $k_i$ , giving the number of nodes in  $G^{\text{stat}}$  that are connected to the node  $i$ . Figure 7(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.

**Example III.2.** *Two-level sequence of instantaneous degrees.* A generalization to temporal networks of the static degree  $k_i$  of a node is the *instantaneous degree*  $d_i^t$ . It is given by the number of nodes that the node  $i$  is in contact with at time  $t$ . The (*two-level*) sequence of instantaneous degrees is  $\mathbf{d} = (\mathbf{d}^t)_{t \in \mathcal{T}} = ((d_i^t)_{i \in \mathcal{V}})_{t \in \mathcal{T}}$ , or alternatively



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

Symbol	Meaning of symbol	Definition
$[t_{\min}, t_{\max}]$	Period of observation.	
$G$	(Instant-event) temporal network.	$G = (\mathcal{V}, \mathcal{C})$ (Def. II.1) <sup>a</sup> / $G = (\mathcal{V}, \mathcal{E})$ (Def. II.2) <sup>b</sup>
$\mathcal{V}$	Set of nodes in $G$ .	$\mathcal{V} = \{1, 2, \dots, N\}$
$\mathcal{C} / \mathcal{E}$	Set of (instantaneous) events in $G$ .	$\mathcal{C} = \{c_1, c_2, \dots, c_C\}$ <sup>a</sup> / $\mathcal{E} = \{e_1, e_2, \dots, e_E\}$ <sup>b</sup>
$c_q / e_q$	$q$ th (instantaneous) event.	$c_q = (i_q, j_q, t_q, \tau_q)$ <sup>a</sup> / $e_q = (i_q, j_q, t_q)$ <sup>b</sup>
$i_q, j_q$	Indices for nodes partaking in the $q$ th event.	
$t_q$	Start time of the $q$ th event.	
$\tau_q$	Duration of the $q$ th event. <sup>c</sup>	
$N$	Number of nodes in $G$ .	$N =  \mathcal{V} $
$C / E$	Number of events in $G$ .	$C =  \mathcal{C} $ <sup>a</sup> / $E =  \mathcal{E} $ <sup>b</sup>
<b>Link-timeline representation</b>		
$G_{\mathcal{L}}$	Link-timeline network.	$G_{\mathcal{L}} = (G^{\text{stat}}, \Theta)$ (Def. II.5)
$G^{\text{stat}}$	Static graph.	$G^{\text{stat}} = (\mathcal{V}, \mathcal{L})$
$\mathcal{L}$	Links in $G^{\text{stat}}$ .	$\mathcal{L} = \{(i, j) : (i, j, t, \tau) \in \mathcal{C}\}$ <sup>a</sup> / $\mathcal{L} = \{(i, j) : (i, j, t) \in \mathcal{E}\}$ <sup>b</sup>
$L$	Number of links in $G^{\text{stat}}$ .	$L =  \mathcal{L} $
$\mathcal{V}_i$	Neighborhood of node $i$ .	$\{j : (i, j) \in \mathcal{L}\}$
$\Theta$	Sequence of timelines.	$\Theta = (\Theta_{(i,j)})_{(i,j) \in \mathcal{L}}$
$\Theta_{(i,j)}$	Link timeline.	$\Theta_{(i,j)} = \left( (t_{(i,j)}^1, \tau_{(i,j)}^1), (t_{(i,j)}^2, \tau_{(i,j)}^2), \dots, (t_{(i,j)}^{n(i,j)}, \tau_{(i,j)}^{n(i,j)}) \right)$ <sup>a</sup> / $\Theta_{(i,j)} = \left( t_{(i,j)}^1, t_{(i,j)}^2, \dots, t_{(i,j)}^{w(i,j)} \right)$ <sup>b</sup>
<b>Snapshot-sequence representation</b> <sup>d</sup>		
$G_{\mathcal{T}}$	Snapshot-graph sequence	$G_{\mathcal{T}} = (\mathcal{T}, \Gamma)$ (Def. II.6) <sup>d</sup>
$\mathcal{T}$	Sequence of snapshot times.	$\mathcal{T} = (t_m)_{m=1}^T$ <sup>d</sup>
$\Gamma$	Sequence of snapshot graphs.	$\Gamma = (\Gamma^t)_{t \in \mathcal{T}}$ <sup>d</sup>
$\Gamma^t$	Snapshot graph at time $t$ .	$\Gamma^t = (\mathcal{V}, \mathcal{E}^t)$ <sup>d</sup>
$\mathcal{E}^t$	Instantaneous events at time $t$ .	$\mathcal{E}^t = \{(i, j) : (i, j, t) \in \mathcal{E}\}$ <sup>d</sup>
<b>Topological-temporal (two-level) features</b>		
$t_{(i,j)}^m$	Event start time.	Start time of $m$ th event in timeline $\Theta_{(i,j)}$ (Def. II.5)
$\tau_{(i,j)}^m$	Event duration.	Duration of $m$ th event in timeline $\Theta_{(i,j)}$ (Def. II.5) <sup>c</sup>
$\Delta\tau_{(i,j)}^m$	Inter-event duration.	$\Delta\tau_{(i,j)}^m = t_{(i,j)}^{m+1} - (t_{(i,j)}^m + \tau_{(i,j)}^m)$ <sup>a</sup> / $\Delta\tau_{(i,j)}^m = t_{(i,j)}^{m+1} - t_{(i,j)}^m$ <sup>b</sup>
$t_{(i,j)}^w$	End time of last event on timeline.	$t_{(i,j)}^w = t_{(i,j)}^{n(i,j)} + \tau_{(i,j)}^{n(i,j)}$ <sup>a</sup> / $t_{(i,j)}^w = t_{(i,j)}^{w(i,j)}$ <sup>b</sup>
$d_i^t$	Instantaneous degree at time $t$ .	$d_i^t =  \{j : (i, j, t, \tau) \in \mathcal{C} \text{ and } t' \leq t < t' + \tau\} $ <sup>a</sup> / $d_i^t =  \{j : (i, j) \in \mathcal{E}^t\} $ <sup>b</sup>
$v_i^m$	Activity start time.	Start time of $m$ th interval of consecutive activity of node $i$ .
$\alpha_i^m$	Activity duration.	Duration of $m$ th interval of consecutive activity of node $i$ . <sup>c</sup>
$\Delta\alpha_i^m$	Inactivity duration.	$\Delta\alpha_i^m = v_i^{m+1} - (v_i^m + \alpha_i^m)$ <sup>a</sup> / $\Delta\alpha_i^m = v_i^{m+1} - v_i^m$ <sup>b</sup>
<b>Aggregated (one-level) features</b>		
$n_{(i,j)}$	Link event frequency.	$n_{(i,j)} =  \Theta_{(i,j)} $ <sup>c</sup>
$w_{(i,j)}$	Link weight.	$w_{(i,j)} = \sum_{m=1}^{n(i,j)} \tau_{(i,j)}^m$ <sup>a</sup> / $w_{(i,j)} =  \Theta_{(i,j)} $ <sup>b</sup>
$a_i$	Node activity.	$a_i = \sum_{j \in \mathcal{V}_i} n_{(i,j)}$ <sup>c</sup>
$s_i$	Node strength.	$s_i = \sum_{j \in \mathcal{V}_i} w_{(i,j)}$
$k_i$	Node degree.	$k_i =  \mathcal{V}_i $
$A^t$	Cumulative activity at time $t$ .	$A^t = \sum_{i \in \mathcal{V}} d_i^t = 2 \mathcal{E}^t $
<b>Special features</b>		
$\Phi_i$	Node timeline.	$\Phi_i = \left( (v_i^1, \alpha_i^1), (v_i^2, \alpha_i^2), \dots, (v_i^{n_i^a}, \alpha_i^{n_i^a}) \right)$ <sup>a</sup> / $\Phi_i = \left( v_i^1, v_i^2, \dots, v_i^{n_i^a} \right)$ <sup>b</sup>
$\mathbb{I}_{\lambda}$	Indicator of connectedness of $G^{\text{stat}}$ .	$\mathbb{I}_{\lambda} = 1$ if $G^{\text{stat}}$ is connected, $\mathbb{I}_{\lambda} = 0$ otherwise.
$\text{iso}(\Gamma^t)$	Isomorphism class of $\Gamma^t$ .	Set of graphs obtained by all permutations of node indices in $\Gamma^t$ .

<sup>a</sup> Definition for a temporal network with event durations.

<sup>b</sup> Definition for an instant-event temporal network.

<sup>c</sup> Only defined for a temporal network with event durations.

<sup>d</sup> Only defined for an instant-event temporal network.

TABLE II: **Sequences, distributions, and moments of features.** Below,  $(\cdot)$  denotes an ordered sequence and  $[\cdot]$  denotes a multiset, equivalent to the empirical distribution.

Symbol	Meaning of symbol	Definition	Example(s)
$\mathbf{x}$	One-level sequence of link features.	$\mathbf{x} = (x_{(i,j)})_{(i,j) \in \mathcal{L}}$	Fig. 7(a)
	One-level sequence of node features.	$\mathbf{x} = (x_i)_{i \in \mathcal{V}}$	
	One-level sequence of snapshot features.	$\mathbf{x} = (x^t)_{t \in \mathcal{T}}$	
	Two-level sequence of link features.	$\mathbf{x} = (\mathbf{x}_{(i,j)})_{(i,j) \in \mathcal{L}}$ <sup>a</sup>	
	Two-level sequence of node features.	$\mathbf{x} = (\mathbf{x}_i)_{i \in \mathcal{V}}$ <sup>b</sup>	
$\pi_{\mathcal{L}}$	Sequence of local distributions on links.	$\pi_{\mathcal{L}}(\mathbf{x}) = (\pi_{(i,j)}(\mathbf{x}_{(i,j)}))_{(i,j) \in \mathcal{L}}$ <sup>d</sup>	Fig. 9(b)
$\pi_{\mathcal{V}}$	Sequence of local distributions on nodes.	$\pi_{\mathcal{V}}(\mathbf{x}) = (\pi_i(\mathbf{x}_i))_{i \in \mathcal{V}}$ <sup>e</sup>	Fig. 8(b)
$\pi_{\mathcal{T}}$	Sequence of local distributions in snapshots.	$\pi_{\mathcal{T}}(\mathbf{x}) = (\pi^t(\mathbf{x}^t))_{t \in \mathcal{T}}$ <sup>f</sup>	Fig. 8(e)
$p_{\mathcal{L}}$	Distribution of local sequences on links.	$p_{\mathcal{L}}(\mathbf{x}) = [\mathbf{x}_{(i,j)}]_{(i,j) \in \mathcal{L}}$ <sup>a</sup>	Fig. 9(c)
$p_{\mathcal{V}}$	Distribution of local sequences on nodes.	$p_{\mathcal{V}}(\mathbf{x}) = [\mathbf{x}_i]_{i \in \mathcal{V}}$ <sup>b</sup>	Fig. 8(c)
$p_{\mathcal{T}}$	Distribution of local sequences in snapshots.	$p_{\mathcal{T}}(\mathbf{x}) = [\mathbf{x}^t]_{t \in \mathcal{T}}$ <sup>c</sup>	Fig. 8(d)
$p$	Distribution of one-level link features.	$p(\mathbf{x}) = [x_{(i,j)}]_{(i,j) \in \mathcal{L}}$	Fig. 7(b)
	Distribution of one-level node features	$p(\mathbf{x}) = [x_i]_{i \in \mathcal{V}}$	
	Distribution of one-level snapshot features	$p(\mathbf{x}) = [x^t]_{t \in \mathcal{T}}$	
	Global distribution of two-level link features.	$p(\mathbf{x}) = \cup_{(i,j) \in \mathcal{L}} \pi_{(i,j)}(\mathbf{x}_{(i,j)})$ <sup>d</sup>	
	Global distribution of two-level node features.	$p(\mathbf{x}) = \cup_{i \in \mathcal{V}} \pi_i(\mathbf{x}_i)$ <sup>e</sup>	
$\mu_{\mathcal{L}}$	Sequence of local means on links.	$\mu_{\mathcal{L}}(\mathbf{x}) = (\mu_{(i,j)}(\mathbf{x}_{(i,j)}))_{(i,j) \in \mathcal{L}}$ <sup>g</sup>	Fig. 9(d)
$\mu_{\mathcal{V}}$	Sequence of local means on nodes.	$\mu_{\mathcal{V}}(\mathbf{x}) = (\mu_i(\mathbf{x}_i))_{i \in \mathcal{V}}$ <sup>h</sup>	Fig. 8(f)
$\mu_{\mathcal{T}}$	Sequence of local means in snapshots.	$\mu_{\mathcal{T}}(\mathbf{x}) = (\mu^t(\mathbf{x}^t))_{t \in \mathcal{T}}$ <sup>i</sup>	Fig. 8(i)
$\mu$	Mean of one-level link features.	$\mu(\mathbf{x}) = \sum_{(i,j) \in \mathcal{L}} x_{(i,j)} / L$	Fig. 7(c)
	Mean of one-level node features.	$\mu(\mathbf{x}) = \sum_{i \in \mathcal{V}} x_i / N$	
	Mean of one-level snapshot features.	$\mu(\mathbf{x}) = \sum_{t \in \mathcal{T}} x^t / T$	
	Global mean of two-level link features.	$\mu(\mathbf{x}) = \sum_{(i,j) \in \mathcal{L}} \sum_{m \in \mathcal{M}_{(i,j)}} x_{(i,j)}^m / (\sum_{(i,j) \in \mathcal{L}} M_{(i,j)})$	
	Global mean of two-level node features.	$\mu(\mathbf{x}) = \sum_{i \in \mathcal{V}} \sum_{m \in \mathcal{M}_i} x_i^m / (\sum_{i \in \mathcal{V}} M_i)$	
–	Feature is not conserved.		

<sup>a</sup>  $\mathbf{x}_{(i,j)}$ : Local sequence on link,  $\mathbf{x}_{(i,j)} = (x_{(i,j)}^m)_{m \in \mathcal{M}_{(i,j)}}$ , where  $\mathcal{M}_{(i,j)}$  is a temporally ordered index set.

<sup>b</sup>  $\mathbf{x}_i$ : Local sequence on node,  $\mathbf{x}_i = (x_i^m)_{m \in \mathcal{M}_i}$ , where  $\mathcal{M}_i$  is a temporally ordered index set.

<sup>c</sup>  $\mathbf{x}^t$ : Local sequence in snapshot,  $\mathbf{x}^t = (x_i^t)_{i \in \mathcal{V}}$ .

<sup>d</sup>  $\pi_{(i,j)}(\mathbf{x}_{(i,j)})$ : Local distribution on link,  $\pi_{(i,j)}(\mathbf{x}_{(i,j)}) = [x_{(i,j)}^m]_{m \in \mathcal{M}_{(i,j)}}$ .

<sup>e</sup>  $\pi_i(\mathbf{x}_i)$ : Local distribution on node,  $\pi_i(\mathbf{x}_i) = [x_i^m]_{m \in \mathcal{M}_i}$ .

<sup>f</sup>  $\pi^t(\mathbf{x}^t)$ : Local distribution in snapshot,  $\pi^t(\mathbf{x}^t) = [x_i^t]_{i \in \mathcal{V}}$ .

<sup>g</sup>  $\mu_{(i,j)}(\mathbf{x}_{(i,j)})$ : Local mean on link,  $\mu_{(i,j)}(\mathbf{x}_{(i,j)}) = \sum_{m \in \mathcal{M}_{(i,j)}} x_{(i,j)}^m / M_{(i,j)}$ .

<sup>h</sup>  $\mu_i(\mathbf{x}_i)$ : Local mean on node,  $\mu_i(\mathbf{x}_i) = \sum_{m \in \mathcal{M}_i} x_i^m / M_i$ .

<sup>i</sup>  $\mu^t(\mathbf{x}^t)$ : Local mean in snapshot,  $\mu^t(\mathbf{x}^t) = \sum_{i \in \mathcal{V}} x_i^t / N$ .

$\mathbf{d} = (\mathbf{d}_i)_{i \in \mathcal{V}} = ((d_i^t)_{t \in \mathcal{T}})_{i \in \mathcal{V}}$  since the order of the indices  $i$  and  $t$  does not matter here. Figure 8(a) shows the sequence of instantaneous degrees  $\mathbf{d}^*$  of the temporal network shown in Figs. 1 and 2.

**Example III.3.** *Two-level sequence of inter-event durations.* A feature of temporal networks that has been shown to have a profound impact on dynamic processes is the durations between consecutive events in the timelines, termed the *inter-event durations* and defined by  $\Delta\tau_{(i,j)}^m = t_{(i,j)}^{m+1} - (t_{(i,j)}^m + \tau_{(i,j)}^m)$ . Their (two-level) sequence is  $\Delta\tau = (\Delta\tau_{(i,j)})_{(i,j) \in \mathcal{L}}$ , where  $\Delta\tau_{(i,j)} = (\Delta\tau_{(i,j)}^m)_{m \in \mathcal{M}_{(i,j)}}$ . Here  $\mathcal{M}_{(i,j)} = \{1, 2, \dots, n_{(i,j)} - 1\}$

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. Note that due to the temporal extent of the inter-event durations, we cannot inverse the order of the indices  $m$  and  $(i, j)$  as we could for the instantaneous degrees presented in the previous example. Figure 9(a) shows the two level sequence of inter-event durations  $\Delta\tau^*$  of the temporal network in Fig. 1.

**Example III.4.** *Sequence of snapshot graphs.* A notable example of a sequence of features that are neither scalar nor sequences of scalars is the sequence of snapshot graphs  $\Gamma = (\Gamma^t)_{t \in \mathcal{T}}$  (Def. II.6). The sequence of snapshot graphs for the temporal network of Fig. 1 is shown

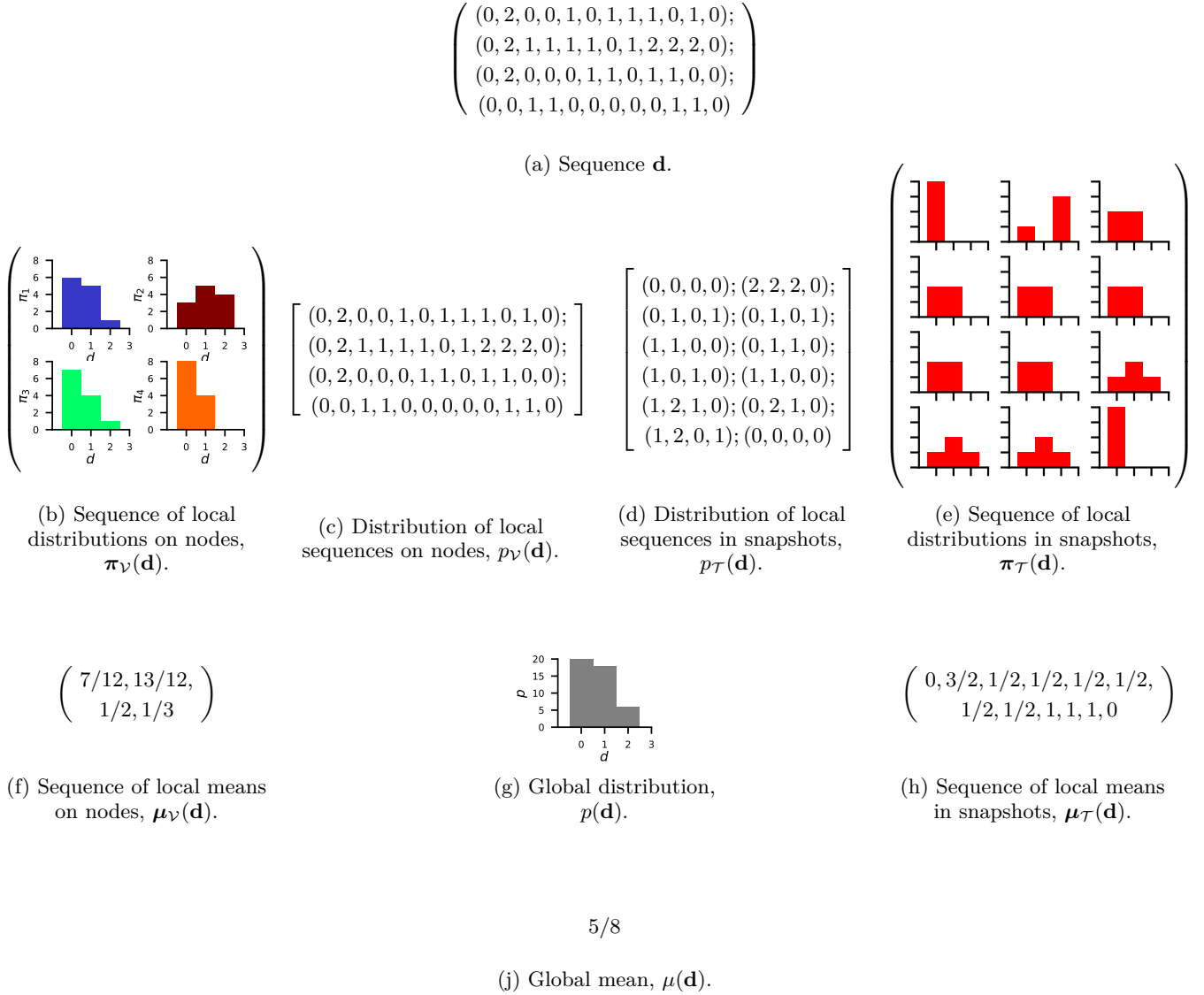


FIG. 8: **Example: Marginals and moments of the two-level sequence of instantaneous node degrees,  $\mathbf{d}$ .** (a) Sequence and (b)–(j) marginals and moments of the instantaneous degrees  $\mathbf{d}^*$  of the network shown in Figs. 1 and 2. Distributions are shown as multisets whenever this is most convenient.

in Fig. 2.

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 as a simpler example the degree sequence of a static graph.

**Example III.5.** 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. III.3 below), as well as their mean  $\mu(\mathbf{k})$  (Fig. 7). 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 features is possible than for a simple static graph. We define below those needed to characterize the MRRMs surveyed in this article, but many more may be defined (we provide an extended list in Supplementary Table I as well as hier-

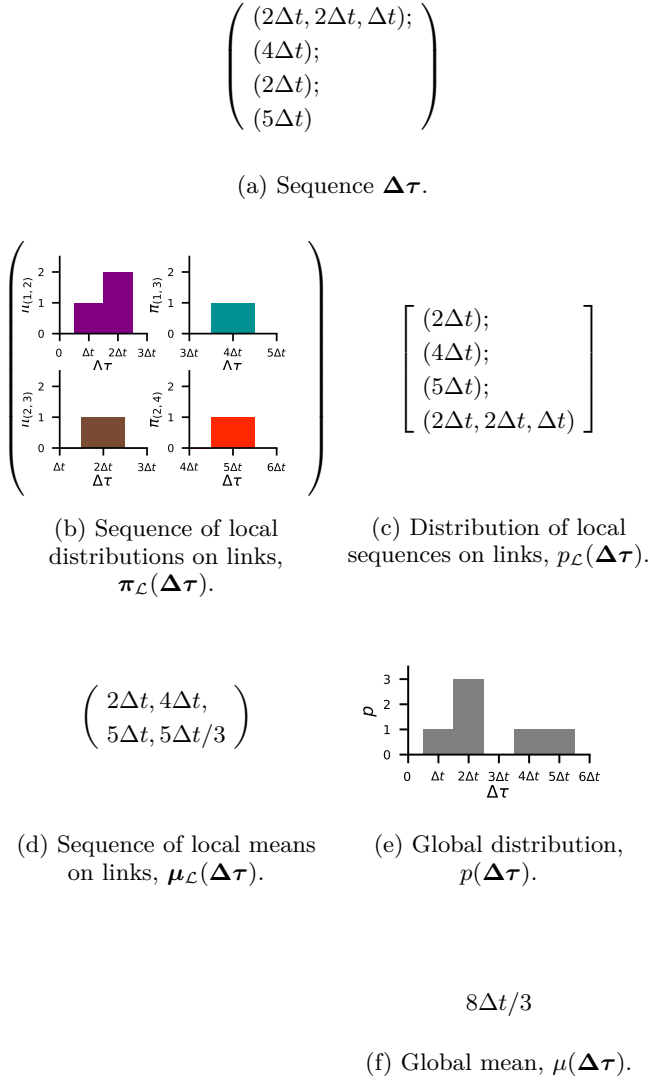


FIG. 9: **Example: Marginals and moments of the two-level sequence of inter-event durations,  $\Delta\tau$ .** (a) Sequence and (b)–(f) marginals and moments of the inter-event durations  $\Delta\tau$  for the network shown in Figs. 1 and 2. Distributions are shown as multisets whenever this is most convenient.

archies of this extended list of features in Supplementary Figs. 1 and 2).

**Definition III.3.** *Distribution of feature values.* Consider a sequence of features,  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$ , where the individual features  $\mathbf{x}_q$  may be scalar, sequences of scalars, or more general functions. The distribution of feature values, denoted  $p_{\mathcal{Q}'}(\mathbf{x})$ , returns the number of times each possible value of  $\mathbf{x}_q$  or of  $x_q^r \in \mathbf{x}_q$  appears in a measured sequence  $\mathbf{x}^* = \mathbf{x}(G^*)$ . Formally, the distribution can be defined as 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  $p_{\mathcal{Q}'}(\mathbf{x}^*) = [x_q^*]_{q \in \mathcal{Q}'}$ , where  $\mathcal{Q}'$  is the index set that

is marginalized over.

The index set  $\mathcal{Q}'$  that is marginalized over to construct the distribution  $p_{\mathcal{Q}'}(\mathbf{x})$  is not necessarily the same as the original  $\mathcal{Q}$  used to define the sequence  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$ . For distributions obtained from a one-level sequence  $\mathbf{x} = (x_q)_{q \in \mathcal{Q}}$ , we shall only consider  $\mathcal{Q}' = \mathcal{Q}$ , while for distributions over a two-level sequence  $\mathbf{x} = ((x_q^r)_{r \in \mathcal{R}_q})_{q \in \mathcal{Q}}$ , we may marginalize over the outer index  $\mathcal{Q}' = \mathcal{Q}$ , the inner indices  $\mathcal{R}_q$ , or both. Finally, we may also define a distribution over a sequence of features that are neither scalar nor sequences, e.g. snapshot graphs. The different ways to marginalize a sequence of features give rise to the following different three types of distributions.

**Definition III.4.** *Distributions over a sequence.* We here list different ways to construct distributions of feature values by marginalizing over a sequence  $\mathbf{x}$ . The different types of distributions are defined in more detail in Table II (symbols:  $p$ ,  $p_{\mathcal{L}}$ ,  $p_{\mathcal{V}}$ ,  $p_{\mathcal{T}}$ ,  $\pi_{\mathcal{L}}$ ,  $\pi_{\mathcal{V}}$ , and  $\pi_{\mathcal{T}}$ ).

1. *Global distribution  $p(\mathbf{x})$ .* The global distribution  $p(\mathbf{x})$  returns the number of times each possible scalar value appears in a measured sequence  $\mathbf{x}^* = \mathbf{x}(G^*)$ . For a one-level sequence  $\mathbf{x}^* = (x_q^*)_{q \in \mathcal{Q}}$ , it is obtained by marginalizing over the sole index set  $\mathcal{Q}$ :  $p(\mathbf{x}^*) = [x_q^*]_{q \in \mathcal{Q}}$ . For a two-level sequence  $\mathbf{x}^* = (((x_q^r)^*)_{r \in \mathcal{R}_q})_{q \in \mathcal{Q}}$ , it is obtained by marginalizing both over the inner and outer index sets,  $\mathcal{R}_q$  and  $\mathcal{Q}$ :  $p(\mathbf{x}^*) = [(x_q^r)^*]_{r \in \mathcal{R}_q, q \in \mathcal{Q}}$ . For simplicity, we have left out the subscripts in the notation  $p(\mathbf{x})$  since all indices are marginalized over.
2. *Distribution of local features  $p_{\mathcal{Q}}(\mathbf{x})$ .* For a sequence of non-scalar features,  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$ , the distribution of local features  $p_{\mathcal{Q}}(\mathbf{x}^*) = [\mathbf{x}_q^*]_{q \in \mathcal{Q}}$  reports the number of times each possible value of the local features  $\mathbf{x}_q$  appears in a measured sequence  $\mathbf{x}^*$ . Here each  $\mathbf{x}_q$  may be a local one-level sequence or a more general feature such as a graph.
3. *Sequence of local distributions  $\pi_{\mathcal{Q}}(\mathbf{x})$ .* For a two-level sequence of features,  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$ , the sequence of local distributions  $\pi_{\mathcal{Q}}(\mathbf{x})$  is given by the ordered tuple  $\pi_{\mathcal{Q}}(\mathbf{x}^*) = (\pi_q(\mathbf{x}_q^*))_{q \in \mathcal{Q}}$ , where each local distribution  $\pi_q(\mathbf{x}_q^*) = [(x_q^r)^*]_{r \in \mathcal{R}_q}$  is the distribution of the scalar features in the local sequence  $\mathbf{x}_q^*$ . To avoid confusion with the distribution of local sequences,  $p_{\mathcal{Q}}(\mathbf{x})$ , we use the different symbol  $\pi_{\mathcal{Q}}(\mathbf{x})$  to designate the sequence of local distributions.

The following examples illustrate the different types of distributions.

**Example III.6.** *Global distribution.* The global distribution  $p(\mathbf{k}^*) = [k_i^*]_{i \in \mathcal{V}}$  of the one-level static degrees of the temporal network of Figs. 1 and 2 is shown in Fig. 7(b). The global distributions of the two-level instantaneous degrees and inter-event durations,  $p(\mathbf{d}^*) = [(d_i^t)^*]_{t \in \mathcal{T}, i \in \mathcal{V}}$  and  $p(\Delta\tau^*) = [(\Delta\tau_{(i,j)}^m)^*]_{m \in \mathcal{M}_{(i,j)}, (i,j) \in \mathcal{L}}$

respectively, of the same network are shown in Figs. 8(g) and 9(e).

**Example III.7. Distribution of local features.** Two different types of distributions of local sequences of instantaneous degrees can be constructed from the sequence of instantaneous degrees: the distribution of local sequences of the instantaneous degrees of each node,  $p_{\mathcal{V}}(\mathbf{d}^*) = [\mathbf{d}_i^*]_{i \in \mathcal{V}}$ , shown in Fig. 8(c)], and the distribution of local sequences of instantaneous degrees of nodes in each snapshot,  $p_{\mathcal{T}}(\mathbf{d}^*) = [(\mathbf{d}^t)^*]_{t \in \mathcal{T}}$ , shown in Fig. 8(d) for the temporal network illustrated in Figs. 1 and 2. From the sequence of inter-event durations, we can construct the distribution of local sequences of inter-event durations on the links,  $p_{\mathcal{L}}(\Delta\tau^*) = [\Delta\tau_{(i,j)}^*]_{(i,j) \in \mathcal{L}}$ , shown in Fig. 9(c). From the snapshot-graph sequence, we can construct the distribution of snapshot graphs  $p_{\mathcal{T}}(\Gamma^*) = [(\Gamma^t)^*]_{t \in \mathcal{T}}$ .

**Example III.8. Sequence of local distributions.** We can also construct two different sequences of local distributions from the sequence of instantaneous degrees: the sequence of local distributions of the instantaneous degrees of each node,  $\pi_{\mathcal{V}}(\mathbf{d}^*) = [(d_i^t)^*]_{t \in \mathcal{T}}_{i \in \mathcal{V}}$ , shown in Fig. 8(b), and the sequence of local distributions of instantaneous degrees in each snapshot,  $\pi_{\mathcal{T}}(\mathbf{d}^*) = [(d_i^t)^*]_{i \in \mathcal{V}}_{t \in \mathcal{T}}$ , shown in Fig. 8(e). The sequence of local distributions of inter-event durations,  $\pi_{\mathcal{L}}(\Delta\tau^*) = [(\Delta\tau_{(i,j)}^m)^*]_{m \in \mathcal{M}_{(i,j)}}_{(i,j) \in \mathcal{L}}$ , is shown in Fig. 9(b). We cannot construct a sequence of local distributions from the sequence of snapshot graphs since they are not sequences.

After the above definitions of different local and global marginalizations of a sequence of features, 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. As for the distributions, it is natural to define the mean of a one- or two-level sequence simply as the average over the values of their scalar elements. For a two-level sequence, we may additionally construct a sequence of means of the local sequences.

**Definition III.5. Means of a one- or two-level sequence of features.** We shall consider two different ways to average over a one- or two-level sequence. Detailed definitions for specific kinds of features are given in Table II (symbols:  $\mu$ ,  $\mu_{\mathcal{L}}$ ,  $\mu_{\mathcal{V}}$ , and  $\mu_{\mathcal{T}}$ ).

1. *Global mean  $\mu(\mathbf{x})$ .* The global mean  $\mu(\mathbf{x})$  of a sequence of features is defined as the average over all individual scalar features in  $\mathbf{x}$ . For a one-level sequence, it is given by  $\mu(\mathbf{x}^*) = \sum_{q \in \mathcal{Q}} x_q^*/Q$ , where  $Q$  is the number of elements in  $\mathbf{x}^*$ . For a two-level sequence, it is  $\mu(\mathbf{x}^*) = \sum_{q \in \mathcal{Q}} \sum_{r \in \mathcal{R}_q} (x_q^r)^*/(\sum_{q \in \mathcal{Q}} R_q)$ , where  $R_q$  is the number of elements in  $\mathbf{x}_q^*$ .
2. *Sequence of local means  $\mu_{\mathcal{Q}}(\mathbf{x})$ .* For a two-level sequence  $\mathbf{x} = (\mathbf{x}_q)_{q \in \mathcal{Q}}$ , the sequence of local means

$\mu_{\mathcal{Q}}(\mathbf{x})$  is defined as the sequence of the means of each local sequence  $\mathbf{x}_q$ . Each of these *local* means is given by  $\mu_q(\mathbf{x}_q^*) = \sum_{r \in \mathcal{R}_q} (x_q^r)^*/R_q$ .

The different types of means obtained are illustrated in the following examples.

**Example III.9. Global mean.** The mean of the one-level sequence of static degrees,  $\mu(\mathbf{k}^*) = \sum_{i \in \mathcal{V}} k_i^*/N$ , of the temporal network in Fig. 1 is shown in Fig. 7(c). The global mean of the two-levels instantaneous degrees,  $\mu(\mathbf{d}^*) = \sum_{i \in \mathcal{V}} \sum_{t \in \mathcal{T}} (d_i^t)^*/(NT)$ , is shown in Fig. 8(j), and the global mean of the inter-event durations,  $\mu(\Delta\tau^*) = \sum_{(i,j) \in \mathcal{L}} \sum_{m \in \mathcal{M}_{(i,j)}} (\Delta\tau_{(i,j)}^m)^*/\sum_{(i,j) \in \mathcal{L}} M_{(i,j)}$ , is shown in Fig. 9(f).

**Example III.10. Sequence of local means.** Sequences of local means of the instantaneous degrees can be constructed in two ways: as the sequence of local means of the instantaneous degrees of each node,  $\mu_{\mathcal{V}}(\mathbf{d}^*) = (\sum_{t \in \mathcal{T}} (d_i^t)^*/T)_{i \in \mathcal{V}}$ , shown in Fig. 8(f), and as the sequence of local means of the instantaneous degrees in each snapshot,  $\mu_{\mathcal{T}}(\mathbf{d}^*) = (\sum_{i \in \mathcal{V}} (d_i^t)^*/N)_{t \in \mathcal{T}}$ , shown in Fig. 8(h). For the inter-event durations, as single sequence of local means can be constructed, namely the sequence of local means on each link,  $\mu_{\mathcal{L}}(\Delta\tau^*) = (\sum_{m \in \mathcal{M}_{(i,j)}} (\Delta\tau_{(i,j)}^m)^*/M_{(i,j)})_{(i,j) \in \mathcal{L}}$  [Fig. 9(d)].

The distributions and means defined above are functions of the sequence of features, 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 II.1. Table II lists all distributions and moments for features of links, nodes, and snapshots, and we establish their hierarchies in Fig. 10.

By combining Tables I and II, as shown in the examples above, we may describe most features constrained by MRRMs found in the literature.

Some of the different basic features listed in Table I are also pairwise comparable. This enables us to construct a hierarchy of the different features listed in Table I together with their marginals and moments (Table II). Figure 11 shows such a hierarchy. It may be used to derive which features are conserved by a MRRM that constrains a given feature: the MRRM 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. 11 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.

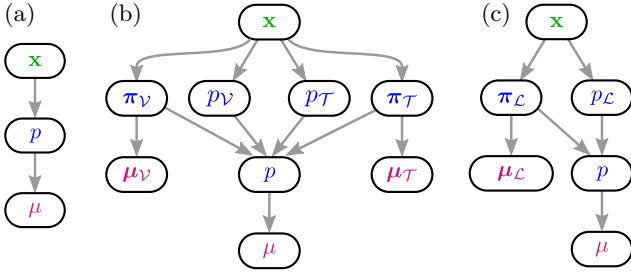


FIG. 10: **Hierarchies of the marginals and moments of a sequence of features.** An arrow from a higher node to a lower one indicates that the former feature is finer than the latter. Thus, a MRRM that conserves the former feature necessarily conserves all downstream features. Conversely, a MRRM that randomizes a given feature also randomizes any features upstream of it as well. (a) For one-level sequences of features, 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 features of nodes, namely  $\alpha$ ,  $\Delta\alpha$  and  $\mathbf{d}$ . (c) For two-level sequences of features of link timelines, namely  $\tau$  and  $\Delta\tau$ .

#### A. Notation for features of instant-event networks

Whenever possible we use the same symbols and names for features of instant-event temporal networks as for networks with event durations. However, we have adopted a different notation for the number of instantaneous events on a link,  $w_{(i,j)}$ , than for the number of events on a link,  $n_{(i,j)}$  (Table I). This is needed to make our description of MRRMs consistent when they are applied to both temporal networks and instant-event networks (see discussion in Section V B below). This furthermore means that  $s_i$  denotes the total number of instantaneous events that a node partakes in for instant-event networks, and that  $a_i$ , and  $n_{(i,j)}$  are not defined for these networks. Similarly the event durations  $\tau_{(i,j)}^m$  and activity durations  $\alpha_{(i,j)}^m$  are not defined for instant-event networks.

Several other definitions are changed slightly to accommodate the fact that instantaneous events do not have durations. This is namely the case for  $\mathcal{L}$ ,  $\Theta_{(i,j)}$ ,  $\Delta\tau_{(i,j)}^m$ ,  $t_{(i,j)}^w$ ,  $\Delta\alpha_{(i,j)}^m$ , and  $\Phi_i$  (Table I). Conversely, the snapshot-graph sequence is only defined for instant-event networks, and thus so are also the following associated features:  $\mathcal{T}$ ,  $\Gamma^t$ , and  $\mathcal{E}^t$ .

## IV. SHUFFLING METHODS

We give in this section a description of several important shuffling methods. These are used to formulate and generate MRRMs 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 (Def. II.13), and we discuss how this can be used in the design of shuffling methods.

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

Since all event and instant-event shufflings constrain  $\mathcal{V}$ ,  $[t_{\min}, t_{\max}]$ , and  $E$ , we shall omit these from the names of methods in order to avoid clutter.

**Definition IV.1. Event shuffling.** We define an *event shuffling* as a shuffling 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  $p(\tau)$ .

**Definition IV.2. Instant-event shuffling.** We define an *instant-event shuffling* as a shuffling 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 instantaneous 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 shuffling method that randomizes everything but  $\mathcal{V}$ ,  $[t_{\min}, t_{\max}]$ , and  $p(\tau)$ . Leaving out  $\mathcal{V}$  and  $[t_{\min}, t_{\max}]$  since all MRRMs we consider constrain these, it is named  $P[p(\tau)]$ . The coarsest possible instant-event shuffling, which constrains only  $\mathcal{V}$ ,  $\mathcal{T}$ , and  $E$ , is recognized as the unity element (Def. II.10) and is consequently named  $P[1]$ .

Formally speaking any event shuffling is a refinement of an instant-event shuffling, so all event shufflings are also instant-event shufflings. In practice, any event shuffling may be applied to shuffle an instant-event network, where it then simply corresponds to the instant-event shuffling that conserves all the same features except  $p(\tau)$ . It is useful however to distinguish between the two since instant-event shufflings may be used to randomize the event durations of temporal networks, differently from any event shuffling. This is done by first representing each event in the temporal network by a sequence of consecutive instantaneous events and then shuffling this instant-event network using the chosen instant-event shuffling. Contiguous series of events are then concatenated to turn the generated instant-event temporal network into a temporal network with randomized event durations (see Section V B below).

We will now 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 representa-

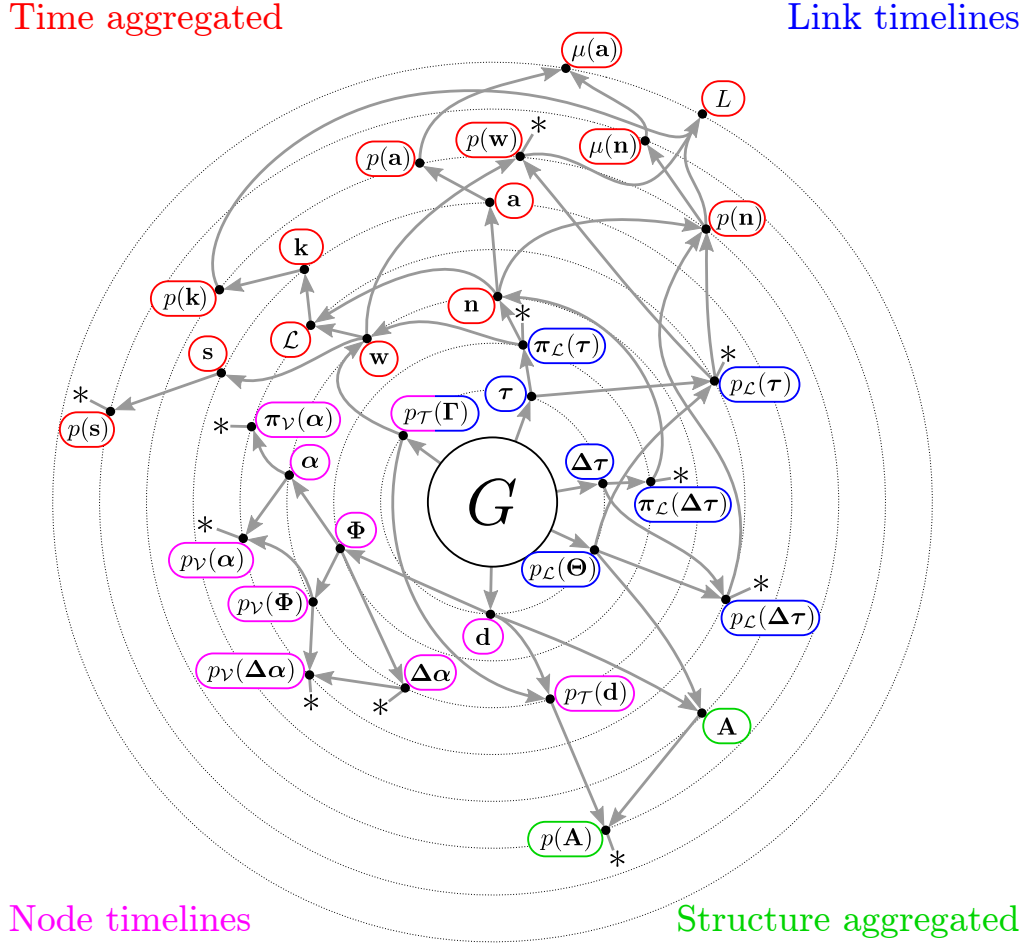


FIG. 11: **Feature hierarchy diagram.** An arrow from a higher ranking (more central) to a lower ranking feature (vertex in the diagram) indicates that the former feature is finer than the latter. Thus, a MRRM that constrains a given feature also constrains all downstream features. Conversely, a MRRM that randomizes (i.e. does not constrain) a given feature does not constrain any of the upstream features 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. 10. The color coding shows what type of features the features correspond to: time-aggregated features (i.e. topological and weighted), link-timeline features, node-timeline features, and structure-aggregated features (i.e. purely temporal).

tions introduced in Section II 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 compatible. This lets us generate new microcanonical RRM as compositions of these.

#### A. Link and timeline shufflings

**Definition IV.3.** *Link shuffling.* A link shuffling  $P[\mathbf{f}(\mathcal{L}), p_{\mathcal{L}}(\Theta)]$  is an event or instant-event shuffling that constrains all the individual timelines, i.e. the multiset  $p_{\mathcal{L}}(\Theta) = [\Theta_{(i,j)}]_{(i,j) \in \mathcal{L}}$ . 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 ( $P[\mathbf{k}]$ ) or the Erdős-Rényi model ( $P[L]$ ), and redistributing the timelines  $\Theta_{(i,j)} \in p_{\mathcal{L}}(\Theta)$  at random on the new links without replacement. The coarsest link shuffling is  $P[p_{\mathcal{L}}(\Theta)]$ , which shuffles the static graph using  $P[L]$ .

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

Timeline shufflings conserve the static graph  $G^{\text{stat}}$  by construction. The coarsest timeline instant-event shuffling is  $P[\mathcal{L}]$ , while the coarsest timeline event shuffling is  $P[\mathcal{L}, p(\tau)]$ .

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

*Proof.* It is clear that the content of the individual timelines  $\Theta_{(i,j)} \in p_{\mathcal{L}}(\Theta)$  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  $\Omega_{\mathcal{L}', p_{\mathcal{L}}(\Theta^*)} = \Omega_{\mathcal{L}'', p_{\mathcal{L}}(\Theta^*)}$  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  $\Omega_{\mathcal{L}', L^*} \propto \Omega_{\mathcal{L}', p_{\mathcal{L}}(\Theta^*)}$  for all  $L'$ , and since the conditional probabilities must be normed, that  $P_{\mathcal{L}|p_{\mathcal{L}}(\Theta)}(\mathcal{L}^\dagger|p_{\mathcal{L}}(\Theta^*)) = P_{\mathcal{L}|L}(\mathcal{L}^\dagger|L^*)$ , i.e. that  $\mathcal{L}$  and  $p_{\mathcal{L}}(\Theta)$  are independent conditioned on  $L$ . Since  $L \geq \mathcal{L}$  and  $L \geq p_{\mathcal{L}}(\Theta)$ , it then follows from Theorem 1 that  $\mathcal{L}$  and  $p_{\mathcal{L}}(\Theta)$  are compatible. This shows that the coarsest link shuffling,  $P[p_{\mathcal{L}}(\Theta)]$ , and the coarsest timeline shuffling,  $P[\mathcal{L}]$ , are compatible. We next note that any link and timeline shufflings are adapted refinements of  $P[p_{\mathcal{L}}(\Theta)]$  and  $P[\mathcal{L}]$ , respectively (compare Defs. IV.3 and IV.4 with Def. II.16). So applying Theorem 2 gives that any link shuffling  $P[p_{\mathcal{L}}(\Theta), \mathbf{f}(\mathcal{L})]$  and any timeline shuffling  $P[\mathcal{L}, \mathbf{g}(p_{\mathcal{L}}(\Theta))]$  are compatible and that their composition is  $P[L, \mathbf{f}(\mathcal{L}), \mathbf{g}(p_{\mathcal{L}}(\Theta))]$ .  $\square$

## B. Sequence and snapshot shufflings

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

The coarsest sequence shuffling is  $P[p_{\mathcal{T}}(\Gamma)]$ . As noted in the definition above, all sequence shufflings are timeline shufflings as  $p_{\mathcal{T}}(\Gamma) \leq \mathcal{L}$ . 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 IV.6.** *Snapshot shuffling.* A snapshot shuffling  $P[\mathbf{t}, \mathbf{f}(p_{\mathcal{T}}(\Gamma))]$  is an event or instantaneous-event shuffling that constrains the time of each event, i.e.  $\mathbf{t}$ . It randomizes each snapshot graph  $\Gamma^t$  individually in a manner that may be constrained by any function  $\mathbf{f}$  of  $p_{\mathcal{T}}(\Gamma)$ .

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

Sequence and snapshot shufflings are naturally defined as instant-event shufflings since they rely on randomizing either the order of temporal snapshots and the events inside each snapshot, respectively. It is however possible to define snapshot event shufflings which conserve event durations of a temporal network (see Section V G), while it is not generally possible to design microcanonical sequence shufflings that conserve event durations in practice since event durations induce correlations between neighboring snapshots.

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

*Proof.* Following the same reasoning as in the proof of Proposition IV.1, we note that  $\Omega_{\mathbf{t}', p_{\mathcal{T}}(\Gamma^*)} = \Omega_{\mathbf{t}', p(\mathbf{A}^*)}$  for all  $\mathbf{t}'$  and that  $p(\mathbf{A})$  satisfies  $p(\mathbf{A}) \geq \mathbf{t}$  and  $p(\mathbf{A}) \geq p_{\mathcal{T}}(\Gamma)$ . Thus,  $\mathbf{t}$  is independent of  $p_{\mathcal{T}}(\Gamma)$  conditioned on  $p(\mathbf{A})$ . So the coarsest sequence and snapshot shufflings,  $P[p_{\mathcal{T}}(\Gamma)]$  and  $P[\mathbf{t}]$  are compatible by Theorem 1. Consequently, since all sequence shufflings  $P[\mathbf{f}(\mathbf{t}), p_{\mathcal{T}}(\Gamma)]$  and snapshot shufflings  $P[\mathbf{A}, \mathbf{f}(p_{\mathcal{T}}(\Gamma))]$  are adapted refinements of  $P[p_{\mathcal{T}}(\Gamma)]$  or  $P[\mathbf{t}]$ , respectively (Def. II.16), Theorem 2 tells us that they are compatible and that their composition is  $P[p(\mathbf{A}), \mathbf{f}(\mathbf{t}), \mathbf{g}(p_{\mathcal{T}}(\Gamma))]$ .  $\square$

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

*Proof.* Since sequence shufflings are timeline shufflings, they are by virtue of Proposition IV.1 also compatible with link shufflings.  $\square$

## V. CLASSIFYING RANDOMIZED REFERENCE MODELS

In this section we describe and classify MRRMs found in the literature based on the theory developed in Sections II and IV and the features defined in Section III. We also introduce several new MRRMs, which primarily are event-shuffling versions of existing instant-event shufflings and vice-versa. We provide unambiguous canonical names for the MRRMs (see Def. V.1 below) and we describe how they affect the temporal network features listed in Table I.

All the MRRMs described here fall in two general classes depending on whether they are applied to temporal networks with event durations (Def. II.1) or to temporal networks with instantaneous events (Def. II.2): *event shufflings* (Def. IV.1) shuffle events in temporal networks while conserving event durations; *instant-event shufflings* (Def. IV.2) shuffle instantaneous events in instant-event temporal networks. Note that it is possible to randomize the event durations of a temporal network by first discretizing its events and then shuffling this instant-event



network using an instant-event shuffling. We describe how this can be done in detail below.

It is also possible to combine two different MRRMs to form a new MRRM by applying the second MRRM to each graph in the ensemble generated by the first. This defines a *composition* of the MRRMs (Def. II.12) and results in a model that 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* (Def. II.13). As shown in Section IV, link shufflings and timeline shufflings are compatible (Proposition IV.1), while sequence shufflings are compatible with both snapshot shufflings and link shufflings (Propositions IV.2 and IV.3). Some MRRMs generated by composition of two other MRRMs are listed in Subsection VI, 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 given by the number of pairs of compatible MRRMs that are not comparable (for the shufflings listed in Table III, this number is 156).

The MRRMs are described in detail below and their effects on network features are summarized in Table III. Three figures additionally provide hierarchical orderings of the MRRMs: Fig. 12 shows the hierarchy of link shufflings (Def. IV.3); Fig. 13 shows the hierarchy of timeline shufflings (Def. IV.4), including sequence shufflings (Def. IV.5); Fig. 14 shows the hierarchy of snapshot shufflings (Def. IV.6).

This section is organized as follows. Subsection V A first defines the naming convention we will use for MRRMs for temporal networks. Subsection V B describes how to apply instant-event shufflings to generate MRRMs that randomize event durations of temporal networks. Subsection V C presents the basic, i.e. the *coarsest* (Def. II.9), instant-event and event shufflings. Subsections V D–V G present the four restricted classes of shufflings defined in Section IV, namely link shufflings (V D), timeline shufflings (V E), sequence shufflings (V F), and snapshot shufflings (V G). Event and instant-event shufflings are presented in separate lists in each subsection. Subsection V H describes several MRRMs that can be classified as intersections of link and timeline shufflings or of timeline and snapshot shufflings. Subsection V I describes compositions of MRRMs found in the literature. Subsection V J discusses microcanonical RRM models that use additional metadata on nodes. Subsection V K finally surveys reference models that are not MRRMs. These are namely *canonical* RRM models and reference models that do not maximize entropy. Of the latter we discuss in particular models based on bootstrapping.

### A. Naming convention

**Definition V.1.** *Naming convention for MRRMs for*

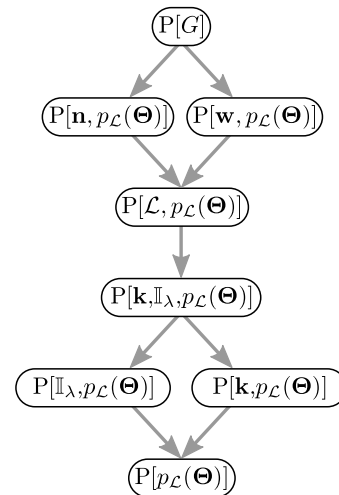


FIG. 12: **Hierarchy of link shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features and Section V D for definitions of the link shufflings.  $P[\mathbf{n}, p_{\mathcal{L}}(\Theta)]$ ,  $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$ , and  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  are also timeline shufflings (Section V E).

*temporal networks.* A MRRM is completely defined by the feature(s) it constrains (Def. II.7), so we use a naming 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 Subsection V H below). However it is unambiguous as a set of features uniquely defines a single MRRM (Def. II.7). This means that a name always uniquely defines a MRRM.

Since MRRMs in practice always constrain the set of nodes  $\mathcal{V}$ , the recording time-interval  $[t_{\max}, t_{\min}]$ , and the number of instantaneous events  $E$ , we exclude these features from the canonical names of MRRMs in order to avoid clutter.

### B. Applying instant-event shufflings to temporal networks with event durations

Event shufflings by definition conserve the events' durations in a temporal network, but we may randomize the durations by first representing the temporal network as an instant-event network and then applying an instant-event shuffling to it using the following procedure:

- i) Choose an appropriate time-resolution for discretization. A natural choice may be the time-resolution of recordings, but for high resolution measurements a lower time-resolution may be more practical.
- ii) Construct the corresponding instant-event temporal network by defining an instantaneous event between a

TABLE III: **Effects of MRRMs on features of temporal networks.** See Tables I and II for symbol definitions and Def. V.1 for naming. Note that a feature that is not constrained (–) by a shuffling method is not necessarily completely randomized either (see discussion at the end of Sec. III).

Canonical name	Common name*	One-level						Two-level								
		topological		weighted				temp.	node			link				
		$G^{\text{stat}}$	$k_i$	$L$	$a_i^\dagger$	$s_i$	$n_{(i,j)}^\dagger$	$w_{(i,j)}$	$A^t$	$\alpha_i^{m\dagger}$	$\Delta\alpha_i^m$	$d_i^t$	$\tau_{(i,j)}^m$	$\Delta\tau_{(i,j)}^m$	$t_{(i,j)}^1$	$t_{(i,j)}^w$
<b>Instant-event shufflings:</b>																
P[1]	<i>Instant-event shuffling</i>	–	–	–	–	$\mu$	–	–	$\mu$	–	–	$\mu$	–	–	–	–
<i>Timeline shufflings:</i>																
P[ $\mathcal{L}$ ]	<i>Timeline shuffling</i>	x	x	x	–	$\mu$	–	$\mu$	$\mu$	–	–	$\mu$	–	–	–	–
P[ $\mathbf{w}$ ]		x	x	x	–	x	–	x	$\mu$	–	–	$\mu$	–	–	–	–
P[ $\mathbf{w}, \mathbf{t}^1, \mathbf{t}^w$ ]		x	x	x	–	x	–	x	$\mu$	–	–	$\mu$	–	–	x	x
P[ $\pi_{\mathcal{L}}(\Delta\tau)$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	$\mu_{\mathcal{L}}$	$\pi_{\mathcal{L}}$	–	–
P[ $\pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^1, \mathbf{t}^w$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	$\mu_{\mathcal{L}}$	$\pi_{\mathcal{L}}$	x	x
<i>Sequence shufflings:</i>																
P[ $p_{\mathcal{T}}(\Gamma)$ ]	<i>Sequence shuffling</i>	x	x	x	–	x	–	x	$p$	–	–	$p_{\mathcal{T}}$	–	–	–	–
P[ $p_{\mathcal{T}}(\Gamma), \text{sgn}(\mathbf{A})$ ]		x	x	x	–	x	–	x	$p, \text{sgn}$	–	–	$p_{\mathcal{T}}$	–	–	–	–
<i>Snapshot shufflings:</i>																
P[ $\mathbf{t}$ ]	<i>Snapshot shuffling</i>	–	–	–	–	$\mu$	–	–	x	–	–	$\mu_{\mathcal{T}}$	–	–	–	–
P[ $\mathbf{t}, \Phi$ ]		–	–	–	–	$\mu$	–	–	x	x	x	$\mu_{\mathcal{T}}$	–	–	–	–
P[ $\mathbf{d}$ ]		–	–	–	–	$\mu$	–	–	x	x	x	x	–	–	–	–
P[ $\text{iso}(\Gamma)$ ]		–	–	–	–	$\mu$	–	–	x	–	–	$\pi_{\mathcal{T}}$	–	–	–	–
P[ $\text{iso}(\Gamma), \Phi$ ]		–	–	–	–	$\mu$	–	–	x	x	x	$\pi_{\mathcal{T}}$	–	–	–	–
<i>Intersections:</i>																
P[ $\mathcal{L}, \mathbf{t}$ ]		x	x	x	–	$\mu$	–	$\mu$	x	–	–	$\mu_{\mathcal{T}}$	–	–	–	–
P[ $\mathbf{w}, \mathbf{t}$ ]	<i>Timestamp shuffling</i>	x	x	x	–	x	–	x	x	–	–	$\mu_{\mathcal{T}}$	–	–	–	–
<i>Compositions:</i>																
P[ $L$ ]		–	$\mu$	x	–	$\mu$	–	$\mu$	$\mu$	–	–	$\mu$	–	–	–	–
P[ $\mathbf{k}, p(\mathbf{w}), \mathbf{t}$ ]		–	x	x	–	$\mu$	–	$p$	x	–	–	$\mu_{\mathcal{T}}$	–	–	–	–
P[ $\mathbf{k}, \mathbb{I}_\lambda, p(\mathbf{w}), \mathbf{t}$ ]		$\mathbb{I}_\lambda$	x	x	–	$\mu$	–	$p$	x	–	–	$\mu_{\mathcal{T}}$	–	–	–	–
<b>Event shufflings:</b>																
P[ $p(\tau)$ ]	<i>Event shuffling</i>	–	–	–	–	$\mu$	–	–	$\mu$	–	–	$\mu$	$p$	–	–	–
<i>Link shufflings:</i>																
P[ $p_{\mathcal{L}}(\Theta)$ ]	<i>Link shuffling</i>	–	$\mu$	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
P[ $\mathbb{I}_\lambda, p_{\mathcal{L}}(\Theta)$ ]		$\mathbb{I}_\lambda$	$\mu$	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
P[ $\mathbf{k}, p_{\mathcal{L}}(\Theta)$ ]	<i>Maslov-Sneppen</i>	–	x	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
P[ $\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\Theta)$ ]		$\mathbb{I}_\lambda$	x	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
<i>Timeline shufflings:</i>																
P[ $\mathcal{L}, p(\tau)$ ]	<i>Timeline shuffling</i>	x	x	x	–	$\mu$	$\mu$	$\mu$	$\mu$	–	–	$\mu$	$p$	–	–	–
P[ $\pi_{\mathcal{L}}(\tau)$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	$\pi_{\mathcal{L}}$	–	–	–
P[ $\pi_{\mathcal{L}}(\tau), \mathbf{t}^1, \mathbf{t}^w$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	$\pi_{\mathcal{L}}$	$\mu_{\mathcal{L}}$	x	x
P[ $\pi_{\mathcal{L}}(\tau), \pi_{\mathcal{L}}(\Delta\tau)$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	$\pi_{\mathcal{L}}$	$\pi_{\mathcal{L}}$	–	–
P[ $\pi_{\mathcal{L}}(\tau), \pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^1$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	$\pi_{\mathcal{L}}$	$\pi_{\mathcal{L}}$	x	x
P[ $\text{per}(\Theta)$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	x	x	–	–
P[ $\tau, \Delta\tau$ ]		x	x	x	x	x	x	x	$\mu$	–	–	$\mu$	x	x	–	–
<i>Snapshot shufflings:</i>																
P[ $p(\mathbf{t}, \tau)$ ]		–	–	–	–	$\mu$	–	–	x	–	–	$\mu_{\mathcal{T}}$	$\pi_{\mathcal{T}}$	–	–	–
<i>Intersections:</i>																
P[ $\mathcal{L}, p(\mathbf{t}, \tau)$ ]		x	x	x	–	$\mu$	$\mu$	$\mu$	x	–	–	$\mu_{\mathcal{T}}$	$\pi_{\mathcal{T}}$	–	–	–
P[ $\mathbf{n}, p(\mathbf{t}, \tau)$ ]	<i>Timestamp shuffling</i>	x	x	x	x	$\mu$	x	$\mu$	x	–	–	$\mu_{\mathcal{T}}$	$\pi_{\mathcal{T}}$	–	–	–
P[ $\mathcal{L}, p_{\mathcal{L}}(\Theta)$ ]		x	x	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
P[ $\mathbf{w}, p_{\mathcal{L}}(\Theta)$ ]		x	x	x	$\mu$	x	$p$	x	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
P[ $\mathbf{n}, p_{\mathcal{L}}(\Theta)$ ]		x	x	x	x	$\mu$	x	$p$	x	–	–	$\mu_{\mathcal{T}}$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$

\* Multiple common names for a MRRM may exist; in these cases we report the others in the main text.

† Feature only defined for temporal networks with event durations.

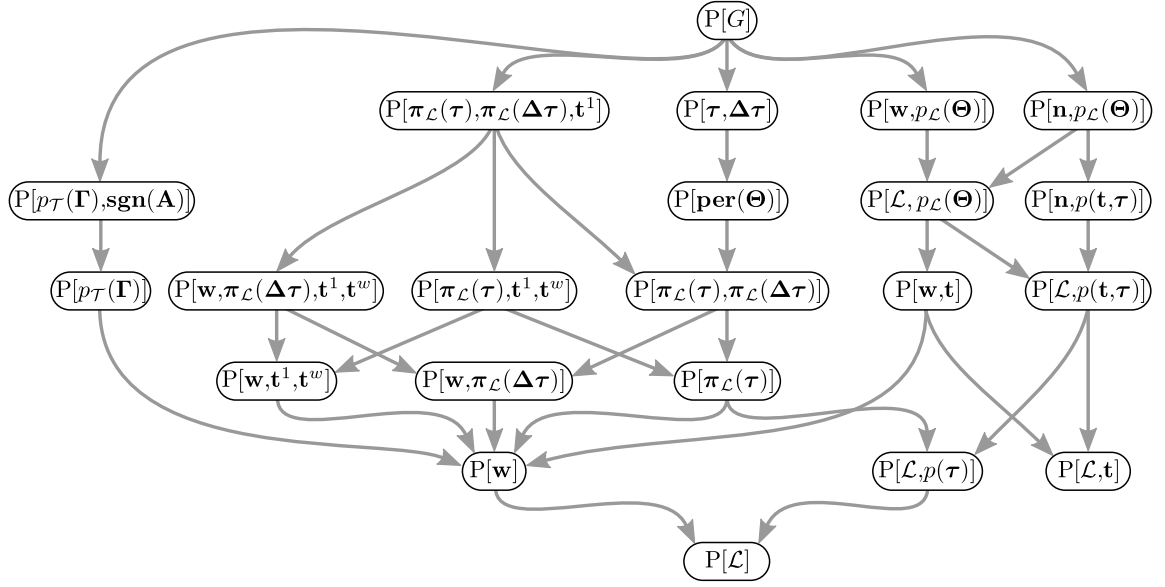


FIG. 13: **Hierarchy of timeline shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features and Section V E for definitions of the timeline shufflings.  $P[p_{\mathcal{T}}(\Gamma), \text{sgn}(\mathbf{A})]$  and  $P[p_{\mathcal{T}}(\Gamma)]$  are sequence shufflings (Section V F).  $P[\mathbf{n}, p_{\mathcal{L}}(\Theta)]$ ,  $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$ , and  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  are also timeline shufflings (Section V E), and  $P[\mathbf{n}, p(\mathbf{t}, \tau)]$ ,  $P[\mathcal{L}, p(\mathbf{t}, \tau)]$ ,  $P[\mathbf{w}, \mathbf{t}]$ , and  $P[\mathcal{L}, \mathbf{t}]$  are also snapshot shufflings (Section V G).

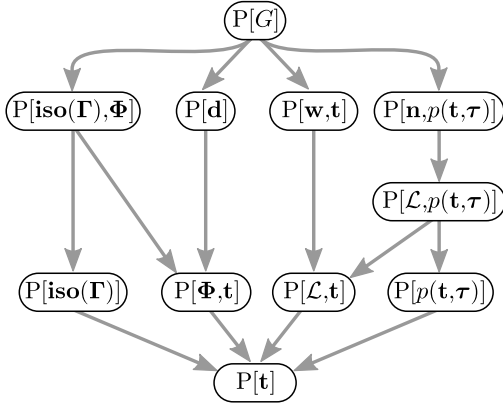


FIG. 14: **Hierarchy of snapshot shufflings.** An arrow from a higher MRRM to a lower one indicates that the former MRRM is finer than the latter and thus strictly more constrained. See Tables I and II for definitions of the features and Section V G for definitions of the snapshot shufflings.  $P[\mathbf{n}, p(\mathbf{t}, \tau)]$ ,  $P[\mathcal{L}, p(\mathbf{t}, \tau)]$ ,  $P[\mathbf{w}, \mathbf{t}]$ , and  $P[\mathcal{L}, \mathbf{t}]$  are also timeline shufflings (Section V E).

pair of nodes at the start of each time-interval during which the nodes are in contact (in the temporal network shown Fig. 1 this splits the four long events into two instantaneous events each and creates one instantaneous event for each of the shorter events).

iii) Randomize this instant-event network using an instant-event shuffling.

iv) Recreate a randomized version of the temporal network by concatenating consecutive instantaneous events between the same nodes into single events.

Table III show the effects of both event and instant-event shufflings on the features of a temporal network. To understand the effects of the same shufflings on an instant-event network one should simply ignore the features that are not defined for instant-event temporal networks, i.e.  $a_i$ ,  $n_{(i,j)}$ ,  $\alpha_{(i,j)}^m$  and  $\tau_{(i,j)}^m$ .

From the above procedure, the number of instantaneous events on a link in the generated instant-event network is seen to correspond to the weight  $w_{(i,j)}$  of the link in the original temporal network. Using  $w_{(i,j)}$  to designate the number of instantaneous events on the links in the instant-event network thus makes it possible to name each event and instant-event shuffling based on the features they conserve in a consistent manner, no matter whether the shuffling is applied to a temporal network or an instant-event temporal network. It follows from this alternative definition of  $w_{(i,j)}$  that  $s_i$  designates the activity of the node  $i$  in an instant-event network (see Section III A for further discussion).

### C. The basic instant-event and event shufflings

We first present the coarsest possible instant-event and event shufflings.

### 1. Instant-event shuffling

$P[1]$  shuffles the instantaneous events at random without any constraints.  $P[1]$  is thus the coarsest instant-event shuffling possible, and more generally the coarsest MRRM in the space of all MRRMs that conserve the nodes  $\mathcal{V}$ , the recording interval  $[t_{\min}, t_{\max}]$  and the number of events  $E$ .  $P[1]$  was employed in Ref. [44].

### 2. Event shuffling

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

## D. Link shufflings

Link shufflings (Def. IV.3) alter the aggregate network topology but conserve temporal structure locally on each link. Link shufflings always conserve  $p_{\mathcal{L}}(\Theta)$  and all coarser features (see Fig. 11), while time-aggregated features may be constrained or randomized depending on the link shuffling. Link shufflings are compatible with timeline shufflings (Subsec. V E) and with sequence shufflings (Subsec. V F).

All link shufflings may be defined as event shufflings since they automatically constrain the contact durations (i.e. since  $p(\tau) \geq p_{\mathcal{L}}(\Theta)$ ). They are ordered hierarchically in the Hasse diagram shown in Fig. 12.

### 1. Event shufflings

$P[p_{\mathcal{L}}(\Theta)]$  shuffles the links and associated timelines between all node pairs  $(i, j)$  without any constraints on the static network structure. This corresponds to drawing  $G^{\text{stat}}$  uniformly from the ensemble of all Erdős-Rényi (ER) [1] random graphs with the same nodes  $\mathcal{V}$  and number of links  $L$  as the original network and redistributing the timelines  $\Theta_{(i,j)} \in p_{\mathcal{L}}(\Theta)$  on the new links at random.  $P[p_{\mathcal{L}}(\Theta)]$  was employed in Ref. [33] where it was referred to as the *Erdős-Rényi model*. It is the coarsest link shuffling, so we may simply refer to it as *link shuffling*.

$P[\mathbb{I}_{\lambda}, p_{\mathcal{L}}(\Theta)]$  adds the additional constraint to  $P[p_{\mathcal{L}}(\Theta)]$  that the static network of the sampled reference networks must be connected (if it was in the input network).  $P[\mathbb{I}_{\lambda}, p_{\mathcal{L}}(\Theta)]$  was called *rewiring* in Ref. [47] and *random network* in Ref. [31].

$P[\mathbf{k}, p_{\mathcal{L}}(\Theta)]$  shuffles 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 [15] or by using the configuration model [2].  $P[\mathbf{k}, p_{\mathcal{L}}(\Theta)]$  was called *randomized edges* in Refs. [20, 46, 53] *random link shuffling* in Ref. [21], and *randomized structure* in Ref. [40].

$P[\mathbf{k}, \mathbb{I}_{\lambda}, p_{\mathcal{L}}(\Theta)]$  adds the additional constraint to  $P[\mathbf{k}, p_{\mathcal{L}}(\Theta)]$  that the static network must be connected.  $P[\mathbf{k}, \mathbb{I}_{\lambda}, p_{\mathcal{L}}(\Theta)]$  was called *configuration model* in Ref. [31].

## E. Timeline shufflings

Timeline shufflings (Def. IV.4) randomize the individual timelines  $\Theta_{(i,j)}$  without changing the topology of the aggregated network. Thus, they always constrain  $G^{\text{stat}}$  and all coarser features (see Fig. 11), while they typically randomize temporal-topological features of both links and nodes. Timeline shufflings are compatible with link shufflings (Subsec. V D).

The timeline shufflings listed below are ordered hierarchically in the Hasse diagram shown in Fig. 13.

### 1. Instant-event shufflings

$P[\mathcal{L}]$  completely randomizes the events over all the links, while conserving  $\mathcal{L}$  and thus  $G^{\text{stat}}$ . It was called *random(ized) contacts* in Refs. [20, 53]. 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 instantaneous events on each individual timeline, i.e. it constrains only  $\mathbf{w}$ .  $P[\mathbf{w}]$  was called *random time(s)* in Refs. [20, 21, 45, 53], *uniformly random times* in Ref. [31], *temporal mixed edges* in Ref. [52], *poisonized IEIs* in Ref. [47], and *SRan* in Ref. [30].  $P[\mathbf{w}]$  was also employed in Refs. [44, 74].

$P[\mathbf{w}, \mathbf{t}^1, \mathbf{t}^w]$  redistributes instantaneous events inside each timeline at random (constraining  $\mathbf{w}$ ) while constraining the times of the first and last events constrained,  $\mathbf{t}^1$ , and  $\mathbf{t}^w$ , respectively. It is a maximum entropy version of one of the *poor man's methods* introduced in [35] (see Section V K 2).

$P[\pi_{\mathcal{L}}(\Delta\tau)]$  shuffles the instantaneous events while conserving the distribution of inter-event durations on each link,  $\pi_{\mathcal{L}}(\Delta\tau)$ . It is the instant-event shuffling equivalent of  $P[\pi_{\mathcal{L}}(\tau), \pi_{\mathcal{L}}(\Delta\tau)]$  defined below.

$P[\pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^1]$  shuffles the inter-event durations between the instantaneous events on each link while keeping the times of the first event on each link fixed.  $P[\pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^1]$  was named *shuffled IEIs* in Refs. [47, 48].

## 2. Event shufflings

$P[\mathcal{L}, p(\boldsymbol{\tau})]$  constrains the static network structure  $G^{\text{stat}}$  and otherwise shuffles the events completely at random between all timelines.  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  is the coarsest timeline event shuffling, and is the event shuffling equivalent of the instant-event shuffling  $P[\mathcal{L}]$ . We may consequently refer to  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  as the *timeline (event) shuffling*.

$P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$  redistributes the events uniformly inside each timeline. The inter-event durations are randomized and asymptotically follow exponential distributions.  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$  is an event-shuffling version of the instant-event shuffling  $P[\mathbf{w}]$ .

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

$P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$  shuffles 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. [32].

$P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}), \mathbf{t}^1]$  adds another constraint to  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$  so that it conserves the first event time on each link (it is an 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 timelines on each link individually using periodic boundary conditions.  $P[\mathbf{per}(\boldsymbol{\Theta})]$  was named *random offset* in Ref. [37].

$P[\boldsymbol{\tau}, \boldsymbol{\Delta\tau}]$  is a refinement of  $P[\mathbf{per}(\boldsymbol{\Theta})]$  that imposes hard boundary conditions instead of periodic ones.

## F. Sequence shufflings

Sequence shufflings (Def. IV.5) are a particular kind of instant-event timeline shufflings. They randomize the sequence of snapshots while leaving the individual snapshots unchanged. 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  $G^{\text{stat}}$ , as well as all instantaneous topological correlations inside snapshots (see Fig. 11). This in particular means that sequence shufflings are also timeline shufflings (see Def. IV.5), and are thus compatible with both snapshot shufflings (Subsec. V G) and timeline shufflings (Subsec. IV.4).

We have identified the following two sequence shufflings in the literature. These are included in the Hasse diagram shown in Fig. 13.

## 1. Instant-event shufflings

$P[p_{\mathcal{T}}(\boldsymbol{\Gamma})]$  randomly shuffles the timestamps of the snapshots.  $P[p_{\mathcal{T}}(\boldsymbol{\Gamma})]$  was named *reshuffled sequences* in Ref. [50]; it appears in Refs. [38], in Ref. [52] as *random ordered*, in Ref. [45] as *shuffled times*, and in Ref. [43] as *reshuffle*. Since it is the coarsest possible sequence shuffling, we simply name it *sequence shuffling*.

$P[p_{\mathcal{T}}(\boldsymbol{\Gamma}), \mathbf{sgn}(\mathbf{A})]$  shuffles the timestamps of the snapshots where at least one event takes place, i.e. where  $\mathbf{sgn}(A^t) = 1$ . It thus constrains the function  $\mathbf{sgn}(\mathbf{A}) = (\mathbf{sgn}(A^t))_{t \in \mathcal{T}}$ . It was employed in Ref. [45] with the name of *shuffled times*.

## G. Snapshot shufflings

Snapshot shufflings (Def. IV.6) conserve the start times  $t$  of all events. They are typically implemented by randomizing the instantaneous snapshot graphs  $\Gamma^t$  corresponding to each time  $t \in \mathcal{T}$ . As a consequence, all snapshot shufflings found in the literature are instant-event shufflings, but they may also be implemented as event shufflings—we give one example in this section and two others in Section V H. 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 (see Fig. 11). Snapshot shufflings are compatible with sequence shufflings (Subsec. V F).

The snapshot shufflings listed below are ordered hierarchically in the Hasse diagram shown in Fig. 14.

### 1. Instant-event shufflings

$P[\mathbf{t}]$  randomly shuffles the instantaneous events inside each snapshot. This is equivalent to generating each snapshot  $\Gamma^t$  as 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. [45].

$P[\mathbf{t}, \Phi]$  shuffles the events inside each snapshot while additionally constraining the set of nodes that are active at each  $t$ , i.e. it constrains  $\Phi$ .  $P[\mathbf{t}, \Phi]$  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. [52] and *degree preserved network* in Ref. [45], and was also applied in Ref. [55].

$P[\mathbf{iso}(\Gamma)]$  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  $\Gamma^t$  of the original network,  $(\Gamma^t)' \simeq \Gamma^t$ , i.e. the shuffling constrains the isomorphism class of all snapshot graphs,  $\mathbf{iso}(\Gamma) = (\mathbf{iso}(\Gamma^t))_{m=1}^T$ .  $P[\mathbf{iso}(\Gamma)]$  was named *anonymize* in Ref. [43].

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

## 2. Event shufflings

$P[p(\mathbf{t}, \boldsymbol{\tau})]$  randomly shuffles the events between links in a temporal network while keeping the time at which each event occurs and its duration constrained.  $P[p(\mathbf{t}, \boldsymbol{\tau})]$  is an event-shuffling version of the instant-event shuffling  $P[\mathbf{t}]$  defined above.

## H. Intersections of two shufflings

Several shuffling methods constrain features corresponding to two of the classes above and can thus be classified as intersections of these. We have namely found shuffling methods in the literature that are intersections of timeline and snapshot shufflings and of link and timeline 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. VD) and sequence shufflings (Subsec. VF). The shufflings are included in the Hasse diagrams in Figs. 13 and 14.

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

$P[\mathbf{w}, \mathbf{t}]$  randomly shuffles 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  $p(\mathbf{t}) = \{t_q\}_{q=1}^Q$ . In a completely equivalent manner, we may define the shuffling by constraining the timestamps  $\mathbf{t}$  and permuting the pairs  $(i_q, j_q)$ , i.e. constraining  $\mathbf{w}$ . Due to the indistinguishability of networks obtained through permutation of event indices, both are equivalent to conserving  $\mathbf{w}$  and  $\mathbf{A}$ . For convenience, we choose the canonical name  $P[\mathbf{w}, \mathbf{t}]$  which conveys that it is both a timeline shuffling and a snapshot shuffling.

$P[\mathbf{w}, \mathbf{t}]$  is a very popular MRRM. It was named *permuted times* in Ref. [53], and called *time-shuffled* or *time-shuffling* in Refs. [22, 25, 27, 31, 39], *randomly permuted times* in Refs. [20, 34, 46, 49], *random dynamic* in Ref. [28], *random time shuffle* in Ref. [37], *reconfigure* in Ref. [43], and *shuffled time stamps* in Ref. [21]. It was also employed in Refs. [29, 33, 48, 54].

## 2. Event shufflings

The two following event shufflings are intersections of timeline and snapshot shufflings. They are thus compatible with both link shufflings (Subsec. VD) and sequence shufflings (Subsec. VF). The shufflings are included in the Hasse diagrams in Figs. 13 and 14.

$P[\mathcal{L}, p(\mathbf{t}, \boldsymbol{\tau})]$  randomly shuffles the events while constraining their starting times as well as  $G^{\text{stat}}$ . It can be seen as a refinement of  $P[\mathcal{L}, \mathbf{t}]$  defined above 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 above. It conserves the distribution of event durations, their starting times, and the links and their event frequencies  $\mathbf{n}$ , but not necessarily their weights  $\mathbf{w}$ .

The three following event shufflings are intersections between link shufflings and timeline shufflings. They are thus compatible with link shufflings (Subsec. VD), as well as timeline (Subsec. VE) and sequence shufflings (Subsec. VF). The shufflings are included in the Hasse diagrams in Figs. 12 and 13.

$P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  randomly shuffles the timelines between all links while keeping the static network  $G^{\text{stat}}$  fixed.  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  was named *link-sequence shuffled* in Refs. [27, 31, 39], *edge randomization* in Ref. [20], and *link shuffling* in Refs. [32, 42].

$P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$  shuffles timelines  $\Theta_{(i,j)}$  between links with the same weight  $w_{(i,j)}$ .  $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$  was named *equal-weight link-sequence shuffled* in Refs. [27, 31, 39] and was also called *equal-weight edge randomization (EWER)* in Ref. [20].

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

## I. 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 II C). For example, complete randomization of an instant-event temporal network, while keeping the number of links fixed, may be obtained by randomly permuting the links between all pairs of nodes using the link shuffling  $P[p_{\mathcal{L}}(\Theta)]$  and randomly permuting the instantaneous events on and between the links using the timeline shuffling  $P[\mathcal{L}]$ . The resulting model is  $P[L]$  (Proposition IV.1). Since compatible MRRMs commute, it does not matter in which order we apply them (Proposition II.2).

We list here only examples we have found in the literature, all of which are compositions of link and timeline shufflings. The number of different MRRMs that may be generated by composition is given by the number of combinations of compatible shufflings, and is thus much larger than this (see discussion at the start of this section).

### 1. Instant-event shufflings

$P[L]$  is generated by the composition of  $P[p_{\mathcal{L}}(\Theta)]$  and  $P[\mathcal{L}]$ . It was called *all random* in Ref. [53].

$P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}]$  applies  $P[p_{\mathcal{L}}(\Theta), \mathbf{k}]$  and  $P[\mathbf{w}, \mathbf{t}]$  in composition. It was called *randomized edges with randomly permuted times* in Refs. [20, 46].

$P[\mathbf{k}, \mathbb{I}_{\lambda}, p(\mathbf{w}), \mathbf{t}]$  adds the additional constraint to  $P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}]$  that the static graph must be connected. It is the composition of  $P[p_{\mathcal{L}}(\Theta), \mathbb{I}_{\lambda}, \mathbf{k}]$  and  $P[\mathbf{w}, \mathbf{t}]$  and was called *configuration model* in Refs. [27, 39].

## J. Randomization based on metadata

The availability of metadata offers the possibility to impose additional external constraints in the MRRMs. This allows studying effects that are not purely due to the structure and dynamics of the network. For instance, in Ref. [25], the age, gender, and type of subscription of mobile phone users were known; in Ref. [28], the authors used shuffling methods respecting the bipartite structure of a sex worker-buyer interaction network, and Ref. [42] used a shuffling that rewired links between each pair of predefined node groups in face-to-face networks.

These metadata MRRMs are all a type of stochastic blockmodel [75]. They may be defined by assigning a *color* to each node, i.e. to which group it belongs among a set of  $R$  predefined groups. The node colors are fixed by the vector  $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_N)$ , where  $\sigma_i \in \{1, 2, \dots, R\}$ . An  $R \times R$  *group contact matrix*,  $\Sigma_{\mathcal{L}}$  [with elements given by the number of links between groups,  $(\Sigma_{\mathcal{L}})_{\sigma\sigma'} = \sum_{(i,j) \in \mathcal{L}} (\delta_{\sigma_i, \sigma} \delta_{\sigma_j, \sigma'} + \delta_{\sigma_j, \sigma} \delta_{\sigma_i, \sigma'})$ ], typically fixes the number of links between members of each group [we may alternatively fix the number of events instead using a matrix  $\Sigma_{\mathcal{L}}$ , with elements given by

$(\Sigma_{\mathcal{L}})_{\sigma\sigma'} = \sum_{(i,j,t,\tau) \in \mathcal{C}} (\delta_{\sigma_i, \sigma} \delta_{\sigma_j, \sigma'} + \delta_{\sigma_j, \sigma} \delta_{\sigma_i, \sigma'})$ ]. These two additional constraints enables us to define MRRMs that impose structure or dynamics determined by the metadata.

We may also directly use this blockmodel MRRM construction to conserve the bipartite structure of a network as in [28] 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 finally allow both  $\sigma$  and  $\Sigma$  to vary over time in order to capture temporal changes in the group structure.

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

$P[p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$  shuffles the links in the static graph while constraining the group appartenance of each node,  $\sigma$ , and the number of links between each group,  $\Sigma_{\mathcal{L}}$ . It was employed in Ref. [42], where it was called *CM-shuffling*.

$P[\mathbf{k}, p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$  randomizes  $G^{\text{stat}}$  while constraining the group structure, as  $P[p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$  does, while additionally constraining the node degrees  $\mathbf{k}$ .  $P[p_{\mathcal{L}}(\Theta), \mathbf{k}, \sigma, \Sigma_{\mathcal{L}}]$  was used to constrain the bipartite structure of the network and was named *random topological* in Ref. [28].

$P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}, \sigma, \Sigma_{\mathcal{L}}]$  is generated by composition of the metadata reliant model  $P[p_{\mathcal{L}}(\Theta), \mathbf{k}, \sigma, \Sigma_{\mathcal{L}}]$  with  $P[\mathbf{w}, \mathbf{t}]$ . It was named *random dynamic topological* in Ref. [28].

$P[G, p(\sigma)]$  shuffles the group affiliations (colors) of the nodes at random (i.e. it randomizes the order of  $\sigma$ ). 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  $P[\text{iso}(G^{\text{stat}}), p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$ , but we use the above name for conciseness. It was employed in Ref. [25], where it was called *node type shuffled data*.

## K. Other reference models

We have above restricted ourselves to microcanonical RRRMs 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 RRRMs 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* RRRMs, 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,

TABLE IV: **Effects of metadata-dependent shufflings on features of temporal networks.** Special metadata symbols are the color (group affiliation) of a node,  $\sigma_i$ , and the group contact matrices  $\Sigma_{\mathcal{L}}$  and  $\Sigma_{\mathcal{E}}$  (see main text for definitions and Tables I and II for other symbols). Note that a feature that is not conserved (–) by a randomization procedure is not necessarily completely randomized either (see discussion at the end of Sec. III).

Canonical name	Meta			One-level							Two-level							
	$\sigma_i$	$\Sigma_{\mathcal{L}}$	$\Sigma_{\mathcal{E}}$	topological	weighted			temp.				node			link			
				$G^{\text{stat}}$	$k_i$	$L$	$a_i^\dagger$	$s_i$	$n_{(i,j)}^\dagger$	$w_{(i,j)}$	$A^t$	$\alpha_i^{m\dagger}$	$\Delta\alpha_i^m$	$d_i^t$	$\tau_{(i,j)}^m$	$\Delta\tau_{(i,j)}^m$	$t_{(i,j)}^1$	$t_{(i,j)}^w$
$P[p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$	x	x	–	–	$\mu$	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu\tau$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
$P[\mathbf{k}, p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$	x	x	–	–	x	x	$\mu$	$\mu$	$p$	$p$	x	–	–	$\mu\tau$	$p_{\mathcal{L}}$	$p_{\mathcal{L}}$	$p$	$p$
$P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}, \sigma, \Sigma_{\mathcal{L}}]$	x	x	–	–	x	x	–	$\mu$	–	$p$	x	–	–	$\mu\tau$	–	–	–	–
$P[G, p(\sigma)]$	$p$	x	–	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

<sup>†</sup> Feature only defined for temporal networks with event durations.

but important, type of reference models that do not maximize entropy.

### 1. Canonical randomized reference models

Canonical RRM present alternatives that are very close in spirit to the microcanonical RRM 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) [2, 76] and allow to model the expected variability between samples (see discussion in [77, Section 4]). They are thus expected to have a lower generalization error than microcanonical RRM. Additionally, due their soft constraints, canonical models are typically more amenable to analytical treatment than their microcanonical counterparts [78].

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, which explains their popularity for analyzing temporal networks. In particular, it is easy to generate microcanonical RRM that impose 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 generative reference models that capture empirical temporal correlations, these are currently almost exclusively defined for static networks or to model either memoryless dynamics [27, 40, 47, 79] or dynamics with limited temporal correlations [80–84]. A notable exception is a recent study combining Markov chains with change point detection to model multiscale temporal dynamics [85]. We shall not discuss canonical

RRMs in more detail here, but refer to [76] for a recent review of ERGMs for temporal networks and to [77, 86] 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 [61]. 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 features of the temporal network. Specifically they constrained all instances of this feature 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. [30] imposes a fixed number of events in each snapshot (equal to the mean number of events per snapshot in the empirical network). Holme [35] 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 [21]): equalizing the inter-event durations  $\Delta\tau_{(i,j)}^m$  while constraining  $t_{(i,j)}^1$ ,  $t_{(i,j)}^w$  and  $w_{(i,j)}$  for each link  $(i, j) \in \mathcal{L}$  (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  $t_{(i,j)}^1 = t_{\min}$  or to make  $t_{(i,j)}^w = t_{\max}$  for all  $(i, j) \in \mathcal{L}$ .

*Biased sampling.* Kovanen et al. [22] proposed a biased version of  $P[\mathbf{w}, \mathbf{t}]$ , where instead of swapping timestamps of events at random, for each instantaneous event  $(i, j, t)$  they drew  $m$  other events at random from the set of instantaneous 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. [25, 56]. Valdano et al. [43] considered a heuristic variant of  $P[p_{\mathcal{T}}(\mathbf{g})]$  (called *reshuffle-social*, where they only permuted snapshots inside intervals where nodes showed approximately the same median *social strategy* [87], 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  $\gamma_i^t$  separated by low- $\gamma_i^t$  intervals, referred to as  $\gamma$ -slices, which allowed them to permute snapshots within each  $\gamma$ -slice only.

*Time reversal.* A quick but informal to gain insight into the role of causality in the contact dynamics is to reverse the order of the snapshots [20–22, 46]. This method obviously does not increase entropy 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 returns an input temporal network and its time-reversed version with equal probability.

### 3. Bootstrap methods

Bootstrap methods are based on sampling with replacement, whereas MRRMs are based on sampling without replacement (i.e. shuffling). Resampling with replacement means that network features are not constrained exactly as for shuffling methods. The hope when using bootstrapping can thus be to capture some of the out-of-sample variability. The set of states that may be generated is strongly constrained by the particular dataset however, 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 and canonical RRM do. So the nice theoretical results and guarantees that exist for microcanonical RRM (see Sec. II) 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. [32] constrains the number of events per link  $\mathbf{n}$  exactly and resamples the event durations  $\tau$  from the global distribution  $p(\tau)$  with replacement. The method called *time shuffling* in Ref. [42] constrains the static network  $G^{\text{stat}}$  and bootstraps  $n_{(i,j)}$ ,  $t_{(i,j)}^1$  for all links from the global distributions  $p(\mathbf{n})$  and  $p(\mathbf{t}^1)$ , respectively, and then bootstraps the  $n_{(i,j)}$  event durations  $\tau_{(i,j)}^m$  and  $(n_{(i,j)} - 1)$  of inter-event durations  $\Delta\tau_{(i,j)}^m$  for each link  $(i, j) \in \mathcal{L}$  from the global distributions  $p(\tau)$  and  $p(\Delta\tau)$ , respec-

tively.

## VI. 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 two walk-through examples of the use of nested series of MRRMs to analyze empirical temporal networks: In Subsection VI A, we analyze the features in a dataset of face-to-face interactions in a primary school; in Subsection VI B, we explore the effects of different features of a mobile phone call network on the temporal distances between nodes, which summarize how fast a dynamical process may spread in the network.

MRRMs permit us to perform null-model based analysis and 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?” [27]. Alternatively: “Is the original network significantly different from a random network with the same link weights and overall activity patterns?” [25].

Practically speaking, statistical testing using a given MRRM,  $P[\mathbf{x}]$ , 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 features or the frequencies of given network sub-graphs [16, 25], 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 [32].
2. Apply a shuffling method corresponding to  $P[\mathbf{x}]$ , 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  $\mathbf{y}$  under the model  $P[\mathbf{x}]$ .
3. Compare the values of  $\mathbf{y}$  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 feature  $\mathbf{x}$  is not enough to explain what we see in the original network. Conversely, if the two do agree, we may conclude that

$\mathbf{x}$  can alone explain what we see in the original network.

The above procedure may also be used to compare two ensembles of randomized networks created using 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 individual and collective effects of a range of network features.

One is often more interested in computing and comparing effect sizes (e.g. how much faster/slower does a contagion process spread in the randomized networks?) and qualitative comparisons (e.g. does the distribution of inter-event durations have a broad tail or not?), rather than performing categorical hypothesis testing (i.e. using the  $p$ -value). Thus, we do not necessarily 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. Walkthrough example: analyzing face-to-face interactions using MRRMs

In this subsection we apply a selection of MRRMs to illustrate how they can be used to methodically explore different features and build a statistical portrait of a temporal network. We analyze a SocioPatterns dataset of face-to-face interactions recorded in a primary school [88, 89], which is freely available at [www.sociopatterns.org/datasets](http://www.sociopatterns.org/datasets). The data were recorded with a time-resolution of 20 s and form a temporal network consisting of 242 nodes and 77 521 events of varying duration. The applied MRRMs are listed hierarchically in Fig. 15.

Figure 16 illustrates our statistical portrait of the temporal network. It quantifies how each MRRM changes a selection of temporal network features. Namely, the timeline of cumulative node activity,  $\mathbf{A}$ ; the distributions of five time-aggregated (one-level) features: the node degrees,  $p(\mathbf{k})$ , the link weights and event frequencies,  $p(\mathbf{w})$  and  $p(\mathbf{n})$ , and the node strengths and activities,  $p(\mathbf{s})$  and  $p(\mathbf{a})$ ; as well as global distributions of four temporal-structural (two-level) features: the event and inter-event durations on links,  $p(\boldsymbol{\tau})$  and  $p(\boldsymbol{\Delta\tau})$ , and the node activity and inactivity durations,  $p(\boldsymbol{\alpha})$  and  $p(\boldsymbol{\Delta\alpha})$ . The differences are quantified by the Jensen-Shannon divergence (JSD) [90] 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 in the empirical network and for each MRRM are shown in Supplementary Figs. 3 and 4.

We first study the activity timeline  $\mathbf{A}$ . It is by construction completely constrained by all link shufflings and snapshot shufflings, while at the opposite end it is essentially completely randomized by  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$ ,

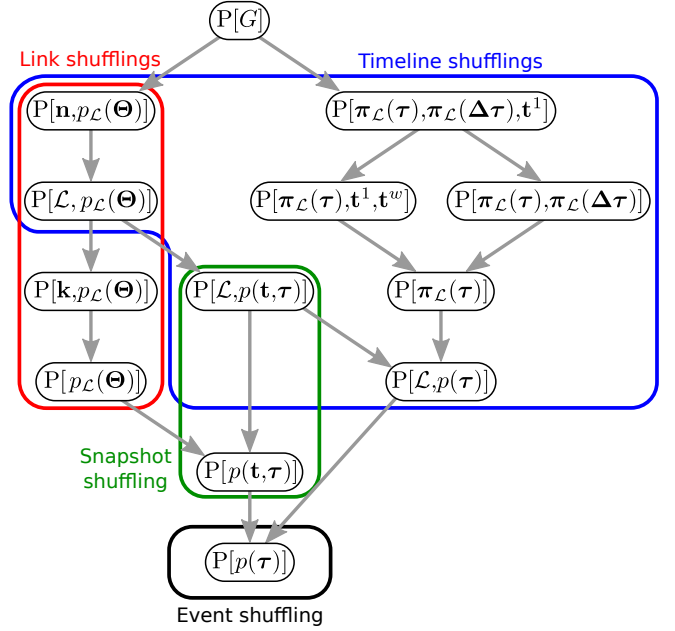


FIG. 15: MRRMs employed in the analysis of face-to-face interactions in a primary school.

$P[p(\boldsymbol{\tau})]$ , described in detail in Section V C, is the coarsest (most random) event shuffling possible. See Section V D for definitions of link shufflings, Section V E for timeline shufflings, Section V G for the snapshot shuffling  $P[p(t, \boldsymbol{\tau})]$ , and Section V H for the intersections  $P[\mathbf{n}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$ ,  $P[\mathcal{L}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$ , and  $P[\mathcal{L}, p(t, \boldsymbol{\tau})]$ .

$P[\mathcal{L}, p(\boldsymbol{\tau})]$ , and  $P[p(\boldsymbol{\tau})]$  (see Supplementary Fig. 4). This shows that  $\mathbf{A}$  is not constrained by the static graph of the network. Comparison between  $P[p(\boldsymbol{\tau})]$  and  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau})]$  shows that the distribution of inter-event durations does affect  $\mathbf{A}$ , but not to a large extent. Comparing this with  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$  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. Furthermore, comparison with  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\Delta\tau}), \mathbf{t}^1]$  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 (see also Supplementary Fig. 4).

We next consider time-aggregated features of the temporal network, starting with the distribution of node degrees,  $p(\mathbf{k})$ . This feature is constrained by most of the MRRMs applied, with the exception of  $P[p_{\mathcal{L}}(\boldsymbol{\Theta})]$  (which draws  $G^{\text{stat}}$  from an Erdős-Rényi model), and  $P[p(t, \boldsymbol{\tau})]$  and  $P[p(\boldsymbol{\tau})]$  (which do not conserve the number of links in  $G^{\text{stat}}$ ). The high divergence seen for  $P[p_{\mathcal{L}}(\boldsymbol{\Theta})]$  shows that the empirical network's degree distribution is significantly nonrandom (even if it does not seem to follow a broad-tailed distribution, see Supplementary Fig. 3).

Most of the MRRMs also conserve the distributions of link weights and event frequencies,  $p(\mathbf{w})$  and  $p(\mathbf{n})$ , respectively, with the exceptions  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  (which conserves the static structure, but not the heterogeneity in

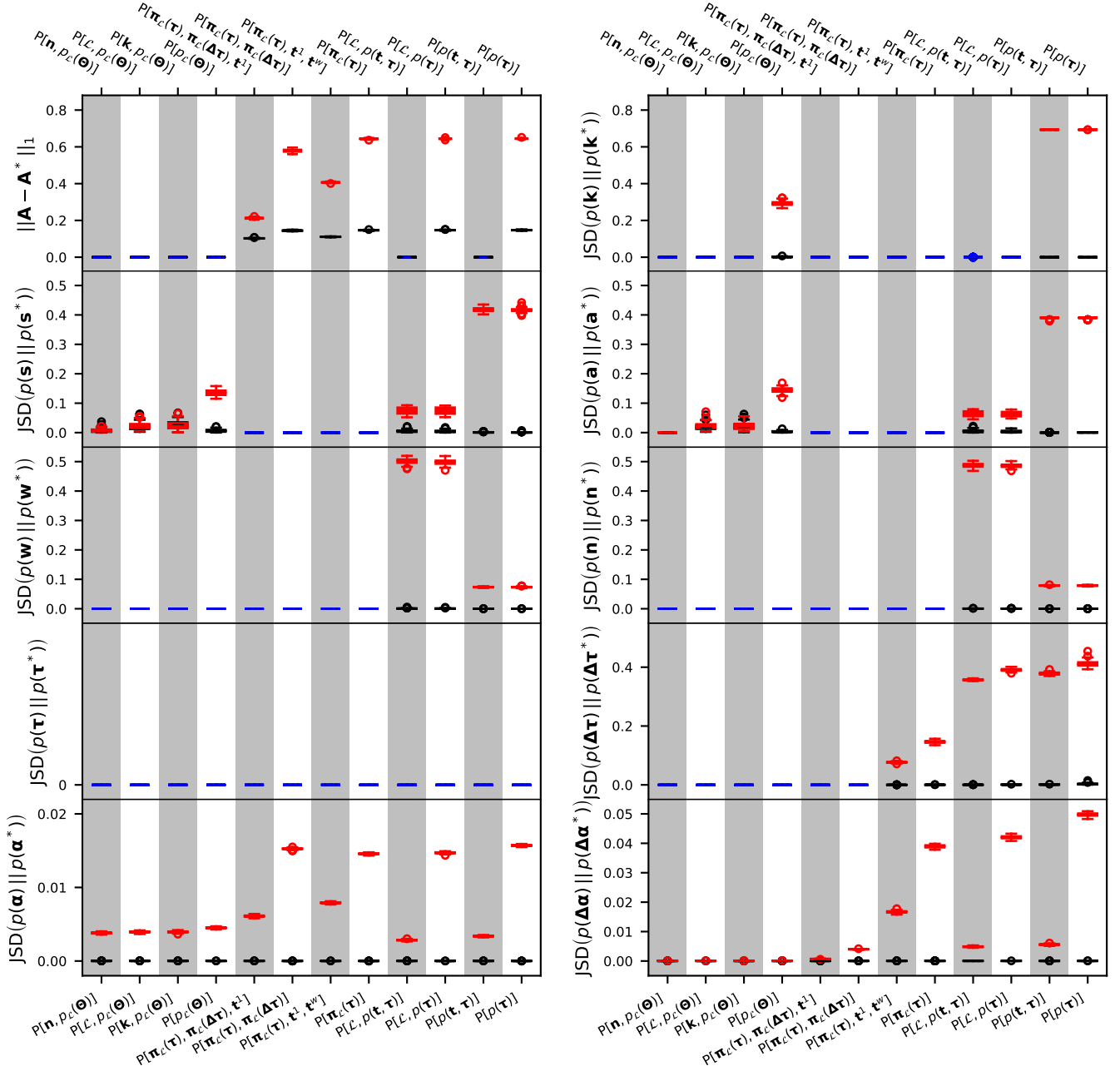


FIG. 16: **Effects of the MRRMs on different features of a temporal network of face-to-face interactions in a primary school.** Each panel shows the difference between the value of the feature in the empirical network and its null distribution under each model (red symbols for MRRMs that do not constrain the feature, and blue symbols for MRRMs that do) as well as the differences between its value in different randomized networks in each null ensemble (black). The latter serves as a benchmark that shows the expected difference due to random fluctuations if a null model were true. 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 (JSD) between the global distributions of the values of individual scalar features. Each box-and-whiskers summarizes the distribution of the differences over 100 randomized networks generated by the MRRM in question: boxes show the 1st and 3rd quartiles; whiskers extend to 1.5 times the interquartile range or to the minimum (maximum) value, whichever is smaller; values beyond the whiskers are marked by open circles.

the number and durations of events in timelines), and  $P[p(\mathbf{t}, \boldsymbol{\tau})]$  and  $P[p(\boldsymbol{\tau})]$  (which do not conserve the number of links in  $G^{\text{stat}}$ ). The effects of these shufflings on  $p(\mathbf{w})$  and  $p(\mathbf{n})$  are very similar, highlighting the fact that  $\mathbf{w}$  and  $\mathbf{n}$  are highly correlated features. Note that the smaller divergences seen for the more random  $P[p(\boldsymbol{\tau})]$  than for  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  are due to the JSD putting most weight on low values of  $w_{(i,j)}$  and  $n_{(i,j)}$  since these are most probable. Since  $P[p(\boldsymbol{\tau})]$  produces a much larger number of links than in the original network but conserves the number of events, it also produces a large fraction of links with low  $w_{(i,j)}$  and  $n_{(i,j)}$ , similarly to the original network. Conversely,  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  conserves the number of links and homogenizes  $w_{(i,j)}$  and  $n_{(i,j)}$ , leading to fewer low values of these.

The majority of the shufflings do not constrain the distributions of node strengths and activities,  $p(\mathbf{s})$  and  $p(\mathbf{a})$ , but as for  $p(\mathbf{w})$  and  $p(\mathbf{n})$ , their effects on the two features are very similar. Due to this we take  $p(\mathbf{a})$  as example and note that the results are similar for  $p(\mathbf{s})$ . The distribution of  $a_i$  in the empirical network is indistinguishable from networks generated by  $P[\mathbf{k}, p_{\mathcal{L}}(\boldsymbol{\Theta})]$ . This shows that  $p(\mathbf{a})$  is simply determined by the convolution of the individual distributions of  $\mathbf{k}$  and  $\mathbf{n}$  and that correlations between the two are unimportant. Comparison with  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  and  $P[p_{\mathcal{L}}(\boldsymbol{\Theta})]$  shows that both  $p(\mathbf{n})$  (randomized by  $P[\mathcal{L}, p(\boldsymbol{\tau})]$ ) and  $p(\mathbf{k})$  (randomized by  $P[p_{\mathcal{L}}(\boldsymbol{\Theta})]$ ) are needed to reproduce the non-random shape of  $p(\mathbf{a})$  though.

We finally investigate temporal-structural features of nodes and links. We note first that the distribution of event durations,  $p(\boldsymbol{\tau})$  is conserved by all MRRMs by construction.

The distribution of inter-event durations on the links,  $p(\Delta\boldsymbol{\tau})$ , is constrained by all link shufflings, but not by most of the other shufflings. Comparison of the effects of  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$  and  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$  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 Supplementary Fig. 4). The much larger divergence found for  $P[\mathcal{L}, p(\boldsymbol{\tau})]$  highlights that the number of events  $n_{(i,j)}$  on each link strongly influences the inter-event durations.

None of the MRRMs completely constrain the distributions of the nodes activity and inactivity durations,  $p(\boldsymbol{\alpha})$  and  $p(\Delta\boldsymbol{\alpha})$ . However, all link shufflings produce null distributions that are relatively close to the empirical ones, though they are still statistically significantly different. This indicates that the temporal correlations of the individual links' activity strongly constrain the nodes' activity. More surprisingly, the small divergence observed in  $p(\boldsymbol{\alpha})$  for  $P[\mathcal{L}, p(\mathbf{t}, \boldsymbol{\tau})]$  and  $P[p(\mathbf{t}, \boldsymbol{\tau})]$  as compared to the other MRRMs points to the global timing of the events as the most important of the features in determining the node activity durations in the network. It is more important than the number of events and the distributions of inter-event durations on the links. Conversely, we see that the distribution of inter-

event durations,  $\boldsymbol{\pi}_{\mathcal{L}}(\Delta\boldsymbol{\tau})$ , is the most important temporal feature in determining the nodes' inactivity durations,  $p(\Delta\boldsymbol{\alpha})$ , while the timing of the events is a close second (compare  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\Delta\boldsymbol{\tau}), \mathbf{t}^1]$  and  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \boldsymbol{\pi}_{\mathcal{L}}(\Delta\boldsymbol{\tau})]$  to  $P[\mathcal{L}, p(\mathbf{t}, \boldsymbol{\tau})]$ , and these three to  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau}), \mathbf{t}^1, \mathbf{t}^w]$  and  $P[\boldsymbol{\pi}_{\mathcal{L}}(\boldsymbol{\tau})]$ ).

As seen in Supplementary Figs. 3 and 4, the distributions of the different features obtained from a single randomized network generally vary little around their median, even though the empirical network studied here is of relatively modest size.

## B. Walkthrough example: analyzing temporal distances in a communication network using MRRMs

In this subsection we give an illustrative example how to use the hierarchy of MRRMs to analyze how the different features of a temporal communication network affects the temporal distances between nodes in the network (defined as the minimal times required for any contagion process to spread between the nodes). This example additionally serves to showcase a graphical representation that incorporates both the hierarchy of the MRRMs and their effects on a scalar feature, and which provides an intuitive way to interpret the results (see Fig. 17). As discussed in Section VII below, understanding how different features affect spreading was the starting point of some of the early studies employing MRRMs in temporal networks, and here we reproduce some of those results with a different data set. However, the analysis pipeline introduced here does not only work for temporal distances, but can be used for any other scalar-valued feature.

The data used here is a publicly available temporal mobile phone communication network published by Wu et al. [91]. Here we focus on the first company with 44431 nodes and around  $5.5 \times 10^5$  instantaneous events taking place over 30 days. Distances in temporal networks is a multifaceted topic [92], but here we quantify the distances in a network by a single number describing the typical temporal distance in the network. More specifically, we calculate the expected temporal distance to reach half of the nodes in the network, i.e. the *expected median temporal distance*  $\langle d_{1/2}(G) \rangle$ , where the expectation is evaluated over all nodes and all times as source points. The temporal distance from one node  $i$  to another node  $j$  is defined as the time required for the fastest possible spreading process starting at a given time  $t$  to spread from  $i$  to  $j$  [93]. Formally, this fastest possible spreading is modeled by a deterministic susceptible-infectious (SI) process where susceptible nodes are infected immediately by contact with an infectious node. When evaluating the distances we use periodic boundary conditions in time to remove boundary effects [27].

Figure 17 displays  $\langle d_{1/2}(G) \rangle$  for the original data and for several MRRMs. The figure is organized in a way that the hierarchies (see Section V) are visible similar to

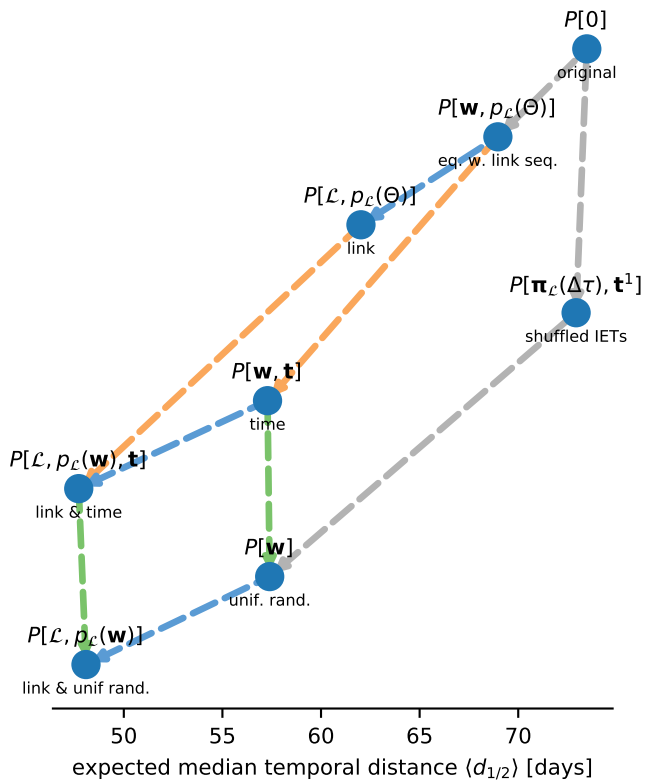


FIG. 17: **The expected median temporal distance values for a hierarchy of MRRMs.** Each circle in the figure represents a single MRRM. The horizontal location of the circle reports the expected median temporal distance  $\langle d_{1/2}(G) \rangle$  of the MRRM. An arrow from a MRRM ( $P[\mathbf{x}]$ ) at a higher location to a lower one ( $P[\mathbf{y}]$ ) means that the former shuffles less than the latter ( $P[\mathbf{x}] \leq P[\mathbf{y}]$ ). A canonical name of each MRRM is given above each circle and a common name below it (see Section V). In the common names the word *randomization* is always removed for brevity, and the &-sign denotes that both MRRMs are applied to the data in composition (Def. II.12). Colored links indicate that the same features were removed:  $\mathbf{t}$  for green links,  $\mathbf{w} \leftrightarrow \mathcal{L}, p_{\mathcal{L}}(\mathbf{w})$  for blue links, and  $p_{\mathcal{L}}(\Theta) \leftrightarrow \mathbf{t}$  for orange links.

Figures 12 – 15. Reading the figure from top to bottom now yields a picture of what happens when the original data is shuffled more and more, i.e., when the temporal features present in the data are destroyed one by one by the MRRMs. All of the arrows are pointing either almost directly downwards or down and left, which means that, for this network and these set of MRRMs, randomizing more never leads to longer temporal distances.

The overall activity sequence  $\mathbf{t}$ , including the daily and weekly changes in the activity, does not have a noticeable effect on the temporal distances on these MRRMs: Re-

moving the constraint on  $\mathbf{t}$  when going from  $P[\mathbf{w}, \mathbf{t}]$  to  $P[\mathbf{w}]$  and from  $P[\mathcal{L}, p_{\mathcal{L}}(\mathbf{w}), \mathbf{t}]$  to  $P[\mathcal{L}, p_{\mathcal{L}}(\mathbf{w})]$  almost does not change the temporal distances at all. Similarly, shuffling the inter-event times while keeping the first activation time with  $P[\pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^\dagger]$  barely changes  $\langle d_{1/2}(G) \rangle$ , showing that higher-order temporal correlations between events over the same link has a very small effect on the temporal distances of the original data.

Adding the shuffling of the weights of the network – i.e. replacing the feature that keeps the weights of the links,  $\mathbf{w}$ , with the one only keeping the links and the weight distribution,  $\mathcal{L}$  and  $p_{\mathcal{L}}(\mathbf{w})$  – makes the temporal paths around 7–9 days faster. The pairs of MRRMs corresponding to this replacement are  $P[\mathbf{w}, \mathbf{t}]$  to  $P[\mathcal{L}, p_{\mathcal{L}}(\mathbf{w}), \mathbf{t}]$ ,  $P[\mathbf{w}]$  to  $P[\mathcal{L}, p_{\mathcal{L}}(\mathbf{w})]$ , and  $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$  to  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$ . Note that in the MRRM  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  the weight distribution  $p_{\mathcal{L}}(\mathbf{w})$  is kept implicitly by the link sequence distribution  $p_{\mathcal{L}}(\Theta)$ , because  $p_{\mathcal{L}}(\Theta) \leq p_{\mathcal{L}}(\mathbf{w})$ .

Finally the largest change in the temporal distance are seen when the times of the link sequences,  $p_{\mathcal{L}}(\Theta)$ , are shuffled such that they simply follow the overall activity sequence  $\mathbf{t}$ . In these transitions, from  $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$  to  $P[\mathbf{w}, \mathbf{t}]$  and from  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  to  $P[\mathcal{L}, p_{\mathcal{L}}(\mathbf{w}), \mathbf{t}]$ , the temporal distances are reduced on average by around 12–14 days.

Almost no combination effects were observed for these data: removing each feature had a very similar effect – with variations of around 2 days – independently of the other features that were kept. This allows a very simple summarization of the results: The typical temporal distance in the data is around 73 days and in the most random MRRM applied here around 48 days. Out of that difference, around 12–14 days is explained by link activation sequence features (such as bursts [27]), 7–9 days are by weight-topology correlations (such as weak links located in bridge positions [18, 94]), and 4 days by link-sequence-topology correlations (such as correlations in times at which two neighbors of a node are communicated with [27, 31]).

This analysis can be made more detailed by adding more fine-scaled features related to timings of events or link weights. Alternatively, the analysis could be widened by including topological MRRMs such as the configuration model.

## VII. 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 features in temporal networks.

(i) Dynamical processes have been studied by using data-driven models, where temporal interactions are ob-

tained from real data, while the ongoing dynamical process is modeled by using any conventional process definition [20, 95] and typically simulated numerically on the empirical and randomized temporal networks [95, 96]. 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 features in temporal networks (often termed interaction motifs) or correlations between the network dynamics and node attributes. This approach is conceptually the same as using the configuration model to detect overrepresented subgraphs (termed *motifs*) in static networks [16, 97, 98]. The difference here is that the studied networks vary in time, which induces further challenges.

In the first three subsections of this section (Subsecs. VII A–VII C), we review studies applying MRRMs to study various dynamical processes in empirical temporal networks. In the fourth part of this section (Subsec. VII D) applications to inferring statistically significant motifs and correlations in network dynamics will be discussed briefly. Finally, in the last subsection (Subsec. VII 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. V K 2–V K 3). While they do not come with the same formal statistical guarantees as maximum entropy RRs 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 features of the different contagion processes we refer readers to the recent review by Pastor-Satorras et al. [95].

### 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  $\beta$  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  $\beta$  controls the speed of saturation, thus by considering the limit  $\beta \rightarrow \infty$  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 [53].

Early motivation to use RRs 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  $\beta \rightarrow \infty$  [27]. In this study two mobile communication networks and an email network were taken as temporal networks to study this phenomenon [99]. 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  $P[\mathbb{I}_\lambda, \mathbf{k}, p(\mathbf{w}), \mathbf{t}]$  (Sec. VI), 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  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  models (Sec. VH). This led to the conclusions that the bursty interaction dynamics and the Granovetterian weight-topology correlations [94] are dominantly responsible for the slow spreading of information in these systems. On the other hand the  $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$  model (Sec. VH), 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 canonical RRs where interaction times were generated by either a ho-

mogeneous 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 Kivela et al. [31] to study the same process on the same empirical networks. They applied  $P[\mathbf{w}]$  (Sec. V E) to randomize all temporal correlations while conserving the aggregated structure, and  $P[\mathbf{k}, \mathbb{I}_\lambda, p_{\mathcal{L}}(\Theta)]$  and  $P[\mathbb{I}_\lambda, p_{\mathcal{L}}(\Theta)]$  (Sec. V D) 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. [32] analyzed face-to-face interaction networks 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  $P[\pi_{\mathcal{L}}(\tau), \pi_{\mathcal{L}}(\Delta\tau)]$  (Sec. V E), they shuffled the timelines between existing links using  $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$  (Sec. V H), and they finally bootstrapped the global distribution of event durations  $p(\tau)$  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 infection, starting from a random seed node, 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  $\mathbf{n}$  and the synchronized contact patterns (typical in a school during breaks) also have a strong effect on the contagion dynamics.

Perotti et al. [74] studied the effect of temporal sparsity, an entropy-based measure quantifying temporal heterogeneities on the empirical scale of average inter-event durations  $\langle \Delta\tau_{(i,j)}^m \rangle$ . As a reference model the authors used  $P[\mathbf{w}]$  (Sec. V E). 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. [28], which recorded the interaction events of sex sellers and buyers in Brazil. The system is a temporal bipartite network where connections 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 (obtained using the metadata-dependent blockmodel MRRMs discussed in Sec. V J with two groups and a perfectly anti-diagonal contact matrix between the groups. Their first model,  $P[\mathbf{k}, p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$ , 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,  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H), destroyed all the temporal structure except global activity patterns, but kept the weighted (bipartite) network structure unchanged. Their third model,  $P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}, \sigma, \Sigma_{\mathcal{L}}]$  (Sec. V J) was generated as the composition of the two others. Interestingly, they observed that bursty patterns accelerate the spreading dynamics, contrary to other studies [27, 29, 31, 32, 74]. 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 VII 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 [95]. Note that the accelerating effect of burstiness in this case was explained later by the non-stationarity of the temporal network [35, 44]

## 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  $S \rightarrow I$  transition of the SI process, in the SIR (SIS) process infected nodes transition spontaneously to a recovered, R (susceptible, S), state with rate  $\gamma$  (or after a fixed time  $\theta$ ), 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. [29] and investigated mobile phone communication networks. In their model they used  $\beta$  as the control parameter while letting the recovery time  $\theta$  for each node be constant. They used two reference models. The first was the  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H) model, used to study the effects of bursty interaction dynamics on global information spreading. 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 ( $i \rightarrow j$ ) and its preceding one ( $* \rightarrow i$ ) reaching the node  $i$ . To eliminate local causal correlations between the two events they randomized relay times by selecting randomly a time for the ( $* \rightarrow i$ ) event from the times of any event observed in the dataset. Both reference models preserve the link weights  $\mathbf{w}$ , 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  $\beta$ . More precisely, if  $\beta$  is small the spreading is faster and reach a larger fraction of nodes in the original temporal network than in the  $P[\mathbf{w}, \mathbf{t}]$  model. While on the contrary, if  $\beta$  is large the process evolves slower and unfolds in smaller cascades on the original data relative to the  $P[\mathbf{w}, \mathbf{t}]$  model. If  $\beta$  is large, the information propagates easily but its is affected strongly by large inter-event durations and local correlations, while if  $\beta$  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 *dynam-*

*ical 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 [42], 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  $n_{(i,j)}$ ,  $\tau_{(i,j)}^m$ , and  $\Delta\tau_{(i,j)}^m$ . 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:  $P[\pi_{\mathcal{L}}(\boldsymbol{\tau}), \pi_{\mathcal{L}}(\Delta\boldsymbol{\tau})]$ ; the metadata-dependent shuffling  $P[p_{\mathcal{L}}(\boldsymbol{\Theta}), \boldsymbol{\sigma}, \Sigma_{\mathcal{L}}]$ ; a bootstrap method, resampling  $p(\mathbf{n})$ ,  $p(\boldsymbol{\tau})$ ,  $p(\Delta\boldsymbol{\tau})$ , and  $p(\mathbf{t}^1)$ ; and finally applied  $P[p_{\mathcal{L}}(\boldsymbol{\Theta}), \boldsymbol{\sigma}, \Sigma_{\mathcal{L}}]$  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 [35] 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* [21] (see Sec. V K 2). Their first reference model conserved  $\mathbf{t}^1$ ,  $\mathbf{t}^w$ , and  $\mathbf{w}$ , 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  $t_{(i,j)}^1 = t_{\min}$  or  $t_{(i,j)}^w = t_{\max}$  for all  $(i, j) \in \mathcal{L}$ , respectively, while keeping the original sequence  $\Delta\boldsymbol{\tau}$  of inter-event durations 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. [43] 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 were:  $P[p_{\mathcal{T}}(\Gamma)]$  (Sec. V F),  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H), and  $P[\mathbf{iso}(\Gamma)]$  (Sec. V G). 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 (i) were or (ii) were not 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. V K 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 [36]. 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 snapshot shuffling,  $P[\mathbf{d}_{\rightarrow}]$ , which constrained the instantaneous out-degree  $d_{i\rightarrow}^m$  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 [100] 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  $\phi_i$  defining the necessary number  $k_i^I$  or fraction  $k_i^I/k_i$  of adopter neighbors to make the node (with total 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 [100], 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 RRRMs.

Karimi and Holme [33] 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 [100], where the numbers of adopted neighbors  $k_i^I(\delta)$  and all neighbors  $k_i(\delta)$  were calculated over a given *memory* time window  $\delta$ . 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  $\phi$  (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  $\phi$  and with the time window length, while for the absolute threshold model it increased with longer time windows. They furthermore employed two MRRMs:  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H) and  $P[p_{\mathcal{L}}(\Theta)]$  (Sec. V D). 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. [37] 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. [33]), 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 denominator, and not the degree  $k_i(\delta)$ , aggregated over the time window  $\delta$  only as in [33]. They applied two MRRMs to four different temporal interaction datasets. They used the  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H) model to destroy all temporal correlations while keeping circadian fluctuations, and introduced another model,  $P[\mathbf{per}(\Theta)]$ , 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  $P[\mathbf{w}, \mathbf{t}]$ . 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  $P[\mathbf{per}(\Theta)]$  (Sec. V E), 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. [34], where the authors used a threshold model denoted *history dependent contagion*. This model is an extension of an SI process with a threshold mechanism. 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 inter-

action 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  $P[\mathbf{w}, \mathbf{t}]$  model (Sec. V H), which lead to a slower spreading dynamics. From this they argued that burstiness increases 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 chooses 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. [30] 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 [92]. Their second model,  $P[\mathbf{w}]$  (Sec. V E), kept all weighted features 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 V K 2), randomly drawing the pairs of nodes that interact in order to approximately conserve  $\mathbf{n}$  and finally bootstrap the event durations from  $p(\tau)$ . This approximately conserves certain important statistical properties of the empirical event sequence, namely  $p(\mathbf{n})$  and  $p(\tau)$ , but not  $\mathbf{A}$  and  $p(\Delta\tau)$ . 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 heterogene-

ity 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. [40] 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 [101]) and the decay rate of its tail. They demonstrated their methodology on six empirical temporal interaction networks and used two RRM. The link shuffling  $P[\mathbf{k}, p_{\mathcal{L}}(\Theta)]$  (Sec. VD) 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  $G^{\text{stat}}$  and the mean number of events  $\langle E \rangle$  (see Sec. VK1), 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. [41]. 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 the  $P[\mathbf{w}, \mathbf{t}]$  model (Sec. VH) and measuring the coverage of the walker after a fixed number of moves. They found that after removing temporal correlations using  $P[\mathbf{w}, \mathbf{t}]$ , 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 en-

tropy 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 [102] 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 maximize their benefits.

Cardillo et al. [38] 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 sequence shuffling  $P[p_{\mathcal{T}}(\Gamma)]$  (Sec. VF), and the activity-driven model [79]. As a parameter to control the time-scale of the network, they varied the size of the integration time window 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 RRM 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. [23], 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,  $P[\mathbf{s}_{\rightarrow}, p(\Delta\tau)]$  (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. [24] 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,  $P[\mathbf{w}, p(\Delta\tau)]$  (see Appendix A), 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 are shuffled, potential correlations that induce bursty trains on single links are eliminated. Results showed 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. [22]. 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 RRM to quantify the significance of different temporal motifs. They used  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H), and they introduced a non-maximum-entropy reference model which biases the sampling of the ensemble of temporal networks defined by  $P[\mathbf{w}, \mathbf{t}]$  in order to keep some temporal correlations in the sequence (see Sec. V K 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  $P[\mathbf{w}, \mathbf{t}]$ , 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. V K 2). They used these reference models in the same spirit as the configuration model is used to identify motifs in static networks [97, 98]. Here, applying  $P[\mathbf{w}, \mathbf{t}]$  and its biased version as null models to detect 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 than non-causal ones.

In another study by the same authors [25], the same methodology was used to identify motifs in temporal networks where nodes (individuals) were assigned with metadata attributes like gender, age, and mobile subscription types. Beyond the  $P[\mathbf{w}, \mathbf{t}]$  model (Sec. V H), the authors introduced the metadata MRRM  $P[G, p(\sigma), \Sigma_{\mathcal{L}}]$  (Sec. V J), which shuffles single attributes between nodes. In addition, they applied the biased version of  $P[\mathbf{w}, \mathbf{t}]$  introduced in [22] (see Sec. V K 2), which accounts for the frequency of motif emergence in the corresponding static weighted network without considering node attributes. Using this non-maximum-entropy reference model and the two MRRMs they found gender-related differences in communication patterns and showed the existence of temporal homophily, i.e. 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. [103], where they used a large evolving online social network to analyze 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 directed-network MRRM  $P[\mathbf{s}_{\rightarrow}, p(\Delta\tau)]$  (see Appendix A), 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 links added within a given month). They showed that bursty periods of link creation are likely to appear shortly after the creation of a user account, or when a user actively use free or paid services provided by the online social service. In order to verify these correlations they used a reference model 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. [104] to demonstrate whether the effect of social influence or homophily is dominating during the adoption dynam-

ics 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. V J – coupled with a dynamical process on the network); we include it in this survey to demonstrate the scope of maximum entropy shuffling methods beyond randomizing network features solely. In case of real adoption cascades, these two mechanisms may lead to similar collective adoption patterns 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 [100, 104]) 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 predominantly 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 [105]. 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 [106]. 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 [106]. An algorithm to approximate the minimal set of driver nodes was also proposed in [106], 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 [45]. 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  $\tau$  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  $\pi_{\mathcal{T}}(\mathbf{d})$  is more difficult to control than a network with a homogeneous  $\pi_{\mathcal{T}}(\mathbf{d})$  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:  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H),  $P[p_{\mathcal{T}}(\mathbf{\Gamma}), \mathbf{sgn}(\mathbf{t})]$  (Sec. V F, and  $P[\mathbf{t}]$  and  $P[\mathbf{d}]$  (Sec. V G). 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  $\pi_{\mathcal{T}}(\mathbf{d})$  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  $P[p_{\mathcal{T}}(\mathbf{\Gamma}), \mathbf{sgn}(\mathbf{A})]$  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. [46] 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  $P[\mathbf{w}, \mathbf{t}]$  (Sec. V H),  $P[\mathbf{k}, p_{\mathcal{L}}(\Theta)]$  (Sec. V D), and  $P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}]$  (Sec. V I), as well as the time reversal reference model (Sec. V K 2), achieve more efficient controllability than their aggregated counterparts.

## VIII. CONCLUSION

We have here introduced a fundamental framework for microcanonical randomized reference models (MRRMs). This enabled 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 RRM, built a taxonomy of these RRM, 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 shuffling method. This means that they can be used to test the importance of any given feature provided a corresponding shuffling 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. V K 1). We believe that shuffling methods are best used as exploratory tools to identify important qualitative features 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 II to any other type of network is trivial as it can be applied to any discrete structure. 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 [107] require the definition of three-level nested features. Furthermore, a MRRM-based framework can be developed for any other types of multilayer networks, such as multiplex networks or networks of networks, and even for structures beyond networks such as hypergraphs. Finally, it should be helpful to define a similar framework for canonical reference models (see Section V K 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. With the foundations for MRRMs aid here we are ready to repeat the success stories of RRM for static networks, and may even go much further.

Notable important challenges remain which may now be addressed using the formalism defined 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 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 (for single features this corresponds to cuts in the dependency diagram, Fig. 11). 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 shuffling methods for MRRMs that take higher order topological correlations into account [26, 108]. This 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.

## ACKNOWLEDGMENTS

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 HARMS-flu, 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. VII) considered MRRMs specifically defined for directed temporal networks, namely  $P[\mathbf{d}_{\rightarrow}]$  and  $P[\mathbf{s}_{\rightarrow}, p(\Delta\boldsymbol{\tau})]$ . The inclusion of MRRMs for directed networks is straightforward as it simply requires defining the appropriate directed features. For features of links, no generalizations are necessary since they all generalize automatically to directed networks by using the convention that  $(i, j)$  des-

ignates 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 feature 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 features to directed temporal networks in Table V.

- 
- [1] 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).
- [2] Newman, M. E. The structure and function of complex networks. *SIAM Review* **45**, 167–256 (2003).
- [3] Fosdick, B. K., Larremore, D. B., Nishimura, J. & Ugander, J. Configuring random graph models with fixed degree sequences. *SIAM Review* **60**, 315–355 (2018).
- [4] Barrat, A., Barthélemy, M. & Vespignani, A. *Dynamical Processes on Complex Networks* (Cambridge University Press, 2008).
- [5] Gleeson, J. P. High-accuracy approximation of binary-state dynamics on networks. *Phys. Rev. Lett.* **107**, 068701 (2011).
- [6] Pastor-Satorras, R. & Vespignani, A. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.* **86**, 3200 (2001).
- [7] Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N. & Barabási, A. L. The large-scale organization of metabolic networks. *Nature* **407**, 651–654 (2000).
- [8] Solé, R. V. & Montoya, J. M. Complexity and fragility in ecological networks. *Proc. R. Soc. Lond. B* **268**, 2039–2045 (2001).
- [9] Holme, P., Kim, B. J., Yoon, C. N. & Han, S. K. Attack vulnerability of complex networks. *Phys. Rev. E* **65**, 056109 (2002).
- [10] Bollobás, B. & Riordan, O. Robustness and vulnerability of scale-free random graphs. *Internet Mathematics* **1**, 1–35 (2004).
- [11] Albert, R., Jeong, H. & Barabási, A.-L. Error and attack tolerance of complex networks. *Nature* **406**, 378 (2000).
- [12] Cohen, R., Erez, K., Ben-Avraham, D. & Havlin, S. Resilience of the internet to random breakdowns. *Phys. Rev. Lett.* **85**, 4626 (2000).
- [13] Newman, M. E. J. & Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113 (2004).
- [14] Karrer, B. & Newman, M. E. Stochastic blockmodels and community structure in networks. *Phys. Rev. E* **83**, 016107 (2011).
- [15] Maslov, S. & Sneppen, K. Specificity and Stability in Topology of Protein Networks. *Science* **296**, 910–913 (2002).
- [16] Milo, R. *et al.* Network motifs: simple building blocks of complex networks. *Science* **298**, 824–827 (2002).
- [17] Watts, D. J. & Strogatz, S. H. Collective dynamics of “small-world” networks. *Nature* **393**, 440–442 (1998).
- [18] Onnela, J.-P. *et al.* Structure and tie strengths in mobile communication networks. *Proc. Natl. Acad. Sci. U.S.A.* **104**, 7332–7336 (2007).
- [19] In the temporal network literature RRM are also known as *null models* [20–22, 25, 27, 32, 33, 38, 39, 43, 52, 56–59]; *reshuffling methods* [42], *randomization techniques* [21, 45], *randomization procedures* [21, 32, 34, 45, 52], *randomization strategies* [32], *randomization schemes* [21], *randomization methods* [47], or simply *randomizations* [20, 21, 34, 45, 47–49, 53]).
- [20] Holme, P. & Saramäki, J. Temporal networks. *Phys. Rep.* **519**, 97–125 (2012). 1108.1780.
- [21] Holme, P. Modern temporal network theory: a colloquium. *Eur. Phys. J. B* **88**, 1–30 (2015).
- [22] Kovanen, L., Karsai, M., Kaski, K., Kertész, J. & Saramäki, J. Temporal motifs in time-dependent networks. *J. Stat. Mech. Theory Exp* **2011**, P11005 (2011).
- [23] Karsai, M., Kaski, K., Barabási, A.-L. & Kertész, J. Universal features of correlated bursty behaviour. *Sci. Rep.* **2**, 397 (2012).
- [24] Karsai, M., Kaski, K. & Kertész, J. Correlated dynamics in egocentric communication networks. *PLoS One* **7**, e40612 (2012).
- [25] Kovanen, L., Kaski, K., Kertész, J. & Saramäki, J. Temporal motifs reveal homophily, gender-specific patterns, and group talk in call sequences. *Proc. Natl. Acad. Sci. U.S.A.* **110**, 18070–5 (2013).
- [26] Orsini, C. *et al.* Quantifying randomness in real networks. *Nat. Commun.* **6**, 8627 (2015).
- [27] Karsai, M. *et al.* Small but slow world: How network topology and burstiness slow down spreading. *Phys. Rev. E* **83**, 025102(R) (2011).
- [28] Rocha, L. E. C., Liljeros, F. & Holme, P. Simulated Epidemics in an Empirical Spatiotemporal Network of 50,185 Sexual Contacts. *PLoS Comp. Biol.* **7**, e1001109 (2011).
- [29] Miritello, G., Moro, E. & Lara, R. Dynamical strength of social ties in information spreading. *Phys. Rev. E* **83**, 045102(R) (2011).
- [30] Starnini, M., Baronchelli, A., Barrat, A. & Pastor-Satorras, R. Random walks on temporal networks. *Phys. Rev. E* **85**, 056115 (2012).
- [31] Kivelä, M. *et al.* Multiscale analysis of spreading in a large communication network. *J. Stat. Mech. Theory Exp* **2012**, P03005 (2012).

TABLE V: **Additional features of directed temporal networks.** Generalizations of node features in Table I to directed networks. Link features 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
<b>Topological-temporal features</b>		
$\mathcal{V}_{i\rightarrow}$	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.	$\mathcal{V}_i = \{j : (i, j) \in \mathcal{L} \text{ or } (j, i) \in \mathcal{L}\}$
<b>Topological-temporal features</b>		
$d_{i\rightarrow}^t$	Instantaneous out-degree.	$d_{i\rightarrow}^t =  \{j : (i, j) \in \mathcal{E}^t\} $
$d_{i\leftarrow}^t$	Instantaneous in-degree.	$d_{i\leftarrow}^t =  \{j : (j, i) \in \mathcal{E}^t\} $
$d_i^t$	Instantaneous (total) degree	$d_i^t =  \{j : (i, j) \in \mathcal{E}^t \text{ or } (j, i) \in \mathcal{E}^t\} $
$\Phi_{i\rightarrow}$	Node activity timeline.	$\Phi_{i\rightarrow} = ((v_{i\rightarrow}^1, \alpha_{i\rightarrow}^1), (v_{i\rightarrow}^2, \alpha_{i\rightarrow}^2), \dots, (v_{i\rightarrow}^{a_{i\rightarrow}}, \alpha_{i\rightarrow}^{a_{i\rightarrow}}))$
$\alpha_{i\rightarrow}^m$	Activity duration.	Consecutive interval during which $i$ has at least one outgoing contact.
$\Delta\alpha_{i\rightarrow}^m$	Inactivity duration.	$\Delta\alpha_{i\rightarrow}^m = v_{i\rightarrow}^{m+1} - (v_{i\rightarrow}^m + \alpha_{i\rightarrow}^m)$
<b>Aggregated features</b>		
$a_{i\rightarrow}$	Outgoing node activity.	$a_{i\rightarrow} = \sum_{j \in \mathcal{V}_{i\rightarrow}} n_{(i,j)}$
$a_{i\leftarrow}$	Ingoing node activity.	$a_{i\leftarrow} = \sum_{j \in \mathcal{V}_{i\leftarrow}} n_{(i,j)}$
$a_i$	(Total) node activity.	$a_i = \sum_{(i,j) \in \mathcal{L}_i} n_{(i,j)}$
$s_{i\rightarrow}$	Node out-strength.	$s_{i\rightarrow} = \sum_{(i,j) \in \mathcal{L}_{i\rightarrow}} w_{(i,j)}$
$s_{i\leftarrow}$	Node in-strength.	$s_{i\leftarrow} = \sum_{(i,j) \in \mathcal{L}_{i\leftarrow}} w_{(i,j)}$
$s_i$	Node (total) strength.	$s_i = \sum_{(i,j) \in \mathcal{L}_i} w_{(i,j)}$
$k_{i\rightarrow}$	Node out-degree.	$k_{i\rightarrow} =  \mathcal{V}_{i\rightarrow} $
$k_{i\leftarrow}$	Node in-degree.	$k_{i\leftarrow} =  \mathcal{V}_{i\leftarrow} $
$k_i$	Node (total) degree.	$k_i =  \mathcal{V}_i $

- [32] Gauvin, L., Panisson, A., Cattuto, C. & Barrat, A. Activity clocks: spreading dynamics on temporal networks of human contact. *Sci. Rep.* **3**, 3099 (2013).
- [33] Karimi, F. & Holme, P. Threshold model of cascades in empirical temporal networks. *Physica A* **392**, 3476–3483 (2013).
- [34] Takaguchi, T., Masuda, N. & Holme, P. Bursty Communication Patterns Facilitate Spreading in a Threshold-Based Epidemic Dynamics. *PLoS One* **8** (2013).
- [35] Holme, P. & Liljeros, F. Birth and death of links control disease spreading in empirical contact networks. *Sci. Rep.* **4**, 4999 (2014).
- [36] Karsai, M., Perra, N. & Vespignani, A. Time varying networks and the weakness of strong ties. *Sci. Rep.* **4**, 4001 (2014).
- [37] Backlund, V.-P., Saramäki, J. & Pan, R. K. Effects of temporal correlations on cascades: Threshold models on temporal networks. *Phys. Rev. E* **89**, 062815 (2014).
- [38] Cardillo, A. *et al.* Evolutionary dynamics of time-resolved social interactions. *Phys. Rev. E* **90**, 052825 (2014).
- [39] Thomas, B., Jurdak, R., Zhao, K. & Atkinson, I. Diffusion in colocation contact networks: the impact of nodal spatiotemporal dynamics. *PLoS One* **11**, e0152624 (2015).
- [40] Delvenne, J.-C., Lambiotte, R. & Rocha, L. E. C. Diffusion on networked systems is a question of time or structure. *Nat. Commun.* **6**, 7366 (2015).
- [41] Saramäki, J. & Holme, P. Exploring temporal networks with greedy walks. *Eur. Phys. J. B* **88**, 1–8 (2015).
- [42] Géniois, M., Vestergaard, C. L., Cattuto, C. & Barrat, A. Compensating for population sampling in simulations of epidemic spread on temporal contact networks. *Nat. Commun.* **6**, 8860 (2015).
- [43] Valdano, E., Poletto, C. & Colizza, V. Infection propagator approach to compute epidemic thresholds on temporal networks: impact of immunity and of limited temporal resolution. *Eur. Phys. J. B* **88**, 341 (2015).
- [44] Holme, P. Temporal network structures controlling disease spreading. *Phys. Rev. E* **94**, 022305 (2016).
- [45] Pósfai, M. & Hövel, P. Structural controllability of temporal networks. *New J. Phys.* **16**, 123055 (2014).
- [46] Li, A., Cornelius, S. P., Liu, Y.-Y., Wang, L. & Barabási, A.-L. The fundamental advantages of temporal networks. *Science* **358**, 1042–1046 (2017).
- [47] Takaguchi, T., Sato, N., Yano, K. & Masuda, N. Importance of individual events in temporal networks. *New J. Phys.* **14**, 093003 (2012).
- [48] 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).
- [49] Takaguchi, T., Yano, Y. & Yoshida, Y. Coverage centralities for temporal networks. *Eur. Phys. J. B* **89**, 35 (2016).



- [50] Tang, J., Scellato, S., Musolesi, M., Mascolo, C. & Latora, V. Small-world behavior in time-varying graphs. *Phys. Rev. E* **81**, 055101 (2010).
- [51] Alessandretti, L., Sapiezynski, P., Lehmann, S. & Baronchelli, A. Evidence for a conserved quantity in human mobility. *Nat. Hum. Behav.* **2**, 485–491 (2018).
- [52] Bajardi, P., Barrat, A., Natale, F., Savini, L. & Colizza, V. Dynamical patterns of cattle trade movements. *PLoS One* **6**, e19869 (2011).
- [53] Holme, P. Network reachability of real-world contact sequences. *Phys. Rev. E* **71**, 046119 (2005).
- [54] 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).
- [55] Sun, K., Baronchelli, A. & Perra, N. Contrasting effects of strong ties on SIR and SIS processes in temporal networks. *Eur. Phys. J. B* **88**, 326 (2015).
- [56] 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).
- [57] Squartini, T., Fagiolo, G. & Garlaschelli, D. Randomizing world trade. i. a binary network analysis. *Phys. Rev. E* **84**, 046117 (2011).
- [58] Squartini, T., Fagiolo, G. & Garlaschelli, D. Randomizing world trade. ii. a weighted network analysis. *Phys. Rev. E* **84**, 046118 (2011).
- [59] Saracco, F., Clemente, R. D., Gabrielli, A. & Squartini, T. Detecting early signs of the 2007-2008 crisis in the world trade. *Sci. Rep.* **30286** (2016).
- [60] Jaynes, E. T. Information theory and statistical mechanics. *Physical Review* **106**, 620–630 (1957).
- [61] Presse, S., Ghosh, K., Lee, J. & Dill, K. A. Principles of maximum entropy and maximum caliber in statistical physics. *Rev. Mod. Phys.* **85**, 1115–1141 (2013).
- [62] Good, P. *Permutation, Parametric and Bootstrap Tests of Hypotheses (3rd ed.)*, (Springer, 2005).
- [63] Katz, L. & Powell, J. H. Probability distributions of random variables associated with a structure of the sample space of sociometric investigations. *Ann. Math. Stat.* **28**, 442 (1957).
- [64] Snijders, T. Enumeration and simulation methods for 0–1 matrices with given marginals. *Psychometrika* **56**, 397–417 (1991).
- [65] To apply the formalism to other types of data it suffices to replace the state space  $\mathcal{G}$  in Definitions II.3 and II.7 by the appropriate state space, e.g. of multilayer networks or of hypergraphs. To describe the MRRMs one must additionally define features corresponding to the data structure as needed.
- [66] Fournet, J. & Barrat, A. Contact patterns among high school students. *PLoS One* **9**, e107878 (2014).
- [67] Stopczynski, A. *et al.* Measuring large-scale social networks with high resolution. *PLoS One* **9**, e95978 (2014).
- [68] A link-timeline network is equivalent to the *interval graph* defined in [20]; it is furthermore equivalent to the link stream defined in [109], but where the set of possible event times was defined explicitly there. If we set  $\tau_m = 0$ , i.e. in the instantaneous event limit, we obtain what is termed *contact sequences* in [20].
- [69] 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. Furthermore, for RRRMs the time can be considered finite as the observation windows and measurement resolutions are finite.
- [70] Hrbacek, K. & Jech, T. *Introduction to Set Theory, Revised and Expanded* (Crc Press, 1999).
- [71] Stanley, R. P. Enumerative combinatorics (Vol. 1, 2nd ed.). *Cambridge studies in advanced mathematics* (2011).
- [72] 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).
- [73] The number of functions leading to different MRRMs is equal to the the number of possible partitions of the state space of temporal networks of a given size. In practice, the number of possible temporal networks is very large and the number of partitions of the state space is a super-exponential function of this number (see Section II B).
- [74] Perotti, J. I., Jo, H.-H., Holme, P. & Saramäki, J. Temporal network sparsity and the slowing down of spreading. *arXiv:1411.5553* (2014).
- [75] These node-grouped MRRMs can be seen as micro-canonical variants of the stochastic block model [110]. However, typical stochastic block models found in the literature assign either the links at random inside each block (i.e. equivalent to  $P[\sigma, \Sigma_{\mathcal{L}}]$ ) or while constraining the degree sequence (equivalent to  $P[\mathbf{k}, \sigma, \Sigma_{\mathcal{L}}]$ ) [14], while the metadata MRRMs we consider here may be any refinement of these.
- [76] Zhang, X., Moore, C. & Newman, M. E. Random graph models for dynamic networks. *Eur. Phys. J. B* **90**, 200 (2017).
- [77] Squartini, T., Mastrandrea, R. & Garlaschelli, D. Unbiased sampling of network ensembles. *New J. Phys.* **17**, 023052 (2015).
- [78] Squartini, T. & Garlaschelli, D. Analytical maximum-likelihood method to detect patterns in real networks. *New J. Phys.* **13**, 083001 (2011).
- [79] Perra, N., Gonçalves, B., Pastor-Satorras, R. & Vespignani, A. Activity driven modeling of time varying networks. *Sci. Rep.* **2**, 469 (2012).
- [80] Pfitzner, R., Scholtes, I., Garas, A., Tessone, C. J. & Schweitzer, F. Betweenness preference: Quantifying correlations in the topological dynamics of temporal networks. *Phys. Rev. Lett.* **110**, 198701 (2013). 1208.0588.
- [81] 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. *Nat. Commun.* **5**, 4630 (2014).
- [82] Scholtes, I. *et al.* Causality-driven slow-down and speed-up of diffusion in non-Markovian temporal networks. *Nat. Commun.* **5**, 5024 (2014).
- [83] Peixoto, T. P. Inferring the mesoscale structure of layered, edge-valued, and time-varying networks. *Phys. Rev. E* **92**, 042807 (2015).
- [84] Peixoto, T. P. & Rosvall, M. Modeling sequences and temporal networks with dynamic community structures. *Nat. Commun.* **8**, 582 (2017).

- [85] Peixoto, T. P. & Gauvin, L. Change points, memory and epidemic spreading in temporal networks. *Sci. Rep.* **8**, 15511 (2018).
- [86] Casiraghi, G., Nanumyan, V., Scholtes, I. & Schweitzer, F. Generalized hypergeometric ensembles: Statistical hypothesis testing in complex networks. *arXiv:1607.02441* (2016).
- [87] Miritello, G., Lara, R., Cebrian, M. & Moro, E. Limited communication capacity unveils strategies for human interaction. *Sci. Rep.* **3**, 1950 (2013).
- [88] Stehlé, J. *et al.* High-resolution measurements of face-to-face contact patterns in a primary school. *PloS One* **6**, e23176 (2011).
- [89] Gemmetto, V., Barrat, A. & Cattuto, C. Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC Infect. Dis.* **14**, 695 (2014).
- [90] Lin, J. Divergence measures based on the Shannon entropy. *IEEE Trans. Inf. Theory* **37**, 145–151 (1991).
- [91] Wu, Y., Zhou, C., Xiao, J., Kurths, J. & Schellnhuber, H. J. Evidence for a bimodal distribution in human communication. *Proc. Natl. Acad. Sci. U.S.A.* **107**, 18803–18808 (2010).
- [92] Pan, R. K. & Saramäki, J. Path lengths, correlations, and centrality in temporal networks. *Phys. Rev. E* **81**, 016105 (2011).
- [93] Note that the temporal distance is not a metric distance as the temporal distance from  $i$  to  $j$  generally differs from the temporal distance from  $j$  to  $i$ .
- [94] Granovetter, M. S. The strength of weak ties. *Am. J. Sociol.* **78**, 1360–1380 (1973).
- [95] Pastor-Satorras, R., Castellano, C., Mieghem, P. V. & Vespignani, A. Epidemic processes in complex networks. *Rev. Mod. Phys.* **87**, 925 (2015).
- [96] Vestergaard, C. L. & Génois, M. Temporal gillespie algorithm: Fast simulation of contagion processes on time-varying networks. *PLoS Comp. Biol.* **11**, e1004579 (2015).
- [97] Alon, U. Network motifs: theory and experimental approaches. *Nat. Rev. Genet.* **8**, 450–461 (2007).
- [98] Shen-Orr, S. S., Milo, R., Mangan, S. & Alon, U. Network motifs in the transcriptional regulation network of *escherichia coli*. *Nat. Genet.* **31**, 64 (2002).
- [99] 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 [92], but with a minor effect which does not qualitatively change the results.
- [100] Watts, D. J. A simple model of global cascades on random networks. *Proc. Natl. Acad. Sci. U.S.A.* **99**, 5766–5771 (2002).
- [101] Goh, K. I. & Barabási, A.-L. Burstiness and memory in complex systems. *EPL* **81**, 48002 (2008).
- [102] Nowak, M. A. *Evolutionary Dynamics: Exploring the Equations of Life* (Harvard University Press, 2006).
- [103] Kikas, R., Dumas, M. & Karsai, M. Bursty egocentric network evolution in Skype. *SNAM* **3**, 1393–1401 (2013).
- [104] Karsai, M., Iñiguez, G., Kikas, R., Kaski, K. & Kertész, J. Local cascades induced global contagion: How heterogeneous thresholds, exogenous effects, and unconcerned behaviour govern online adoption spreading. *Sci. Rep.* **6** (2016).
- [105] Kirk, D. E. *Optimal Control Theory: An Introduction* (Prentice-Hall, Inc, 1971).
- [106] Liu, Y.-Y., Slotine, J.-J. & Barabási, A.-L. Controllability of complex networks. *Nature* **473**, 167–173 (2011).
- [107] Kivelä, M. *et al.* Multilayer networks. *J. Complex Netw.* **2**, 203–271 (2014).
- [108] Jerrum, M. R., Valiant, L. G. & Vazirani, V. V. Random generation of combinatorial structures from a uniform distribution. *Theor. Comput. Sci.* **43**, 169–188 (1986).
- [109] Latapy, M., Viard, T. & Magnien, C. Stream graphs and link streams for the modeling of interactions over time. *arXiv:1710.04073* (2017).
- [110] Holland, P. W., Laskey, K. B. & Leinhardt, S. Stochastic blockmodels: First steps. *Soc. networks* **5**, 109–137 (1983).

## Supplementary tables and figures

**Supplementary TABLE I: Extended list of marginals and moments of a sequence of features.** Below,  $(\cdot)$  denotes an ordered sequence and  $[\cdot]$  denotes a multiset, equivalent to the empirical distribution.

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

<sup>a</sup>  $\mathbf{x}_{(i,j)}$  : Local sequence on link,  $\mathbf{x}_{(i,j)} = (x_{(i,j)}^m)_{m \in \mathcal{M}_{(i,j)}}$ .

<sup>b</sup>  $\mathbf{x}_i$  : Local sequence on node,  $\mathbf{x}_i = (x_i^m)_{m \in \mathcal{M}_i}$ .

<sup>c</sup>  $\mathbf{x}^t$  : Local sequence in snapshot,  $\mathbf{x}^t = (x_i^t)_{i \in \mathcal{V}}$ .

<sup>d</sup>  $\pi_{(i,j)}(\mathbf{x}_{(i,j)})$  : Local distribution on link,  $\pi_{(i,j)}(\mathbf{x}_{(i,j)}) = [x_{(i,j)}^m]_{m \in \mathcal{M}_{(i,j)}}$ .

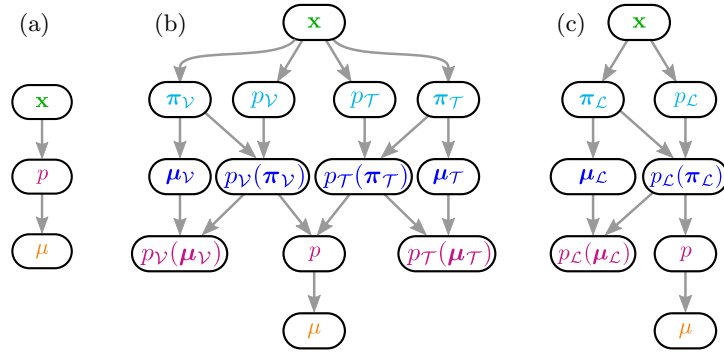
<sup>e</sup>  $\pi_i(\mathbf{x}_i)$  : Local distribution on node,  $\pi_i(\mathbf{x}_i) = [x_i^m]_{m \in \mathcal{M}_i}$ .

<sup>f</sup>  $\pi^t(\mathbf{x}^t)$  : Local distribution in snapshot,  $\pi^t(\mathbf{x}^t) = [x_i^t]_{i \in \mathcal{V}}$ .

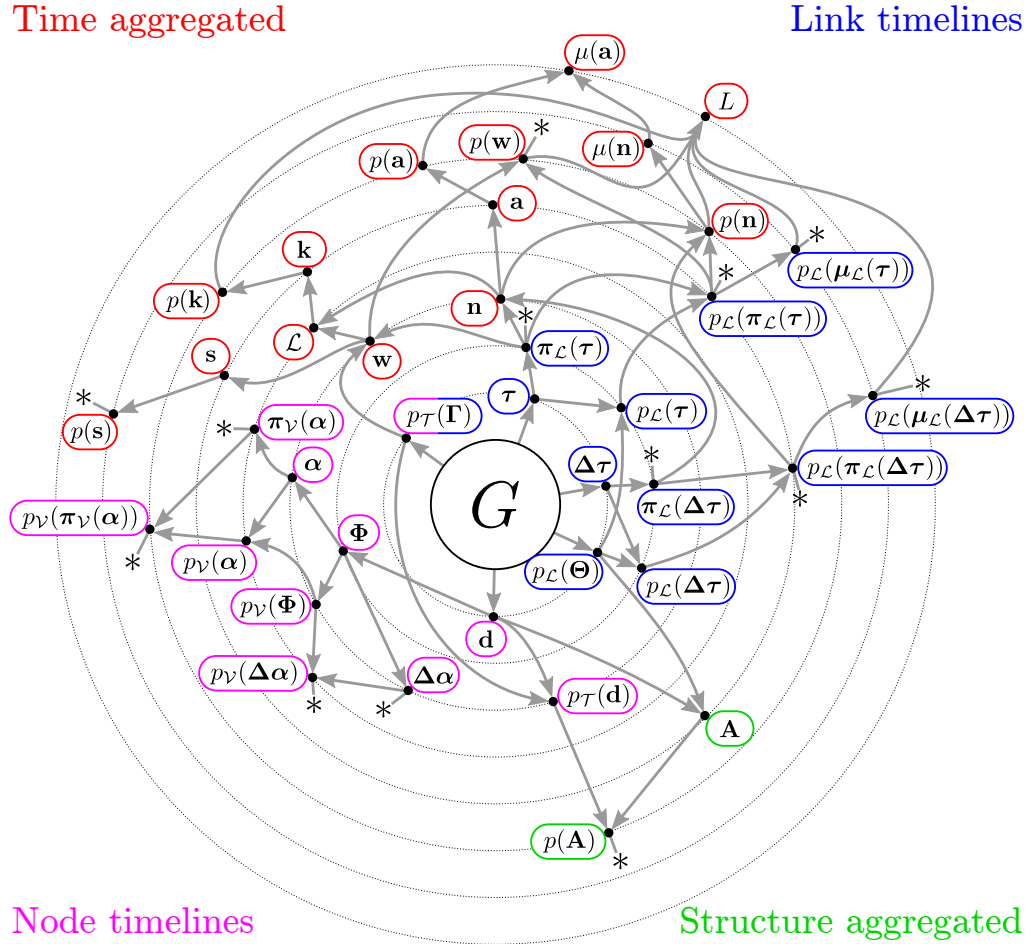
<sup>g</sup>  $\mu_{(i,j)}(\mathbf{x}_{(i,j)})$  : Local mean on link,  $\mu_{(i,j)}(\mathbf{x}_{(i,j)}) = \sum_{m \in \mathcal{M}_{(i,j)}} x_{(i,j)}^m / M_{(i,j)}$ .

<sup>h</sup>  $\mu_i(\mathbf{x}_i)$  : Local mean on node,  $\mu_i(\mathbf{x}_i) = \sum_{m \in \mathcal{M}_i} x_i^m / M_i$ .

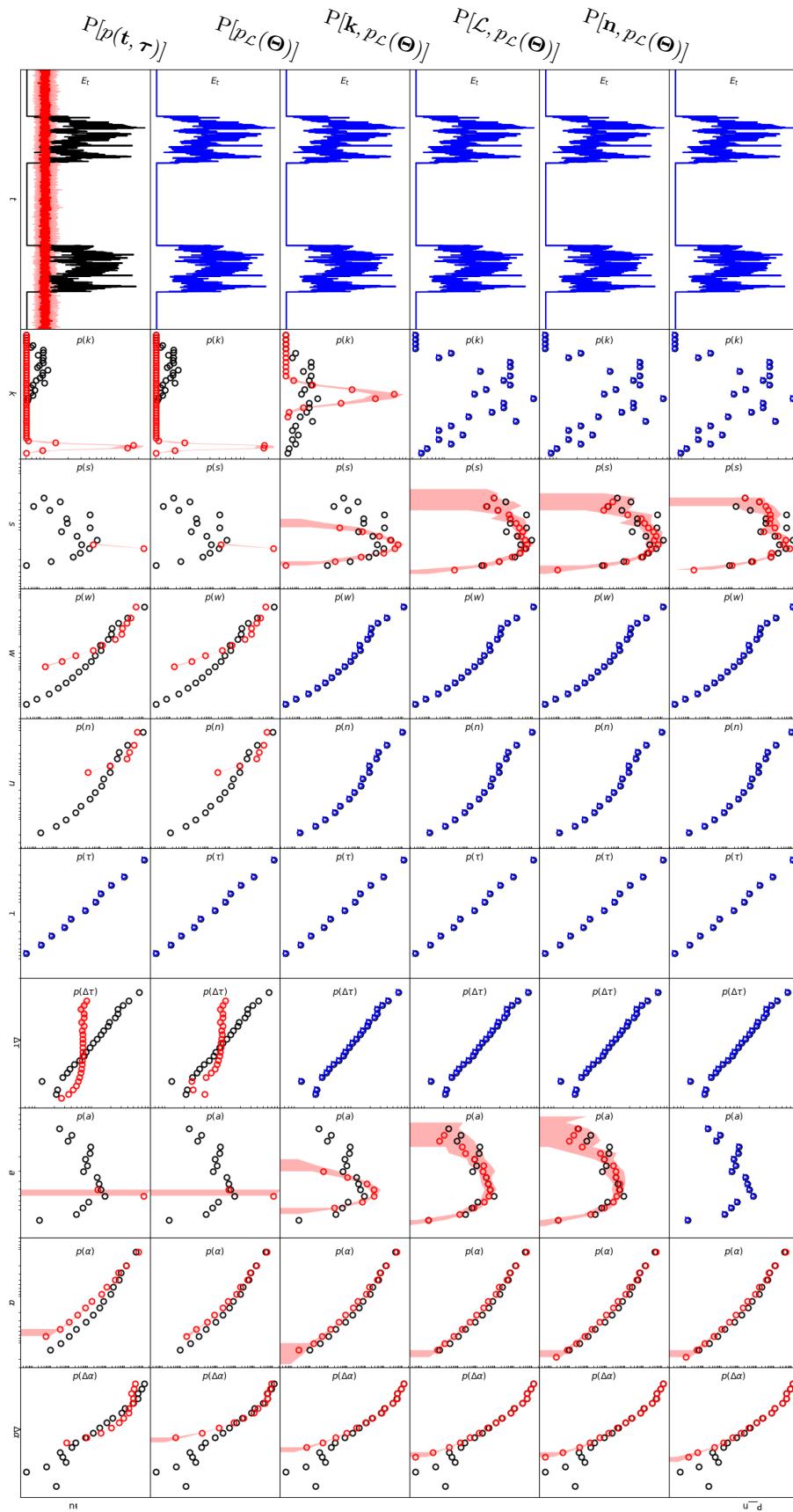
<sup>i</sup>  $\mu^t(\mathbf{x}^t)$  : Local mean in snapshot,  $\mu^t(\mathbf{x}^t) = \sum_{i \in \mathcal{V}} x_i^t / N$ .



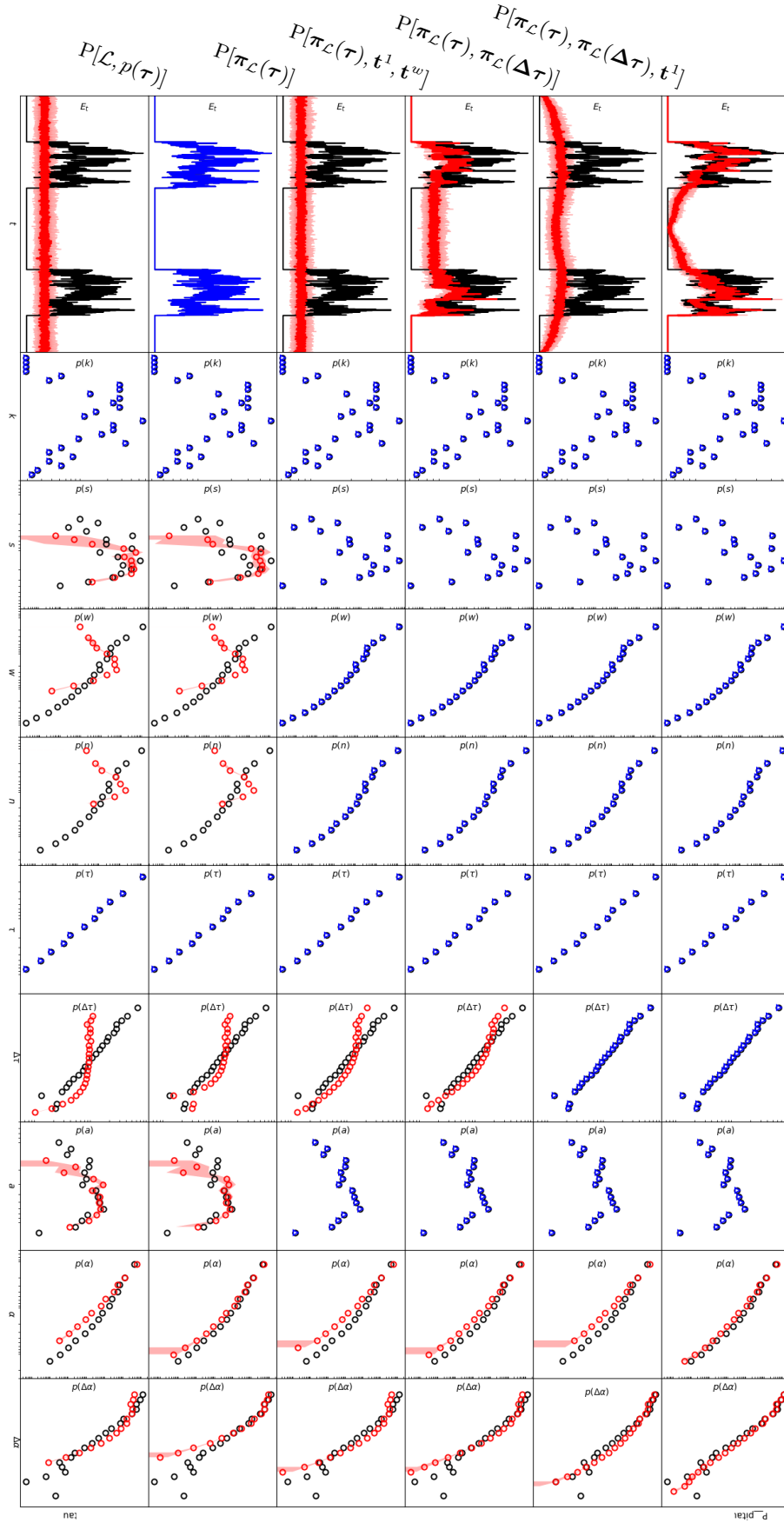
**Supplementary FIG. 1: Extended hierarchy of the marginals and moments of a sequence of characteristics.** An arrow from a higher node to a lower one indicates that the former feature is finer than the latter. Thus, a MRRM that conserves the former feature necessarily conserves all downstream features. Conversely, a MRRM that randomizes a given feature also randomizes any features upstream of it as well. (a) For one-level sequences of features, 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 features of nodes, namely  $\alpha$ ,  $\Delta\alpha$  and  $\mathbf{d}$ . (c) For two-level sequences of features of link timelines, namely  $\tau$  and  $\Delta\tau$ .



**Supplementary FIG. 2: Extended hierarchy of different temporal network features.** 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 Table I in the main text and Supplementary Table I for definitions of the features. A star (\*) emanating from a node indicates that lower hierarchical levels follow as shown in Supplementary Fig. 1. The color coding shows what type of features the features correspond to: time-aggregated features (i.e. topological and weighted), link-timeline features, node-timeline features, and structure-aggregated features (i.e. purely temporal).



**Supplementary FIG. 3: 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 VI A of the manuscript 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.



**Supplementary FIG. 4: 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 VIA of the manuscript 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.



# Supplementary Note: Names of MRRM algorithms in the Python library

## Instant-event shufflings

- $P[1]$ : `P__1`

### Timeline shufflings.

- $P[\mathcal{L}]$ : `P__L`
- $P[\mathbf{w}]$ : `P__w`
- $P[\mathbf{w}, \mathbf{t}^1 \mathbf{t}^w]$ : `P__w_t1_tw`
- $P[\pi_{\mathcal{L}}(\Delta\tau)]$ : `P__pidtau`
- $P[\pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^1 \mathbf{t}^w]$ : `P__pidtau_t1_tw`

### Sequence shufflings.

- $P[p_{\mathcal{T}}(\Gamma)]$ : `P__pGamma`
- $P[p_{\mathcal{T}}(\Gamma), \text{sgn}(\mathbf{A})]$ : `P__pGamma_sgnA`

### Snapshot shufflings.

- $P[\mathbf{t}]$ : `P__t`
- $P[\mathbf{t}, \Phi]$ : `P__t_Phi`
- $P[\mathbf{d}]$ : `P__d`
- $P[\text{iso}(\Gamma)]$ : `P__isoGamma`
- $P[\text{iso}(\Gamma), \Phi]$ : `P__isoGamma_Phi`

### Intersections.

- $P[\mathcal{L}, \mathbf{t}]$ : `P__L_t`
- $P[\mathbf{w}, \mathbf{t}]$ : `P__w_t`

### Compositions.

- $P[L]$ : `P__pTheta` with `P__L_E`
- $P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}]$ : `P__pTheta` with `P__w_t`
- $P[\mathbf{k}, \mathbb{I}_{\lambda}, p(\mathbf{w}), \mathbf{t}]$ : with `P__w_t`

## Event shufflings

- $P[p(\tau)]$ : `P__ptau`

### Link shufflings.

- $P[p_{\mathcal{L}}(\Theta)]$ : `P__pTheta`
- $P[\mathbb{I}_{\lambda}, p_{\mathcal{L}}(\Theta)]$ : `P__I_pTheta`
- $P[\mathbf{k}, p_{\mathcal{L}}(\Theta)]$ : `P__k_pTheta`
- $P[\mathbf{k}, \mathbb{I}_{\lambda}, p_{\mathcal{L}}(\Theta)]$ : `P__k_I_pTheta`

### Timeline shufflings.

- $P[\mathcal{L}, p(\tau)]$ : `P__L_ptau`
- $P[\pi_{\mathcal{L}}(\tau)]$ : `P__pitau`
- $P[\pi_{\mathcal{L}}(\tau), \mathbf{t}^1 \mathbf{t}^w]$ : `P__pitau_t1_tw`
- $P[\pi_{\mathcal{L}}(\tau), \pi_{\mathcal{L}}(\Delta\tau)]$ : `P__pitau_pidtau`
- $P[\pi_{\mathcal{L}}(\tau), \pi_{\mathcal{L}}(\Delta\tau), \mathbf{t}^1]$ : `P__pitau_pidtau_t1`
- $P[\text{per}(\Theta)]$ : `P__perTheta`
- $P[\tau, \Delta\tau]$ : `P__tau_dtau`

### Snapshot shufflings.

- $P[p(\mathbf{t}, \tau)]$ : `P__pttau`

### Intersections.

- $P[\mathcal{L}, p(\mathbf{t}, \tau)]$ : `P__L_pttau`
- $P[\mathbf{n}, p(\mathbf{t}, \tau)]$ : `P__n_pttau`
- $P[\mathcal{L}, p_{\mathcal{L}}(\Theta)]$ : `P__L_pTheta`
- $P[\mathbf{w}, p_{\mathcal{L}}(\Theta)]$ : `P__w_pTheta`
- $P[\mathbf{n}, p_{\mathcal{L}}(\Theta)]$ : `P__n_pTheta`

## Metadata shufflings

### Link shufflings.

- $P[p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$ : `P__pTheta_sigma_SigmaL`
- $P[\mathbf{k}, p_{\mathcal{L}}(\Theta), \sigma, \Sigma_{\mathcal{L}}]$ : `P__k_pTheta_sigma_SigmaL`
- $P[G, p(\sigma)]$ : `P__G_psigma`

### Compositions.

- $P[\mathbf{k}, p(\mathbf{w}), \mathbf{t}, \sigma, \Sigma_{\mathcal{L}}]$ : `P__k_LCM` with `P__w_t`