



## Mutually Exciting Point Process Graphs for Modeling Dynamic Networks

Francesco Sanna Passino & Nicholas A. Heard

To cite this article: Francesco Sanna Passino & Nicholas A. Heard (2022): Mutually Exciting Point Process Graphs for Modeling Dynamic Networks, Journal of Computational and Graphical Statistics, DOI: [10.1080/10618600.2022.2096048](https://doi.org/10.1080/10618600.2022.2096048)

To link to this article: <https://doi.org/10.1080/10618600.2022.2096048>



© 2022 The Author(s). Published with license by Taylor & Francis Group, LLC.



[View supplementary material](#)



Published online: 06 Sep 2022.



[Submit your article to this journal](#)



Article views: 295



[View related articles](#)



[View Crossmark data](#)

# Mutually Exciting Point Process Graphs for Modeling Dynamic Networks

Francesco Sanna Passino and Nicholas A. Heard

Department of Mathematics, Imperial College London, London, United Kingdom

## ABSTRACT

A new class of models for dynamic networks is proposed, called mutually exciting point process graphs (MEG). MEG is a scalable network-wide statistical model for point processes with dyadic marks, which can be used for anomaly detection when assessing the significance of future events, including previously unobserved connections between nodes. The model combines mutually exciting point processes to estimate dependencies between events and latent space models to infer relationships between the nodes. The intensity functions for each network edge are characterized exclusively by node-specific parameters, which allows information to be shared across the network. This construction enables estimation of intensities even for unobserved edges, which is particularly important in real world applications, such as computer networks arising in cyber-security. A recursive form of the log-likelihood function for MEG is obtained, which is used to derive fast inferential procedures via modern gradient ascent algorithms. An alternative EM algorithm is also derived. The model and algorithms are tested on simulated graphs and real world datasets, demonstrating excellent performance. Supplementary materials for this article are available online.

## ARTICLE HISTORY

Received February 2021  
Accepted June 2022

## KEYWORDS

Dynamic networks; Hawkes process; Self-exciting process; Statistical cyber-security

## 1. Introduction

Dynamic networks are encountered in many domains, representing, for example, interactions in social networks, messaging applications, or computer networks. Event data from dynamic networks are observed as triplets  $(t_1, x_1, y_1), \dots, (t_m, x_m, y_m)$ , where  $0 \leq t_1 \leq t_2 \leq \dots$  are event times and the dyadic marks  $(x_k, y_k)$  denote the source and destination nodes, each belonging to a set of nodes  $V = \{1, \dots, n\}$  of size  $n$ . The sequence of graph edges  $(x_1, y_1), \dots, (x_m, y_m)$  induces a directed *network adjacency matrix*  $\mathbf{A} = \{A_{ij}\} \in \{0, 1\}^{n \times n}$  where  $A_{ij} = 1$  if node  $i$  connected to node  $j$  at least once during the entire observation period, and  $A_{ij} = 0$  otherwise. This article presents a new class of models for the arrival of connection events between nodes in a network, called *mutually exciting graphs* (MEG). The MEG model builds upon *mutually exciting point processes* and *latent space models*.

Mutually exciting point processes have been already successfully used for a variety of different applications: modeling of earthquakes (Ogata 1988), financial markets (Bowsher 2007), criminal activities (Mohler et al. 2011; Stomakhin, Short, and Bertozzi 2011), and popularity of tweets (Zhao et al. 2015; Chen and Tan 2018). Let  $t_1, t_2, \dots, t_m$  denote an increasing sequence of observed event times, and  $N(t) = \sum_{k=1}^m \mathbb{1}_{[0,t]}(t_k)$  the corresponding counting process, representing the number of events observed up to time  $t$ . A counting process can be characterized by its *conditional intensity function*  $\lambda(t) = \lim_{\delta \rightarrow 0} \mathbb{E}[N(t + \delta) - N(t) | \mathcal{H}_t] / \delta$ , representing the expected rate of event times conditioned on the history  $\mathcal{H}_t$  of the process up to time  $t$ . For self-exciting processes, the conditional intensity  $\lambda(t)$  is assumed


to depend on the last  $r$  observed arrival times:

$$\lambda(t) = \lambda + \sum_{k>N(t)-r}^{N(t)} \omega(t - t_k), \quad (1)$$

where  $\lambda \in \mathbb{R}_+$  is a baseline intensity level and  $\omega(\cdot)$  is a non-increasing and nonnegative excitation function. For simplicity,  $\omega(\cdot)$  is usually chosen to be a scaled exponential function:  $\omega(t) = \beta \exp\{-(\beta + \theta)t\}$ , where  $\beta \geq 0$  and  $\theta > 0$ . Usually,  $\beta$  is referred to as *jump* and  $\beta + \theta$  as *decay rate*. Alternative choices of  $\omega(\cdot)$  are nonparametric step functions (Price-Williams and Heard 2020), or the power-law  $\omega(t) = \theta(t + \gamma)^{-1-\delta}$ , where  $\theta \geq 0$ ,  $\beta, \delta > 0$  and  $\theta < \delta\beta^\delta$  (Ozaki 1979). In the literature, two extreme cases for the intensity in (1) are usually considered:  $r = 1$ , corresponding to a first order Markov-like structure, and  $r = \infty$ , called a Hawkes process (Hawkes 1971). Intuitively, if  $r = 1$ , the intensity only depends on the time elapsed since the last event. On the other hand, if  $r = \infty$ , the conditional intensity depends on *all* observed events, downweighted according to the elapsed time. If  $r = 0$ , the model reduces to a simple Poisson process, such that all inter-arrival times are independent and exponentially distributed with rate  $\lambda$ .

In large graphs, simultaneously modeling all the edge processes using individual intensities of the form (1) is computationally challenging, and ignores possible correlations between different edges and nodes. Inference would require estimating  $\mathcal{O}(n^2)$  parameters, or  $\mathcal{O}\{\text{nnz}(\mathbf{A})\}$  parameters if the graph is sparse, where  $\text{nnz}(\cdot)$  denotes the number of nonzero entries in a matrix. This is not feasible in most real-world applications.

**CONTACT** Francesco Sanna Passino  [f.sannapassino@imperial.ac.uk](mailto:f.sannapassino@imperial.ac.uk)  Department of Mathematics, Imperial College London, London, UK.

 Supplementary materials for this article are available online. Please go to [www.tandfonline.com/r/JCGS](http://www.tandfonline.com/r/JCGS).

© 2022 The Author(s). Published with license by Taylor & Francis Group, LLC.

This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (<http://creativecommons.org/licenses/by-nc-nd/4.0/>), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited, and is not altered, transformed, or built upon in any way.

Furthermore, this approach would not parameterize *new edges* appearing after the model training period. Hence, traditional statistical models for networks, for example, latent space models (Hoff, Raftery, and Handcock 2002), aim to reduce the representation of the network to  $\mathcal{O}(n)$  parameters. Inspired by the literature on latent space models for network adjacency matrices, here a dynamic graph is modeled through the edge-specific point processes with intensity functions parameterized by node-specific latent features. In standard latent space network models, the probability of a link between two nodes is expressed as a function of node-specific latent vectors  $\mathbf{a}_i, \mathbf{b}_j \in \mathbb{R}^d$ , such that  $\mathbb{P}(A_{ij} = 1) = f(\mathbf{a}_i, \mathbf{b}_j)$ , for some kernel function  $f$ . In this work, it is assumed that the arrival times on each observed network edge can be modeled using a mutually exciting point process depending on node-specific characteristics.

The related literature on mutually exciting point processes is vast, although mostly focusing on univariate and multivariate point processes; limited attention is devoted to using such processes for modeling large dynamic graphs. Hawkes processes are traditionally used to estimate causal interactions within multivariate processes (Linderman and Adams 2014), because of their appealing theoretical properties in terms of Granger causality and directed information (Etesami et al. 2016; Eichler, Dahlhaus, and Dueck 2017). Hawkes processes have also been used in Fox et al. (2016) to analyze e-mail networks, primarily focusing on point processes on each node. Blundell, Beck, and Heller (2012) proposed Hawkes processes to model reciprocating relationships between graph communities. Miscouridou, Caron, and Teh (2018) extend this approach, proposing Hawkes process models for temporal interaction data with reciprocation, using compound completely random measures. The approach proposed in this article is also related to Perry and Wolfe (2013), who consider directed interactions within dynamic networks as a multivariate point process using a Cox multiplicative intensity model, with covariates depending on the history of the process. The MEG model proposed in this work is different from alternative methodologies proposed in the literature, since it uses mutually exciting processes at the edge level, parameterized only by node-specific features.

Furthermore, dynamic models for network snapshots observed at *discrete* points in time have also been proposed in the literature. In particular, the methodology proposed in this work could be related to dynamic latent space models, which are based on latent feature representations of each node, evolving according to a temporal dynamics. Examples are Sarkar and Moore (2006), Krivitsky and Handcock (2014), Sewell and Chen (2015), Durante and Dunson (2016), and Lee et al. (2021). The MEG model proposed in this article extends the latent feature framework to a continuous time setting, using node-specific latent vectors to parameterize point processes on each edge.

The remainder of this article is structured as follows: [Section 2](#) introduces the MEG model, followed by a description of the related inferential procedures in [Section 3](#). [Section 4](#) discusses simulation from the model and the calculation of  $p$ -values for each network event. Results on simulated and real-world computer networks are discussed in [Section 5](#).

## 2. Mutually Exciting Point Process Graphs

The main contribution proposed in this article is a *mutually exciting graph model* (MEG) for dynamic network point processes, defined by an  $n \times n$  time-varying matrix of nonnegative functions  $\boldsymbol{\lambda}(t) = \{\lambda_{ij}(t)\}$ . Each entry  $\lambda_{ij}(t)$  represents the conditional intensity of the counting process  $N_{ij}(t) = \sum_{k=1}^m \mathbb{1}_{[0,t] \times \{i\} \times \{j\}}(t_k, x_k, y_k)$  of events occurring on the graph edge  $(i, j)$ , such that  $\lambda_{ij}(t) = \lim_{\delta \rightarrow 0} \mathbb{E}[N_{ij}(t+\delta) - N_{ij}(t) | \mathcal{H}_t] / \delta$ . For generality, it is assumed that for each edge  $(i, j)$  there exists a changepoint  $\tau_{ij} \geq 0$  after which the edge becomes observable. In the simplest case,  $\tau_{ij} = 0$  for all  $i$  and  $j$ .

To parameterize  $\boldsymbol{\lambda}(t)$ , each entry is represented as an additive model with three nonnegative components. The first, denoted  $\alpha_i(t)$ , characterizes the process of arrival times involving  $i$  as source node; the second,  $\beta_j(t)$ , corresponds to arrivals for which  $j$  is the destination node; the third,  $\gamma_{ij}(t)$ , is an interaction term which will also be parameterized by node-specific parameters, giving:

$$\lambda_{ij}(t) = \alpha_i(t) + \beta_j(t) + \gamma_{ij}(t), \quad t \geq \tau_{ij}. \quad (2)$$

Note that the intensity function (2) resembles the link function used in additive and multiplicative effect network models for network adjacency matrices, proposed in Hoff (2021).

Define the source and destination counting processes as  $N_i(t) = \sum_{k=1}^m \mathbb{1}_{[0,t] \times \{i\}}(t_k, x_k)$  and  $N'_j(t) = \sum_{k=1}^m \mathbb{1}_{[0,t] \times \{j\}}(t_k, y_k)$ . Furthermore, let  $\ell_{i1}, \ell_{i2}, \dots$  denote the indices  $\{k : x_k = i\}$  of the arrival times such that  $i$  appears as source node, and  $\ell'_{j1}, \ell'_{j2}, \dots$  denote the event indices  $\{k : y_k = j\}$  for which  $j$  is the destination node. To allow self-excitation of both source and destination nodes, the latent functions  $\alpha_i(t)$  and  $\beta_j(t)$  are assigned a similar form to the conditional intensity (1):

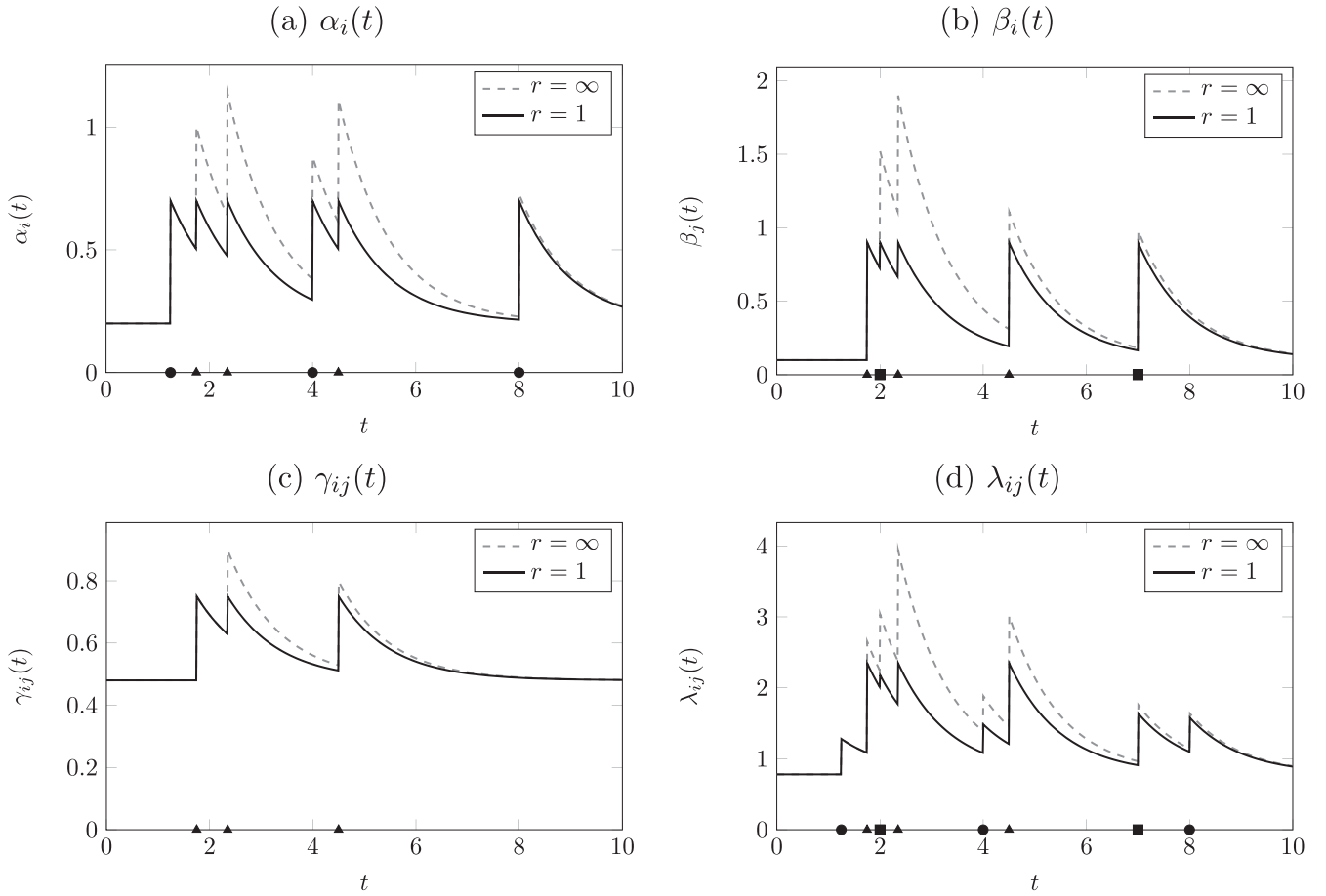
$$\begin{aligned} \alpha_i(t) &= \alpha_i + \sum_{k > N_i(t)-r}^{N_i(t)} \omega_i(t - t_{\ell_{ik}}), \\ \beta_j(t) &= \beta_j + \sum_{k > N'_j(t)-r}^{N'_j(t)} \omega'_j(t - t_{\ell'_{jk}}), \end{aligned} \quad (3)$$

where  $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_n), \boldsymbol{\beta} = (\beta_1, \dots, \beta_n) \in \mathbb{R}_+^n$  are node-specific baseline intensity levels, and  $\omega_i, \omega'_i$  are node-specific, nonincreasing excitation functions from  $\mathbb{R}_+$  to  $\mathbb{R}_+$ . For simplicity, the excitation functions assume the following scaled exponential form, for nonnegative parameters  $\mu_i, \mu'_j, \phi_i, \phi'_j \in \mathbb{R}_+^n$ :

$$\begin{aligned} \omega_i(t) &= \mu_i \exp\{-(\mu_i + \phi_i)t\}, \\ \omega'_j(t) &= \mu'_j \exp\{-(\mu'_j + \phi'_j)t\}. \end{aligned} \quad (4)$$

Scaled exponential excitation functions have significant computational advantages for inference in MEG models: in particular, the log-likelihood can be expressed in recursive form and evaluated in linear time, which speeds up inference and allows the methodology to scale to large graphs. These aspects will be more extensively discussed in [Section 2.1](#).

Similarly, let  $\ell_{ij1}, \ell_{ij2}, \dots$  be the indices  $\{k : x_k = i, y_k = j\}$  of the events observed on the edge  $(i, j)$ . The interaction term



**Figure 1.** Cartoon of a one-dimensional MEG model (2)–(6) for  $r = 1$  and  $r = \infty$ . Event times are marked on the  $x$ -axis. Events with source node  $i$  and destination node  $j$  are denoted by triangles ( $\blacktriangle$ ); other events with source node  $i$  are denoted with circles ( $\bullet$ ), and other events with destination node  $j$  are denoted by squares ( $\blacksquare$ ). For each event time, a jump in the corresponding intensities is observed.

$\gamma_{ij}(t)$  in (2) assumes a similar form to (3), but with a background rate obtained as the inner product between two node-specific  $d$ -dimensional baseline parameter vectors  $\boldsymbol{\gamma}_i, \boldsymbol{\gamma}'_j \in \mathbb{R}_+^d$ ,  $d \in \mathbb{N}$ :

$$\gamma_{ij}(t) = \boldsymbol{\gamma}_i^\top \boldsymbol{\gamma}'_j + \sum_{k > N_{ij}(t) - r}^{N_{ij}(t)} \omega_{ij}(t - t_{\ell_{ijk}}). \quad (5)$$

The excitation function  $\omega_{ij}(t)$  is also expressed as a sum of scaled exponential functions, parameterized by four node-specific, nonnegative latent  $d$ -vectors  $\mathbf{v}_i, \mathbf{v}'_j, \boldsymbol{\theta}_i, \boldsymbol{\theta}'_j \in \mathbb{R}_+^d$ :

$$\omega_{ij}(t) = \sum_{\ell=1}^d v_{i\ell} v'_{j\ell} \exp\{-(\theta_{i\ell} + v_{i\ell})(\theta'_{j\ell} + v'_{j\ell})t\}. \quad (6)$$

The inner product baseline and products within the scaled exponential excitation functions are inspired by random dot product graph models (see, e.g., Athreya et al. 2018) for link probabilities. This choice is helpful to obtain closed form expression for inference in MEG models, as discussed in Section 3.1. Alternative options, inspired by other latent space models, could also be used for the baseline, such as  $\|\boldsymbol{\gamma}_i - \boldsymbol{\gamma}'_j\|_2$  (Hoff, Raftery, and Handcock 2002).

A cartoon example of the intensity  $\lambda_{ij}(t)$  for the  $d = 1$  dimensional MEG model with scaled exponential functions is given in Figure 1, with  $\alpha_i = 0.2$ ,  $\mu_i = 0.5$ ,  $\phi_i = 0.5$ ,  $\beta_j =$

$0.1$ ,  $\mu'_j = 0.8$ ,  $\phi'_j = 0.2$ ,  $\gamma_i = 0.8$ ,  $v_i = 0.9$ ,  $\theta_i = 1.1$ ,  $\gamma'_j = 0.6$ ,  $v'_j = 0.3$ ,  $\theta'_j = 0.2$ . In Figure 1(d), the edge intensity function jumps at each event time involving source node  $i$  or destination node  $j$ , or both. In particular, larger jumps in  $\lambda_{ij}(t)$ , of size  $\mu_i + \mu'_j + v_i v'_j$  for  $r = \infty$ , are observed when events are observed on the edge  $(i, j)$  (triangles). The intensity also increases if events are observed from source node  $i$  (circles, see, Figure 1(a)) or to destination node  $j$  (squares, see, Figure 1(b)), with jumps of size  $\mu_i$  and  $\mu'_j$ , respectively, for  $r = \infty$ . For  $r = 1$ , the intensity  $\lambda_{ij}(t)$  is bounded by construction at  $\alpha_i + \beta_j + \gamma_i \gamma'_j + \mu_i + \mu'_j + v_i v'_j$ .

A key feature of the model is the representation of the intensity (2) with only node-specific parameters  $\Psi = (\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\phi}, \boldsymbol{\beta}, \boldsymbol{\mu}', \boldsymbol{\phi}', \boldsymbol{\gamma}, \mathbf{v}, \boldsymbol{\theta}, \boldsymbol{\gamma}', \mathbf{v}', \boldsymbol{\theta}')$ . This construction allows estimation of intensities even for unobserved edges. Therefore, in practical applications, when a new link is observed, it is possible to immediately provide an estimate of the intensity of the process on that edge. This is a substantial difference with respect to models based on edge-specific parameters for the edge intensities. Such models do not perform well for scoring new links, since the score for a new observation could only be based on a prior guess of the intensity, whereas MEG “borrows strength” from events observed on similar nodes and edges in the graph, providing an informed estimate of the intensity function on the newly observed edge.

For a sequence of observed events  $\mathcal{H}_T = \{(x_1, y_1, t_1), \dots, (x_m, y_m, t_m)\}$ , with event times in  $[0, T]$ , the log-likelihood (Daley and Vere-Jones 2002) of a generic MEG model is

$$\log L(\mathcal{H}_T; \Psi) = \sum_{i=1}^n \sum_{j=1}^n \left\{ \sum_{k=1}^{n_{ij}} \log \lambda_{ij}(t_{\ell_{ijk}}) - \int_{\tau_{ij}}^T \lambda_{ij}(t) dt \right\}, \quad (7)$$

where  $n_{ij}$  is the number of events observed on edge  $(i, j)$ . Explicit forms of the likelihood function (7) with  $d = 1$  and  $r = 1$  or  $r = \infty$ , are presented in the supplementary materials.

## 2.1. Computational Issues with the Calculation of the Likelihood

For  $r = \infty$ , the main computational burden associated with the calculation of the log-likelihood (7) is the double summation over each  $(i, j)$  pair of the sum of intensities  $\lambda_{ij}(t_{\ell_{ijk}})$  for the events  $t_{\ell_{ijk}}$ , and the summations required in (3) and (5) to evaluate that intensity for each event. This section discusses a recursive form of the log-likelihood (7) for the MEG model with  $r = \infty$ , which can be evaluated in linear time on each active edge, significantly reducing the computational requirements. This is a significant computational advantage of scaled exponential excitation functions, which makes them particularly appealing for practical applications. Assume sequences of arrival times  $t_{i1} < \dots < t_{iN_i(T)}$  involving  $i$  as source node, and  $t'_{j1} < \dots < t'_{jN'_j(T)}$  such that  $j$  is the destination of the connection. Within each pair of sequences, assume that a subset of  $n_{ij} \leq \min\{N_i(T), N'_j(T)\}$  events is observed on the edge  $(i, j)$ , and denote the indices of such events as  $u_{ij,1}, \dots, u_{ij,n_{ij}}$  and  $u'_{ij,1}, \dots, u'_{ij,n_{ij}}$ . The terms in the first summation in the log-likelihood (7) can then be written as

$$\begin{aligned} \log \lambda_{ij}(t_{\ell_{ijk}}) = & \log \left\{ \alpha_i + \mu_i \sum_{h=1}^{u_{ij,k}-1} e^{-(\mu_i + \phi_i)(t_{\ell_{ijk}} - t_{ih})} \right. \\ & + \beta_j + \mu'_j \sum_{h=1}^{u'_{ij,k}-1} e^{-(\mu'_j + \phi'_j)(t_{\ell_{ijk}} - t'_{jh})} + \boldsymbol{\gamma}_i^\top \boldsymbol{\gamma}_j \\ & \left. + \sum_{q=1}^d v_{iq} v'_{jq} \sum_{h=1}^{k-1} e^{-(v_{iq} + \theta_{iq})(v'_{jq} + \theta'_{jq})(t_{\ell_{ijk}} - t_{\ell_{ijh}})} \right\}. \end{aligned} \quad (8)$$

Using a technique similar to the method proposed in Ogata (1978), it is possible to calculate (8) in linear time using a recursive formulation of the inner summations. For  $k \in \{1, 2, \dots, n_{ij}\}$ , define  $\psi_{ij}(k)$ ,  $\psi'_{ij}(k)$  and  $\tilde{\psi}_{ijq}(k)$  as follows:

$$\begin{aligned} \psi_{ij}(k) &= \sum_{h=1}^{u_{ij,k}-1} e^{-(\mu_i + \phi_i)(t_{\ell_{ijk}} - t_{ih})}, \\ \psi'_{ij}(k) &= \sum_{h=1}^{u'_{ij,k}-1} e^{-(\mu'_j + \phi'_j)(t_{\ell_{ijk}} - t'_{jh})}, \\ \tilde{\psi}_{ijq}(k) &= \sum_{h=1}^{k-1} e^{-(v_{iq} + \theta_{iq})(v'_{jq} + \theta'_{jq})(t_{\ell_{ijk}} - t_{\ell_{ijh}})}, \end{aligned}$$

$$q = 1, \dots, d. \quad (9)$$

Using (8) and (9), the first term of the log-likelihood (7) becomes:

$$\sum_{k=1}^{n_{ij}} \log \lambda_{ij}(t_{\ell_k}) = \sum_{k=1}^{n_{ij}} \log \left\{ \alpha_i + \beta_j + \gamma_i \gamma'_j + \mu_i \psi_{ij}(k) + \mu'_j \psi'_{ij}(k) + \sum_{q=1}^d v_{iq} v'_{jq} \tilde{\psi}_{ijq}(k) \right\}. \quad (10)$$

The expression can be evaluated in linear time using the recursive equations for  $\psi_{ij}(k)$ ,  $\psi'_{ij}(k)$  and  $\tilde{\psi}_{ijq}(k)$  presented in the following proposition, proved in the supplementary materials.

*Proposition 1.* The terms  $\psi_{ij}(k)$ ,  $\psi'_{ij}(k)$  and  $\tilde{\psi}_{ijq}(k)$  can be written recursively as follows:

$$\begin{aligned} \psi_{ij}(k) &= e^{-(\mu_i + \phi_i)(t_{\ell_{ijk}} - t_{\ell_{ij,k-1}})} [1 + \psi_{ij}(k-1)] \\ &\quad + \sum_{h=u_{ij,k-1}+1}^{u_{ij,k}-1} e^{-(\mu_i + \phi_i)(t_{\ell_{ijk}} - t_{ih})}, \\ \psi'_{ij}(k) &= e^{-(\mu'_j + \phi'_j)(t'_{\ell'_k} - t'_{\ell'_{k-1}})} [1 + \psi'_{ij}(k-1)] \\ &\quad + \sum_{h=u'_{ij,k-1}+1}^{u'_{ij,k}-1} e^{-(\mu'_j + \phi'_j)(t_{\ell_{ijk}} - t'_{jh})}, \\ \tilde{\psi}_{ijq}(k) &= e^{-(v_{iq} + \theta_{iq})(v'_{jq} + \theta'_{jq})(t_{\ell_{ijk}} - t_{\ell_{ij,k-1}})} [1 + \tilde{\psi}_{ijq}(k-1)]. \end{aligned}$$

## 2.2. Extension to Undirected and Bipartite Graphs

The proposed modeling framework has been presented for directed graphs, but could be easily extended to undirected and bipartite networks. For undirected graphs  $\mathbf{A} = \mathbf{A}^\top$ , hence, there is no distinction between source and destination nodes. Therefore,  $\beta_j(t)$  in (2) could be simply be replaced by  $\alpha_j(t)$ , and  $\gamma_{ij}(t)$  modified as follows:

$$\gamma_{ij}(t) = \boldsymbol{\gamma}_i^\top \boldsymbol{\gamma}_j + \sum_{k > N_{ij}(t) - r}^{N_{ij}(t)} \sum_{\ell=1}^d v_{i\ell} v_{j\ell} \exp\{-(\theta_{i\ell} + v_{i\ell})(\theta_{j\ell} + v_{j\ell})t\}.$$

Furthermore, bipartite graphs can be considered as special cases of directed graphs, where the node set  $V = V_1 \cup V_2$  is divided into two sets  $V_1$  and  $V_2$  of cardinality  $n_1$  and  $n_2$ , such that  $V_1 \cap V_2 = \emptyset$ , and all the edges are of the form  $(i, j)$  with  $i \in V_1$  and  $j \in V_2$ . Therefore, the intensity function (2) and the corresponding components (3) and (5) still hold.

## 3. Inference via Maximum Likelihood Estimation

Inference in Hawkes processes is usually carried out using maximum likelihood estimation (MLE) via the EM algorithm or gradient ascent methods, since it is not possible to optimize the likelihood analytically. Similar issues arise for the log-likelihood (7) for MEG models. Only a small subset of the parameters has a closed-form solution for the MLE: the start times  $\tau_{ij}$ . If  $\tau_{ij}$  in

(7) is unknown, the maximum likelihood estimates is simply  $\hat{\tau}_{ij} = t_{\ell_{ij}}$  if at least one event is observed on the edge, and  $\hat{\tau}_{ij} = \infty$  otherwise. Intuitively, this is reasonable: the best guess about the start time of activity on an edge simply corresponds to the first observation on that edge. A formal proof of the result is provided in the supplementary materials. For maximizing (7) with respect to the remaining parameters  $\Psi$ , two strategies are deployed: the Expectation-Maximization algorithm (EM, Dempster, Laird, and Rubin 1977), and the adaptive moment estimation method (Adam, Kingma and Ba 2015).

Note that issues with MLE might arise when the parameters lie at the boundaries of the parameter space. For example, for a nonnegative finite jump  $0 < \mu_i < \infty$ , an infinitely fast decaying rate  $\phi_i \rightarrow \infty$  would make the resulting process a simple Poisson process for  $\alpha_i(t)$  in (2). Similarly, if  $0 < \phi_i < \infty$ , a jump size  $\mu_i \rightarrow 0$  would make  $\alpha_i(t)$  again correspond to a Poisson process with rate  $\alpha_i$ . Similar considerations can be made about the parameters of the excitation functions of the remaining components in (2). Additionally, identifiability issues are observed if further constraints are not imposed on the parameters: for example, subtracting a constant  $c \in [0, \min\{\alpha_1, \dots, \alpha_n\}]$  for all  $\alpha_i$ , and adding the *same* constant to all  $\beta_j$ , returns the same log-likelihood function (7). For identifiability, at least one value of  $\alpha_i$  or  $\beta_j$  must be kept fixed. Identifiability issues also arise from the interaction term: the inner product  $\mathbf{y}_i^T \mathbf{y}'_j$  is invariant to orthogonal transformations of  $\mathbf{y}_i$  and  $\mathbf{y}'_j$  preserving nonnegativity of the vectors. Similarly, for a constant  $c \in \mathbb{R}_+$ , the interaction parameters  $\mathbf{v}_i, \boldsymbol{\theta}_i, \mathbf{v}'_j$  and  $\boldsymbol{\theta}'_j$  produce the same excitation function as  $c\mathbf{v}_i, c\boldsymbol{\theta}_i, \mathbf{v}'_j/c$  and  $\boldsymbol{\theta}'_j/c$ . This leads to highly multimodal log-likelihood functions, and to a nonunique MLE. This problem is inconsequential for prediction, since the predictive distribution of new events depends upon a function of the parameters which is identifiable. Similarly, for assessing robustness of parameter estimation procedures, identifiable transformations of the parameters can be considered: for example, the sums  $\alpha_i + \beta_j$  in the main effects model are identifiable, or the products  $(v_i + \theta_i)(v'_j + \theta'_j)$  for  $d = 1$ . An example will be given in Section 5.1.

### 3.1. Inference via the EM Algorithm

An EM algorithm can be conveniently implemented following a network-wide extension of the procedure of Fox et al. (2016), after adopting a simple reparameterization of the log-likelihood (7). In particular, the scaled exponential decay rates  $\mu_i + \phi_i, \mu'_j + \phi'_j, v_{iq} + \theta_{iq}$ , and  $v'_{jq} + \theta'_{jq}$  are rewritten as  $\tilde{\phi}_i, \tilde{\phi}'_j, \tilde{\theta}_{iq}$ , and  $\tilde{\theta}'_{jq}$ , where  $\tilde{\phi}_i > \mu_i, \tilde{\phi}'_j > \mu'_j, \tilde{\theta}_{iq} > v_i$ , and  $\tilde{\theta}'_{jq} > v'_j$ . Similarly, the jumps  $\mu_i, \mu'_j, v_{iq}$ , and  $v'_{jq}$  are expressed as the product between the decay rates  $\tilde{\phi}_i, \tilde{\phi}'_j, \tilde{\theta}_{iq}$ , and  $\tilde{\theta}'_{jq}$  and the ratios between the jump and decay rates, denoted:

$$\begin{aligned} \tilde{\mu}_i &= \frac{\mu_i}{\mu_i + \phi_i}, & \tilde{\mu}'_j &= \frac{\mu'_j}{\mu'_j + \phi'_j}, \\ \tilde{v}_{iq} &= \frac{v_{iq}}{v_{iq} + \theta_{iq}}, & \tilde{v}'_{jq} &= \frac{v'_{jq}}{v'_{jq} + \theta'_{jq}}, \end{aligned}$$

where such parameters lie in  $[0, 1]$ . For example, under the two equivalent parameterizations,  $\omega_i(t) = \mu_i \exp\{-(\mu_i + \phi_i)t\} = \tilde{\mu}_i \tilde{\phi}_i \exp\{-\tilde{\phi}_i t\}$ . The vector of all parameters can then be equivalently rewritten as  $\tilde{\Psi} = (\boldsymbol{\alpha}, \tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\phi}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\mu}}', \tilde{\boldsymbol{\phi}}', \boldsymbol{\gamma}, \tilde{\mathbf{v}}, \tilde{\boldsymbol{\theta}}, \boldsymbol{\gamma}', \tilde{\mathbf{v}}', \tilde{\boldsymbol{\theta}}')$ , using the updated notation. Furthermore, consider the sequence of arrival times  $t_{i1} < \dots < t_{iN_i(T)}$  involving  $i$  as source node, and  $t'_{j1} < \dots < t'_{jN'_j(T)}$  such that  $j$  is the destination of the connection. Similarly, let the sequence  $t_{ij1} < \dots < t_{ijN_{ij}(T)}$  denote the events on the edge  $(i, j)$ . Using this revised notation, the conditional intensity function (2) for an edge, for  $t \geq \tau_{ij}$ , is:

$$\begin{aligned} \lambda_{ij}(t) &= \alpha_i + \sum_{k > N_i(t)-r}^{N_i(t)} \omega_i(t - t_{ik}) + \beta_j + \sum_{k > N'_j(t)-r}^{N'_j(t)} \omega'_j(t - t'_{jk}) \\ &+ \sum_{q=1}^d \gamma_{iq} \gamma'_{jq} + \sum_{k > N_{ij}(t)-r}^{N_{ij}(t)} \sum_{q=1}^d \omega_{ijq}(t - t_{ijk}), \end{aligned} \quad (11)$$

where the excitation function  $\omega_{ij}(\cdot)$  in (5) has been expressed as a sum of  $d$  functions  $\omega_{ijq} : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ , where  $\omega_{ijq}(t) = \tilde{v}_{iq} \tilde{\theta}_{iq} \tilde{v}'_{jq} \tilde{\theta}'_{jq} \exp\{-\tilde{\theta}_{iq} \tilde{\theta}'_{jq} t\}$  from (6). Therefore, conditional on  $t$ , the subsequent event on the edge  $(i, j)$  could be interpreted as the offspring of one of the  $2 + d + \min\{r, N_i(t)\} + \min\{r, N'_j(t)\} + d \min\{r, N_{ij}(t)\}$  components of the intensity (11), each corresponding to a nonhomogeneous Poisson process in  $(t, \infty)$ . In other words,  $\lambda_{ij}(t)$  is written as a superimposition of conditional intensities of different processes, where the event allocations are missing data, giving a *branching structure* to the event hierarchy.

For missing data problems, the traditional approach in statistics is to deploy the EM algorithm, which in this setting requires to introduce latent binary variables to reconstruct the branching structure. For events generated from the background rates  $\alpha_i, \beta_j$  and  $\gamma_{iq} \gamma'_{jq}$ ,  $q = 1, \dots, d$  (also known as *immigrant events* in the literature), the corresponding latent variables are denoted by the letter  $b$ . In particular  $b_{ij\ell}^{(\alpha)} \in \{0, 1\}$  equals 1 if  $t_{ij\ell}$  is a background event obtained from the Poisson process with rate  $\alpha_i$ , and 0 otherwise. Similarly,  $b_{ij\ell}^{(\beta)}$  and  $b_{ij\ell}^{(\gamma)}$  denote whether the event  $t_{ij\ell}$  is a background event from Poisson processes with rates  $\beta_j$  and  $\gamma_{iq} \gamma'_{jq}$  respectively. On the other hand, for the events that are not generated from the background rates, the corresponding latent variables are denoted with the letter  $z$ . In particular,  $z_{ij\ell k}^{(\alpha)} = 1$  if  $t_{ij\ell}$  is offspring of the  $k$ th event such that node  $i$  is source, and 0 otherwise; a similar reasoning applies for  $z_{ij\ell k}^{(\beta)}$ , which instead considers the sequence of events such that node  $j$  is destination. As before, it is necessary to introduce a further subscript for the interaction term:  $z_{ij\ell k}^{(\gamma)} = 1$  if  $t_{ij\ell}$  is offspring of the  $k$ th event on the edge  $(i, j)$ , from the  $q$ th additive component of the intensity. If such latent variables are known, it is possible to write in simple form the *complete data* log-likelihood, which also includes the information about the branching structure:

$$\begin{aligned} \log L(\mathcal{H}_T; \tilde{\Psi}, \mathbf{B}, \mathbf{Z}) &= \sum_{i=1}^n \sum_{j=1}^n \left\{ \sum_{\ell=1}^{n_{ij}} \left[ b_{ij\ell}^{(\alpha)} \log(\alpha_i) \right. \right. \\ &+ \sum_{k > N_i(t_{ij\ell})-r}^{N_i(t_{ij\ell})} z_{ij\ell k}^{(\alpha)} [\log(\tilde{\mu}_i \tilde{\phi}_i) - \tilde{\phi}_i(t_{ij\ell} - t_{ik})] + b_{ij\ell}^{(\beta)} \log(\beta_j) \end{aligned}$$

$$\begin{aligned}
& + \sum_{k > N'_j(t_{ij\ell}) - r}^{N'_j(t_{ij\ell})} z_{ij\ell k}^{(\beta)} [\log(\tilde{\mu}'_j \tilde{\phi}'_j) - \tilde{\phi}'_j(t_{ij\ell} - t'_{jk})] \\
& + \sum_{q=1}^d \left( b_{ij\ell q}^{(\gamma)} [\log(\gamma_{iq}) + \log(\gamma'_{jq})] \right. \\
& + \sum_{k > N_{ij}(t_{ij\ell}) - r}^{N_{ij}(t_{ij\ell})} z_{ij\ell kq}^{(\gamma)} [\log(\tilde{v}_{iq} \tilde{\theta}'_{iq}) + \log(\tilde{v}'_{jq} \tilde{\theta}'_{jq}) \\
& \left. - \tilde{\theta}'_{iq} \tilde{\theta}'_{jq}(t_{ij\ell} - t_{ijk})] \right) - \int_{\tau_{ij}}^T \lambda_{ij}(t) dt. \tag{12}
\end{aligned}$$

The E-step of the EM algorithm consists in calculating  $\mathbb{E}_{\mathbf{B}, \mathbf{Z} | \mathcal{H}_T, \tilde{\Psi}^*} \{\log L(\mathcal{H}_T; \tilde{\Psi}, \mathbf{B}, \mathbf{Z})\}$ , the expected value of the complete data log-likelihood (12) with respect to the distribution of the latent indicators  $\mathbf{B}$  and  $\mathbf{Z}$ , conditional on the observations  $\mathcal{H}_T$  and parameter values  $\tilde{\Psi}^*$ . From (12), this reduces to calculating:

$$\begin{aligned}
\xi^{(\cdot)} &= \mathbb{P}_{\mathbf{B}, \mathbf{Z} | \mathcal{H}_T, \tilde{\Psi}^*} \left\{ b^{(\cdot)} = 1 \mid \tilde{\Psi}^* \right\}, \\
\zeta^{(\cdot)} &= \mathbb{P}_{\mathbf{B}, \mathbf{Z} | \mathcal{H}_T, \tilde{\Psi}^*} \left\{ z^{(\cdot)} = 1 \mid \tilde{\Psi}^* \right\},
\end{aligned}$$

known as *responsibilities*. Such probabilities are simply represented by the relative contributions of different components to the conditional intensity (11):

$$\begin{aligned}
\xi_{ij\ell}^{(\alpha)} &\propto \alpha_i, & \zeta_{ij\ell k}^{(\alpha)} &\propto \tilde{\mu}_i \tilde{\phi}_i \exp\{-\tilde{\phi}_i(t_{ij\ell} - t_{ik})\} \mathbb{1}_{(t_{ik}, \infty)}(t_{ij\ell}), \\
\xi_{ij\ell}^{(\beta)} &\propto \beta_j, & \zeta_{ij\ell k}^{(\beta)} &\propto \tilde{\mu}'_j \tilde{\phi}'_j \exp\{-\tilde{\phi}'_j(t_{ij\ell} - t'_{jk})\} \mathbb{1}_{(t'_{jk}, \infty)}(t_{ij\ell}), \\
\xi_{ij\ell q}^{(\gamma)} &\propto \gamma_{iq} \gamma'_{jq}, & \zeta_{ij\ell kq}^{(\gamma)} &\propto \tilde{v}_{iq} \tilde{\theta}_{iq} \tilde{v}'_{jq} \tilde{\theta}'_{jq} \exp\{-\tilde{\theta}_{iq} \tilde{\theta}'_{jq}(t_{ij\ell} - t_{ijk})\} \\
& & & \mathbb{1}_{(t_{ijk}, \infty)}(t_{ij\ell}), \tag{13}
\end{aligned}$$

with normalizing constant  $\lambda_{ij}(t_{ij\ell})$ , see (2) and (11), calculated using parameter values  $\tilde{\Psi}^*$ .

At the M-step, the expectation  $\mathbb{E}_{\mathbf{B}, \mathbf{Z} | \mathcal{H}_T, \tilde{\Psi}^*} \{\log L(\mathcal{H}_T; \tilde{\Psi}, \mathbf{B}, \mathbf{Z})\}$  calculated at the E-step is maximized with respect to  $\tilde{\Psi}$ , and updated parameter estimates are obtained. For most of the parameters in the MEG model with scaled exponential excitation function, the maxima are analytically available, and their form depends on the choice of  $r$ . For  $r = \infty$ :

$$\begin{aligned}
\hat{\alpha}_i &= \frac{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \xi_{ij\ell}^{(\alpha)}}{\sum_{j=1}^n (T - \min\{T, \tau_{ij}\})}, \\
\hat{\mu}_i &= \frac{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \sum_{k=1}^{N_i(t_{ij\ell})} \zeta_{ij\ell k}^{(\alpha)}}{\sum_{j=1}^n \sum_{k=1}^{n_i} [e^{-\tilde{\phi}_i \min\{T, \max\{\tau_{ij} - t_{ik}, 0\}} - e^{-\tilde{\phi}_i(T - t_{ik})}], \\
\hat{\beta}_j &= \frac{\sum_{i=1}^n \sum_{\ell=1}^{n_{ij}} \xi_{ij\ell}^{(\beta)}}{\sum_{i=1}^n (T - \min\{T, \tau_{ij}\})}, \\
\hat{\mu}'_j &= \frac{\sum_{i=1}^n \sum_{\ell=1}^{n_{ij}} \sum_{k=1}^{N'_j(t_{ij\ell})} \zeta_{ij\ell k}^{(\beta)}}{\sum_{i=1}^n \sum_{k=1}^{n'_j} [e^{-\tilde{\phi}'_j \min\{T, \max\{\tau_{ij} - t'_{jk}, 0\}} - e^{-\tilde{\phi}'_j(T - t'_{jk})}], \\
\hat{\gamma}_{iq} &= \frac{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \xi_{ij\ell q}^{(\gamma)}}{\sum_{j=1}^n \gamma'_{jq} (T - \min\{T, \tau_{ij}\})},
\end{aligned}$$

$$\hat{v}_{iq} = \frac{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \sum_{k=1}^{N_{ij}(t_{ij\ell})} \zeta_{ij\ell kq}^{(\gamma)}}{\sum_{j=1}^n \tilde{v}'_{jq} \sum_{k=1}^{n_{ij}} [1 - e^{-\tilde{\theta}_{iq} \tilde{\theta}'_{jq}(T - t_{ijk})}], \tag{14}$$

and similarly for  $\hat{\gamma}'_{jq}$  and  $\hat{v}'_{jq}$ . For the remaining parameters, an exact solution is not available, but recursive equations can be obtained. For example, again for  $r = \infty$ :

$$\begin{aligned}
\tilde{\phi}_i &= \frac{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \sum_{k=1}^{N_i(t_{ij\ell})} \zeta_{ij\ell k}^{(\alpha)}}{\sum_{j=1}^n \left\{ \sum_{\ell=1}^{n_{ij}} \sum_{k=1}^{N_i(t_{ij\ell})} \zeta_{ij\ell k}^{(\alpha)} (t_{ij\ell} - t_{ik}) \right. \\
& \quad \left. + \tilde{\mu}_i \sum_{k=1}^{n_i} [(T - t_{ik}) e^{-\tilde{\phi}_i(T - t_{ik})} - \tau_{ijk}^+ e^{-\tilde{\phi}_i \tau_{ijk}^+}] \right\}}, \\
\tilde{\theta}_{iq} &= \frac{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \sum_{k=1}^{N_{ij}(t_{ij\ell})} \zeta_{ij\ell kq}^{(\gamma)}}{\sum_{j=1}^n \sum_{\ell=1}^{n_{ij}} \left\{ \sum_{k=1}^{N_{ij}(t_{ij\ell})} \zeta_{ij\ell kq}^{(\gamma)} \tilde{\theta}'_{jq} (t_{ij\ell} - t_{ijk}) \right. \\
& \quad \left. + \tilde{v}_{iq} \tilde{v}'_{jq} \tilde{\theta}'_{jq} (T - t_{ij\ell}) e^{-\tilde{\theta}_{iq} \tilde{\theta}'_{jq}(T - t_{ij\ell})} \right\}}, \tag{15}
\end{aligned}$$

where  $\tau_{ijk}^+ = \min\{T, \max\{\tau_{ij} - t_{ik}, 0\}\}$ . Similar equations are available for  $\tilde{\phi}'_j$  and  $\tilde{\theta}'_{jq}$ . The full iterative procedure is summarized in Algorithm 1.

---

**Algorithm 1:** EM algorithm for optimization of the log-likelihood (7).

---

**Input:** initial parameter values  $\tilde{\Psi}_0$ .

**Output:** model parameters  $\tilde{\Psi}$  corresponding to a local maximum of  $\log L(\mathcal{H}_T; \tilde{\Psi})$ .

- 1 **for**  $k = 1, 2, \dots$  **do**
  - 2     E-step: calculate responsibilities  $\xi^{(\cdot)}$  and  $\zeta^{(\cdot)}$  using (13) with parameters  $\tilde{\Psi}_k$ ,
  - 3     M-step: calculate  $\tilde{\Psi}_{k+1} = \operatorname{argmax}_{\tilde{\Psi}} \mathbb{E}_{\mathbf{B}, \mathbf{Z} | \mathcal{H}_T, \tilde{\Psi}_k} \{\log L(\mathcal{H}_T; \tilde{\Psi}, \mathbf{B}, \mathbf{Z})\}$ ; for  $r = \infty$ , apply (14) and (15) iteratively, using the *most recent* parameter estimates,
  - 4 **until convergence** in  $\log L(\mathcal{H}_T; \tilde{\Psi})$ .
- 

### 3.2. Inference via Gradient Ascent Methods

The EM algorithm proposed in the previous section has appealing statistical properties, but it is not scalable for large networks or for large numbers of events, since it requires  $n_{ij}[2 + d + N_i(T) + N'_j(T) + dn_{ij}]$  additional latent variables to be defined for each edge, which is not feasible in most practical applications. On the other hand, the log-likelihood in (7) was shown to have a recursive expression for  $r = \infty$ , which also holds for its gradient with respect to the parameters  $\Psi$ . Therefore, in order to make the inferential procedure scalable, gradient-based optimization methods appear to be suitable. Gradient ascent methods are usually based on computing the gradient of the log-likelihood function, and iteratively updating the parameter values in the direction of steepest ascent given by the gradient, for a given step size  $\eta \in \mathbb{R}_+$ , also known as *learning rate*. One of the main issues of standard gradient ascent for high-dimensional parameter estimation is the choice of the learning rate. The adaptive moment estimation method (Adam, Kingma and Ba 2015) is a popular gradient ascent optimization algorithm widely used

in the machine learning and deep learning communities, which adaptively selects and adjusts learning rates for each parameter. Its convergence properties have been extensively studied (Reddi, Kale, and Kumar 2018; Chen et al. 2019; Zou et al. 2019). In *Adam*, the step sizes are adjusted via exponentially weighted moving averages (EWMA) of the estimated gradient and square gradient (respectively denoted  $\mathbf{m}$  and  $\mathbf{v}$ , with decay rates  $\rho_1, \rho_2 \in [0, 1]$ ). Such averages provide estimates for the first and second moment of the gradient respectively; these estimates are then corrected for bias and used to update the parameters in a similar fashion to standard gradient ascent, after adding a small offset  $\varepsilon \in \mathbb{R}_+$  (usually known as smoothing parameter) to the estimate of the second moment, in order to avoid computational issues when its value vanishes toward zero. Considering the high-dimensional maximum likelihood estimation of MEG models, Adam appears to be a suitable inferential choice. Alternative gradient ascent techniques for optimization are surveyed in Ruder (2016). In this work, Adam is implemented after adopting a simple reparameterization and optimizing the logarithm of each parameter, since are all constrained to be positive. The resulting optimization procedure is detailed in Algorithm 2. The gradient  $\mathbf{g} = \frac{\partial}{\partial \Psi} \log L(\mathcal{H}_T; \Psi)$  of the likelihood (7) with respect to  $\Psi$  inherits a recursive form from (10), and therefore it can be calculated in linear time. Explicit forms for  $d = 1$  and  $r = \infty$  are derived in the supplementary materials.

---

**Algorithm 2:** Adam algorithm for optimization of the log-likelihood (7).

---

**Input:** step size  $\eta \in \mathbb{R}_+$ , decay rates  $\rho_1, \rho_2 \in (0, 1)$ , smoothing parameter  $\varepsilon \in \mathbb{R}_+$ , initial parameter values  $\Psi_0$ .

**Output:** model parameters  $\Psi$  corresponding to a local maximum of  $\log L(\mathcal{H}_T; \Psi)$ .

1 Initialize estimates of the first and second moment of the gradient:  $\mathbf{m}_0 = \mathbf{0}, \mathbf{v}_0 = \mathbf{0}$ ,

2 **for**  $k = 1, 2, \dots$  **do**

3 calculate gradient  $\mathbf{g}_k = \frac{\partial}{\partial \Psi} \log L(\mathcal{H}_T; \Psi) \Big|_{\Psi = \Psi_{k-1}}$ ,  
evaluated at  $\Psi_{k-1}$ ,

4 update EWMA estimate of first moment:

$$\mathbf{m}_k = \rho_1 \mathbf{m}_{k-1} + (1 - \rho_1)(\mathbf{g}_k \times \Psi_{k-1}),$$

5 update second moment:

$$\mathbf{v}_k = \rho_2 \mathbf{v}_{k-1} + (1 - \rho_2)[(\mathbf{g}_k \times \Psi_{k-1}) \times (\mathbf{g}_k \times \Psi_{k-1})],$$

6 update parameters:  $\Psi_k =$

$$\Psi_{k-1} \times \exp \left\{ \eta \mathbf{m}_k / (1 - \rho_1^k) \left( \sqrt{\mathbf{v}_k / (1 - \rho_2^k)} + \varepsilon \right) \right\},$$

7 **until** convergence in  $\log L(\mathcal{H}_T; \Psi)$ .

*Sums, products, quotients, exponentials, and square roots are applied element-wise.*

---

#### 4. Simulation and Assessment of the Goodness-of-Fit

In order to validate the inferential procedure, it is necessary to simulate data from the MEG model (2), which can be interpreted as an extended multivariate Hawkes process where some of the parameters are shared across the individual processes.

Therefore, simulating MEG models is possible under the framework described in Ogata (1981), and follows the standard technique of simulation via *thinning*. The procedure is described in Algorithm 3.

---

**Algorithm 3:** Simulation of a MEG in  $[0, T]$ .

---

1 set  $t^* = 0$ ,

2 **repeat**

3 set  $\lambda^* = \sum_{i=1}^n \sum_{j=1}^n \lambda_{ij}(t_+^*)$ , where  $t_+^*$  denotes the limit from the right,

4 generate the inter-arrival time  $q = -\log(u)/\lambda^*$ , where  $u \sim \text{Uniform}[0, 1]$ ,

5 obtain the candidate arrival time  $t^* \leftarrow t^* + q$ ,

6 assign the arrival time  $t^*$  to the edge  $(i, j)$  with probability  $\lambda_{ij}(t^*)/\lambda^*$ , and do not assign to any edge with probability  $1 - \sum_{i=1}^n \sum_{j=1}^n \lambda_{ij}(t^*)/\lambda^*$ , where  $t_-^*$  denotes the limit from the left.

7 **until**  $t^* > T$ ;

---

Furthermore, it is possible to assess the performance of the inferential procedure by evaluating the goodness-of-fit from out-of-sample events. If the model parameters are estimated only from the event times obtained in  $[0, T^*]$ , with  $T^* < T$ , using Algorithm 2, the goodness-of-fit can then be evaluated from the event times in  $(T^*, T]$ . Goodness-of-fit measures can be calculated from functions of the compensator function for the model. Given the conditional intensity  $\lambda_{ij}(t)$ , the compensator  $\Lambda_{ij}(t)$  is:

$$\Lambda_{ij}(t) = \int_{\tau_{ij}}^t \lambda_{ij}(s) ds.$$

Examples of compensator functions for some MEG models, for  $t = T$ , can be found in the Supplementary Material. Given arrival times  $t_1, \dots, t_{n_{ij}}$  on the edge  $(i, j)$ , under the null hypothesis of correct specification of the conditional intensity  $\lambda_{ij}(t)$ , by time rescaling theorem (Brown et al. see, for example, 2002)  $\Lambda_{ij}(t_1), \dots, \Lambda_{ij}(t_{n_{ij}})$  are event times of a homogeneous Poisson process with unit rate. It follows that the upper tail  $p$ -values

$$\begin{aligned} p_{ijk} &= \exp\{-\Lambda_{ij}(t_k) + \Lambda_{ij}(t_{k-1})\} \\ &= \exp\left\{-\int_{t_{k-1}}^{t_k} \lambda_{ij}(s) ds\right\} \end{aligned} \quad (16)$$

follow a standard uniform distribution under the null hypothesis. Therefore, given the estimates of the conditional intensity functions obtained from the arrival times in  $[0, T^*]$ , approximately uniform  $p$ -values for the test event times in  $(T^*, T]$  should be observed if the model is specified and estimated correctly.

#### 5. Applications and Results

In this section, the MEG model is tested on simulated network data and on two real world computer network datasets: the Enron e-mail network, and a bipartite graph obtained from network flow data collected at Imperial College London. Across the experiments, the decay rates  $(\rho_1, \rho_2)$  in Algorithm 2 have been set to  $(0.9, 0.99)$ , and  $\varepsilon = 10^{-8}$ .



### 5.1. Simulated Events on Small Fully Connected Graphs

In order to evaluate [Algorithms 1](#) and [2](#) and their performance at estimating MEG models, simulated network data are initially used. In this section, a small fully connected directed graph with  $n = 2$  is considered. Two types of mutually exciting graphs with  $r = \infty$  and  $\tau_{ij} = 0$  are generated:

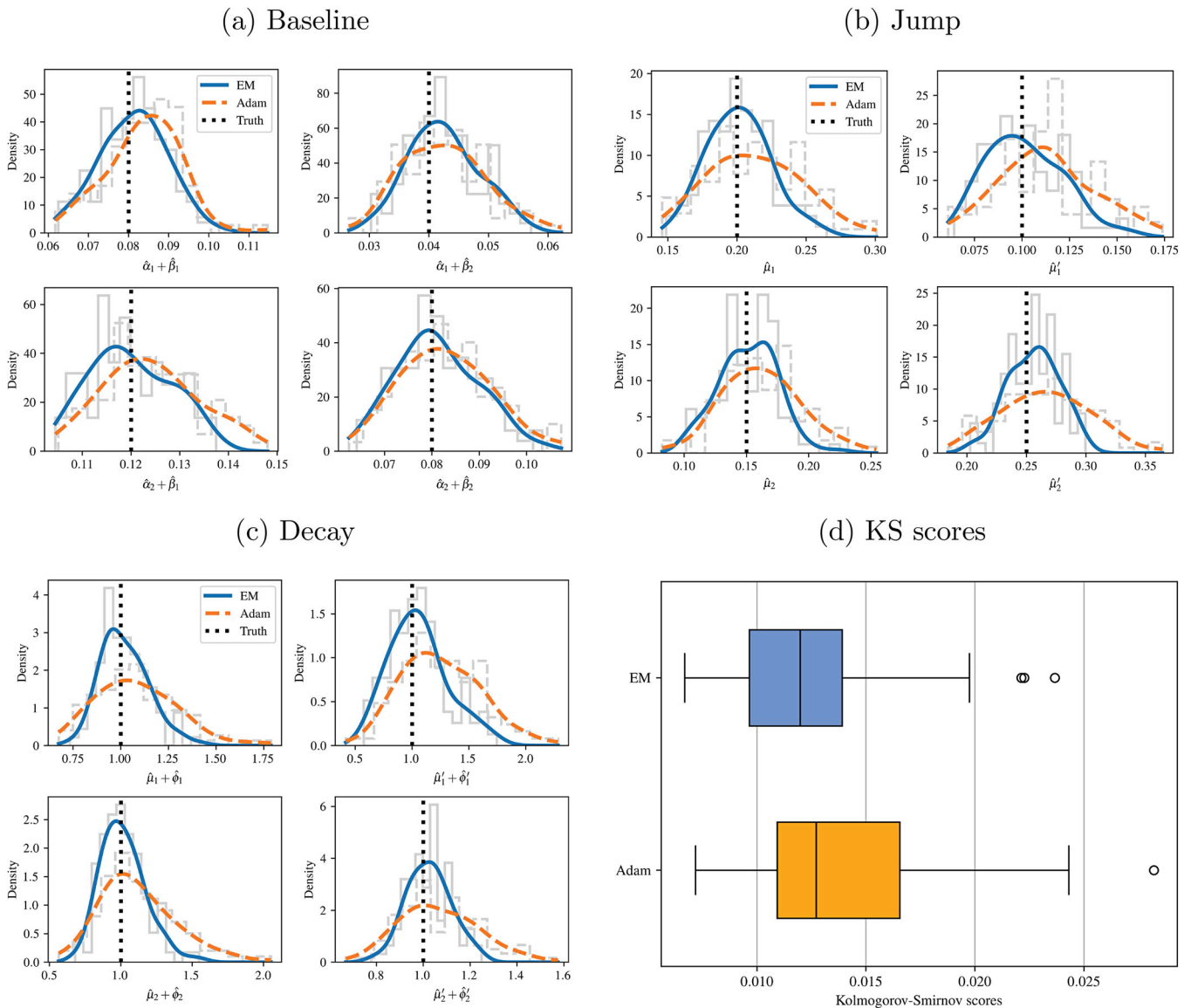
- MEGs with  $\lambda_{ij}(t) = \alpha_i(t) + \beta_j(t)$ ,  $\alpha_i(t)$  and  $\beta_j(t)$  as in [\(3\)](#) and [\(4\)](#), with  $\alpha = [0.01, 0.05]$ ,  $\beta = [0.07, 0.03]$ ,  $\mu = [0.2, 0.15]$ ,  $\mu' = [0.1, 0.25]$ ,  $\phi = [0.8, 0.85]$ ,  $\phi' = [0.9, 0.75]$ , and
- MEGs with  $\lambda_{ij}(t) = \gamma_{ij}(t)$ , see, [\(5\)](#),  $\gamma = [0.1, 0.5]$ ,  $\gamma' = [0.1, 0.3]$ ,  $\nu = [0.6, 0.4]$ ,  $\nu' = [0.5, 0.25]$ ,  $\theta = [0.4, 0.6]$ ,  $\theta' = [0.5, 0.75]$ .

For each of the two MEG models, 3000 events are simulated using [Algorithm 3](#), and the process parameters are estimated from the simulated events via EM ([Algorithm 1](#)) and Adam ([Algorithm 2](#), with  $\eta = 0.05$ ), initializing the parameters at

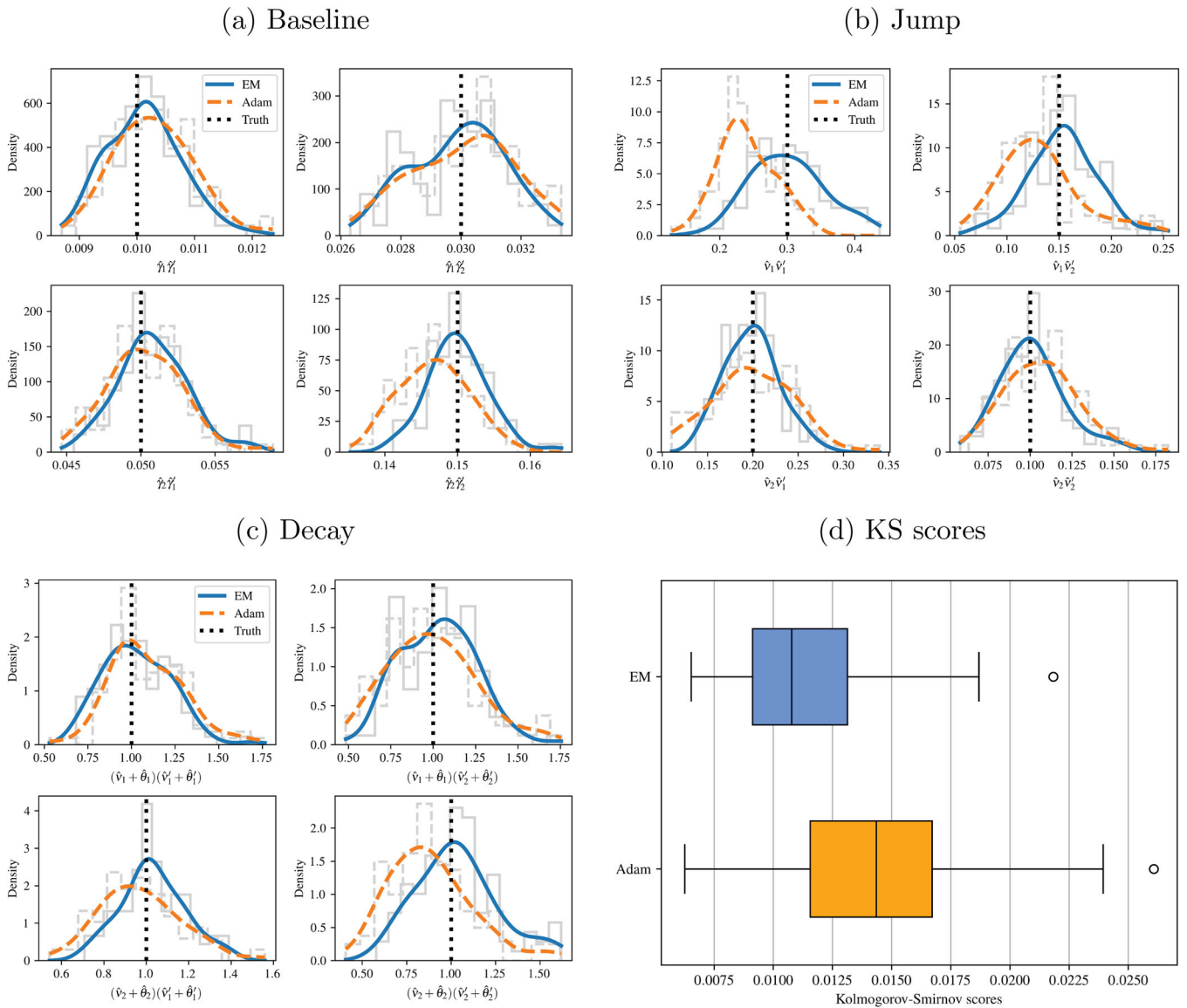
random from a uniform distribution in  $(0.1, 1)$ . The estimation procedure is repeated five times from different random initialization points, and the final estimates corresponding to the highest log-likelihood are retained. Using the estimated parameters, the  $p$ -values [\(16\)](#) are then calculated for all simulated events. Finally, the Kolmogorov-Smirnov (KS) score against the uniform distribution is calculated on those  $p$ -values. The procedure is then repeated 100 times, obtaining a set of parameter estimates for each simulated MEG.

The results are plotted in [Figures 2](#) and [3](#), which report the histograms and kernel density estimates of identifiable transformations of the parameters for each of the four network edges, obtained using the EM and Adam algorithms, compared to the true value of the parameters. Furthermore, the figures report the boxplots of the KS scores.

Overall, it appears that the results obtained using Adam are only marginally worse than those obtained using the EM algorithm. In particular, the distributions of estimates obtained



**Figure 2.** Histograms (with corresponding kernel density estimates) of parameter estimates and boxplots of KS scores obtained using EM and Adam from 100 simulations of 3000 events on a fully connected MEG with  $n = 2$ ,  $\lambda_{ij}(t) = \alpha_i(t) + \beta_j(t)$ ,  $r = \infty$ ,  $\alpha = [0.01, 0.05]$ ,  $\beta = [0.07, 0.03]$ ,  $\mu = [0.2, 0.15]$ ,  $\mu' = [0.1, 0.25]$ ,  $\phi = [0.8, 0.85]$ ,  $\phi' = [0.9, 0.75]$ .



**Figure 3.** Histograms (with corresponding kernel density estimates) of parameter estimates and boxplots of KS scores obtained using EM and Adam from 100 simulations of 3000 events on a fully connected MEG with  $n = 2$ ,  $\lambda_{ij}(t) = \gamma_{ij}(t)$ ,  $r = \infty$ ,  $\gamma = [0.1, 0.5]$ ,  $\gamma' = [0.1, 0.3]$ ,  $\nu = [0.6, 0.4]$ ,  $\nu' = [0.5, 0.25]$ ,  $\theta = [0.4, 0.6]$ ,  $\theta' = [0.5, 0.75]$ .

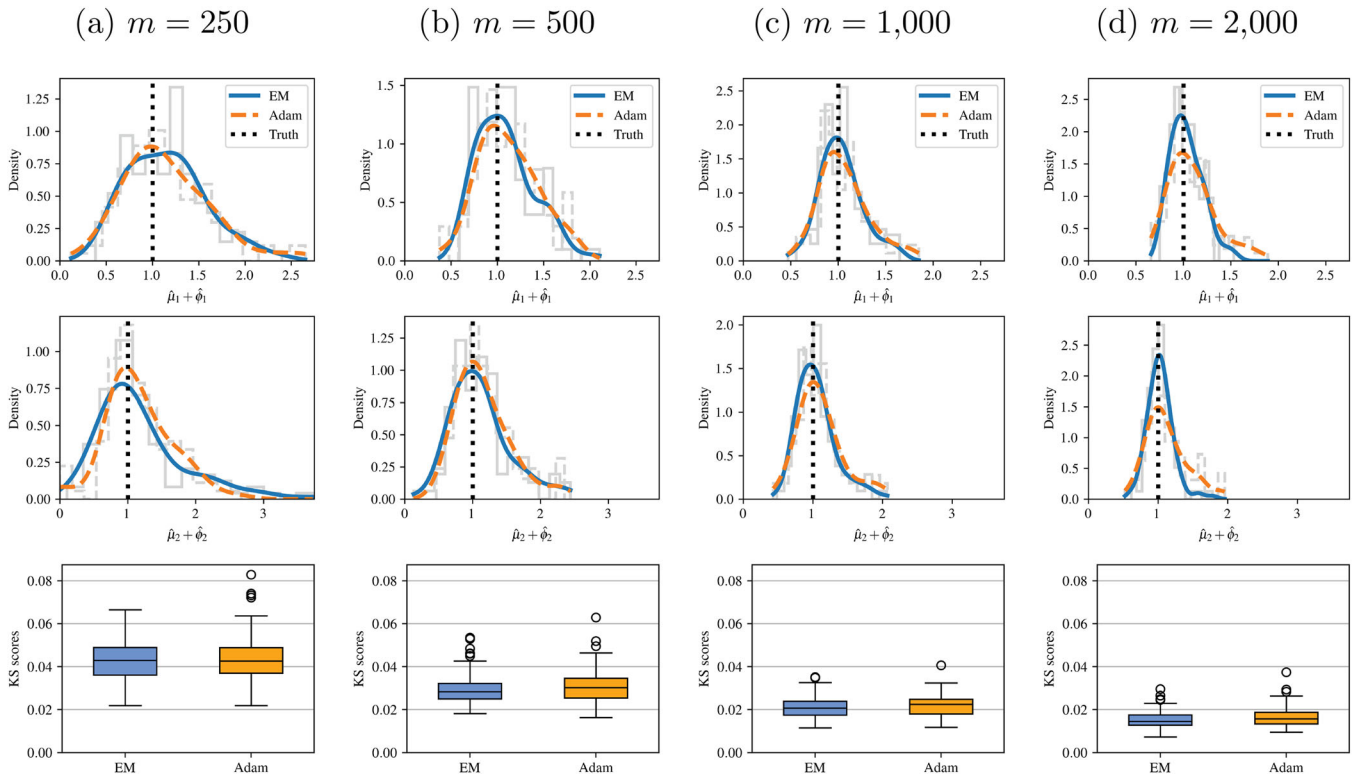
using the two methodologies appear to be roughly centered around the true value of the parameters, but the EM estimates appear to be slightly more precise and accurate when compared to Adam. In both cases, the KS scores are extremely small, demonstrating an excellent fit. Furthermore, it is possible to compare the performance of the two inferential algorithms when the number of events increases: Figure 4 reports the histograms of estimates of the decay  $\mu + \phi$  obtained using only 250, 500, 1000, and 2000 of the 3000 simulated events on each graph. The performance of both estimation procedures improves in terms of KS scores and variance of estimates when more observations are available.

The main advantage of Adam over the EM algorithm for  $r = \infty$  is mainly given by the recursive form of the log-likelihood, which enables calculations in linear time on each edge for the gradients. On the other hand, the EM algorithm requires a quadratic number of additional latent variables defined on each edge, which becomes unsustainable for efficient inference on large graphs. Therefore, for the remainder of this

work, Adam will be used, primarily because of computational reasons.

## 5.2. Simulated Events on Erdős-Rényi Graphs

In order to further evaluate estimation of MEG models, events on an Erdős-Rényi graph are also simulated. First, an adjacency matrix is simulated from an Erdős-Rényi graph with  $n = 10$  nodes, such that  $A_{ij} \sim \text{Bernoulli}(p)$ , with  $p = 1/4$ . For the edges such that  $A_{ij} = 1$ , then  $\tau_{ij} = 0$ , otherwise if  $A_{ij} = 0$ , then  $\tau_{ij} = \infty$ . Second,  $m = 2500$  event times are generated from a MEG model with  $r = \infty$  using Algorithm 3, with parameters in  $\Psi$  sampled at random from uniform distributions, restricted to the following ranges:  $\alpha_i, \beta_j \in (10^{-5}, 10^{-4})$ ,  $\mu_i, \mu'_j, \phi_i, \phi'_j \in (10^{-2}, 10^{-1})$ ,  $\gamma_{i\ell}, \gamma'_{j\ell} \in (10^{-5}, 10^{-1})$ ,  $\nu_{i\ell}, \nu'_{j\ell} \in (10^{-2}, 1)$ , and  $\theta_{i\ell} = 1 - \nu_{i\ell}$ ,  $\theta'_{j\ell} = 1 - \nu'_{j\ell}$ . In the simulation, the expected number of events per active edge is  $m/[pn(n-1)] \approx 111$ . Algorithm 2 is used to estimate  $2n \times 6 = 120$  parameters, with



**Figure 4.** Histograms (with corresponding kernel density estimates) of estimates for  $\mu + \phi$  and boxplots of KS scores obtained using EM and Adam from 100 simulations from the same model as Figure 2, with  $m \in \{250, 500, 1000, 2000\}$  events.

learning rate  $\eta = 0.1$ , after a random initialization from the same uniform distributions used in the data-generating process. The entire procedure is repeated 100 times.

A second simulation is conducted for an Erdős-Rényi graph with  $n = 20$  nodes and  $p = 1/4$ , simulating  $m = 10,000$  events from a MEG model with interaction term only, corresponding to  $\lambda_{ij}(t) = A_{ij}\gamma_{ij}(t)$ , with  $r = 1$  and  $d = 5$ . A minor modification is made to the range of the uniform distributions for sampling some of the interaction term parameters:  $\gamma_{i\ell}, \gamma'_{j\ell} \in (10^{-5}, 10^{-1})$ ,  $v_{i\ell}, v'_{j\ell}, \theta_{i\ell}, \theta'_{j\ell} \in (10^{-2}, 1)$ . Despite the simpler form of the intensity functions  $\lambda_{ij}(t)$ , more parameters must be estimated ( $2n \times 3d = 600$ ) compared to the first simulation, and the expected number of connections per edge is only 105. As before, 100 MEGs are generated, and Adam (Algorithm 2) is used to estimate the parameters, with learning rate  $\eta = 10^{-3}$ . The resulting boxplots of the KS test obtained for the two simulations are plotted in Figure 5. Both boxplots demonstrate that the algorithm is able to recover sensible estimates of the parameter values, resulting in small KS scores, corresponding to a good model fit.

### 5.3. Enron E-mail Network

The Enron e-mail network collection is a record of e-mails exchanged between the employees of Enron Corporation before its bankruptcy. These data have already been demonstrated to be well-modeled as self-exciting point processes by Fox et al. (2016). In this article, the version of these data<sup>1</sup> used in Priebe

et al. (2005) is analyzed, where e-mails recorded multiple times have been used only once, and e-mails with incorrectly recorded sent times (coded in the data with 9 p.m., December 31, 1979) have been removed. After such preprocessing, the e-mail data consist of 34,427 distinct triplets  $(x_k, y_k, t_k)$ , corresponding to messages exchanged between  $n = 184$  employees between November 1998 and June 2002, forming a total of 3007 graph edges. Note that some of the emails are sent to multiple receivers, and only 18,031 unique event times are observed, implying that on average each e-mail is sent to approximately 1.90 nodes.

Because an e-mail can have multiple recipients, and because the event times are recorded to the nearest second, the likelihood (7) must be adapted slightly to handle tied arrival times. An approach used by (Price-Williams and Heard 2020, sec. 8.1) is followed, with the arrivals modeled by an analogous discrete time process: In particular, arrivals at time  $t$  are assumed to contribute to the intensities  $\lambda_{ij}(\cdot)$  from time  $t + dt$  onwards, where  $dt$  is the sampling interval, equal to one second in this example. The  $p$ -values of the process are approximated using (16), following Fox et al. (2016).

The model is trained on 30,704 e-mails sent before December 01, 2001, and tested on the remaining 3723 e-mails. In the training set, 2720 edges are observed, and 811 in the test set, of which 287 are *not* observed in the training period. One of the advantages of the proposed methodology is the possibility to score events for such new links.

A range of MEG models are fitted to the training data, using different combinations of  $r$  and  $d$  for characterizing main effects and interactions. A good configuration for the initial parameter values is obtained through using the quantities  $u_i = \frac{N_i(T)}{nT}$  and

<sup>1</sup>The data are freely available at <http://www.cis.jhu.edu/~parky/Enron/>.

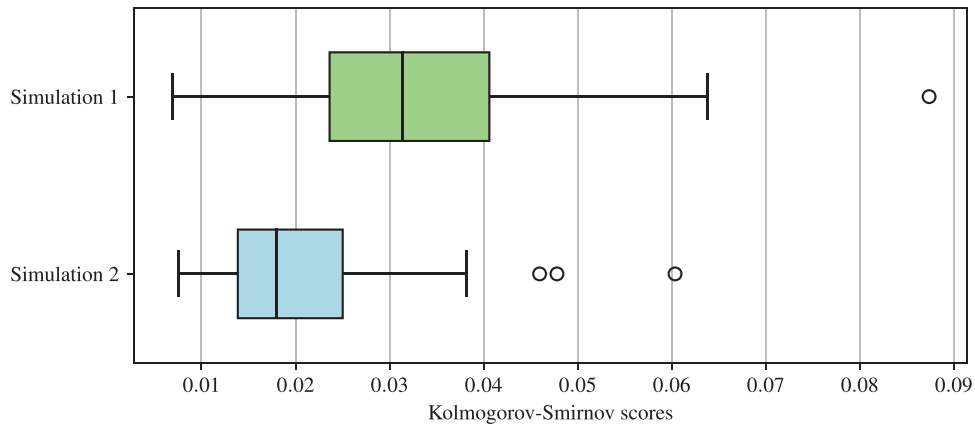


Figure 5. Boxplots of the Kolmogorov-Smirnov scores obtained for the two simulations described in Section 5.2.

$u'_j = \frac{N'_j(T)}{nT}$ , corresponding to the average rate of incoming and outgoing connections observed for each node. In particular, good results and convergence are obtained setting initial values  $\alpha_i = \mu_i = u_i$ ,  $\phi_i = 3u_i$ ,  $\beta_j = \mu'_j = u'_j$ , and  $\phi'_j = 3u'_j$ . For the interaction term, the initial values used to obtain the results are  $\gamma_{i\ell} = \gamma'_{j\ell} = v_{i\ell} = v'_{j\ell} = 10^{-4}$ , and  $\theta_{i\ell} = \theta'_{j\ell} = 5 \cdot 10^{-4}$ . If  $d > 1$ , then Gaussian noise with standard deviation  $2 \cdot 10^{-5}$  is added to the interaction parameters. In general, the algorithm is fairly robust to different initializations if the scale of the parameters is similar to the choices above. The learning rate  $\eta$  is set to 0.1.

Three strategies are used for estimation of  $\tau_{ij}$ :

1. Using the MLE  $\hat{\tau}_{ij} = t_{\ell_{ij}}$ ;
2. Setting  $\tau_{ij} = 0$ ;
3. Setting  $\tau_{ij} = 0$  if  $A_{ij} = 1$ , and  $\tau_{ij} = \infty$  if  $A_{ij} = 0$ .

The MLE approach 1 has a drawback: the  $p$ -values (16) for the first observation on each edge are *always* 1. This implies that the KS scores are bounded below by  $2720/30704 \approx 0.0885$  for the training set and  $287/3723 \approx 0.0770$  for the test set.

The KS scores obtained on the training and test sets after fitting different MEG models are reported in Table 1. The best performance (KS score 0.0152) is achieved when a Markov process is used for the interaction term, with  $d = 5$  or  $d = 10$ , combined with a Hawkes process for the main effects, setting  $\tau_{ij}$  using option 3. The same model achieves the best performance when alternative strategies for estimation of  $\tau_{ij}$  are used. If  $\tau_{ij}$  is set to its MLE 1, then the lower bound for the KS score on the training set is attained. In general, setting  $\tau_{ij}$  using option 3 seems to outperform competing strategies for estimation of  $\tau_{ij}$  in terms of KS scores. More importantly, overall the results demonstrate that the interaction term plays a key role in obtaining a good fit on the observed event times.

The results on the training set can also be compared to alternative node-based models from the literature. For example, Fox et al. (2016) propose the following node-specific intensity function for sending e-mails:

$$\lambda_i(t) = \alpha_i + \sum_{k=1}^{N'_i(t)} \mu_i \exp\{-(\mu_i + \phi_i)(t - t'_{ik})\}, \quad (17)$$

where the intensity jumps according to the event times of the *received* e-mails, see, (3) and (4). Despite the present article

using a slightly different number of e-mails, the Kolmogorov-Smirnov score obtained on the training data using (17) is 0.2806, which corresponds almost exactly to the result in Fox et al. (2016), demonstrating that the MEG appears to have superior performance for the Enron network. The parameters of (17) are estimated by direct optimization using the Nelder-Mead method on the negative log-likelihood function for each source node. Nearly identical results to Fox et al. (2016) are also obtained from fitting an independent Poisson processes  $\lambda_i(t) = \alpha_i$  on each source node, with KS score 0.4088. Finally, independent Hawkes process models of the form (1) are also fitted to each source node, obtaining a KS score of 0.2499 which is significantly outperformed by the best configuration of the MEG model. Since the MEG model KS score outperforms the value obtained using (17), it could be inferred that users tend to respond to multiple e-mails in sessions, and not necessarily immediately after an individual e-mail is received.

#### 5.4. Imperial College London NetFlow data

Many enterprises routinely collect network flow (NetFlow) data, representing summaries of connections between internet protocol (IP) addresses (see, e.g., Hofstede et al. 2014), which should be monitored for detecting unusual network activity, security breaches, and potential intrusions. Modeling arrival times in computer networks is complicated by several factors: events tend to appear in bursts, they might be recorded multiple times, and exhibit polling at regular intervals (Heard, Rubin-Delanchy, and Lawson 2014). In computer network security, it is particularly important to assess the significance of observing *new links*, corresponding to connections on previously unobserved edges (Metelli and Heard 2019). New links might be indicative of lateral movement, which is a common behavior of network attackers (Neil et al. 2013): intruders might move across the network with the purpose of escalating credentials, establishing connections which were previously unseen or unexpected. Therefore, correctly modeling new connections, and consequently providing reliable anomaly scores, is paramount for network security. The proposed MEG framework for modeling point processes on networks simultaneously addresses two fundamental tasks in network security: monitoring the normality of observed traffic, and anomaly detection for unusual new connections.

**Table 1.** Training and test Kolmogorov-Smirnov scores on the Enron e-mail network for different configurations of the MEG model.

KS scores (train and test)			Main effects $\alpha_i(\cdot)$ and $\beta_j(\cdot)$ ↓				
$\tau_{ij}$ ↓	Interactions $\gamma_{ij}(\cdot)$ ↓		Absent	Poisson ( $r = 0$ )	Markov ( $r = 1$ )	Hawkes ( $r = \infty$ )	
$\tau_{ij} = t_{e_{j1}}$ (MLE)	Absent		– –	0.4530 0.4133	0.3678 0.3484	0.4443 0.3586	
		Poisson ( $r = 0$ )	$d = 1$	0.4252 0.4221	0.3946 0.4179	0.3434 0.3574	0.4255 0.3560
			$d = 5$	0.3490 0.3851	0.3498 0.3953	0.3165 0.3677	0.3491 0.3613
	$d = 10$		0.3339 0.3763	0.3347 0.3688	0.3112 0.3470	0.3376 0.3575	
	Markov ( $r = 1$ )	$d = 1$	0.1662 0.2029	0.1491 0.1945	0.1305 0.1777	0.1702 0.1874	
		$d = 5$	0.0916 0.1875	0.0910 0.1684	<b>0.0885 0.1628</b>	0.0916 0.1746	
		$d = 10$	<b>0.0885</b> 0.1743	<b>0.0885</b> 0.1848	<b>0.0885</b> 0.1696	<b>0.0885</b> 0.1743	
	Hawkes ( $r = \infty$ )	$d = 1$	0.2640 0.2755	0.2825 0.2887	0.2538 0.2637	0.2599 0.2871	
		$d = 5$	0.2304 0.2904	0.2284 0.2760	0.2271 0.2774	0.2420 0.2981	
		$d = 10$	0.2461 0.2923	0.2521 0.2865	0.2413 0.3091	0.2498 0.3129	
	$\tau_{ij} = 0$	Absent		– –	0.7678 0.7983	0.7456 0.7360	0.7058 0.6046
			Poisson ( $r = 0$ )	$d = 1$	0.7039 0.7926	0.6627 0.7753	0.6543 0.7148
$d = 5$				0.5623 0.7059	0.5646 0.7206	0.5748 0.7008	0.7060 0.6053
$d = 10$		0.5354 0.6853		0.5332 0.6739	0.5725 0.6952	0.7060 0.6059	
Markov ( $r = 1$ )		$d = 1$	0.3135 0.3324	0.3004 0.3326	0.3262 0.3240	0.2027 0.1999	
		$d = 5$	0.0760 0.1664	0.0825 0.1584	0.0855 0.1782	0.0495 <b>0.0924</b>	
		$d = 10$	0.0775 0.1649	0.0793 0.1546	0.0816 0.1606	<b>0.0402</b> 0.0971	
Hawkes ( $r = \infty$ )		$d = 1$	0.2871 0.2486	0.2333 0.2449	0.2485 0.2379	0.1749 0.1991	
		$d = 5$	0.1939 0.2167	0.1885 0.2246	0.2010 0.2137	0.1467 0.1994	
		$d = 10$	0.2029 0.2395	0.2158 0.2470	0.2207 0.2339	0.1606 0.1943	
$\tau_{ij} = \begin{cases} 0, & A_{ij} = 1 \\ \infty, & A_{ij} = 0 \end{cases}$		Absent		– –	0.5590 0.5941	0.4112 0.3667	0.4593 0.2758
			Poisson ( $r = 0$ )	$d = 1$	0.5158 0.6038	0.4812 0.5864	0.3742 0.3602
	$d = 5$			0.4269 0.5516	0.4309 0.5641	0.3553 0.3598	0.3938 0.2803
	$d = 10$	0.4035 0.5413		0.4084 0.5565	0.3430 0.3537	0.3659 0.2810	
	Markov ( $r = 1$ )	$d = 1$	0.1950 0.2115	0.1600 0.2017	0.1504 0.1422	0.1309 0.1445	
		$d = 5$	0.0709 0.1222	0.0746 0.1008	0.0696 0.0917	<b>0.0152</b> 0.0848	
		$d = 10$	0.0619 0.1029	0.0627 0.1079	0.0634 0.0836	0.0213 <b>0.0800</b>	
	Hawkes ( $r = \infty$ )	$d = 1$	0.1870 0.2084	0.1816 0.2049	0.1783 0.1747	0.1719 0.1879	
		$d = 5$	0.1377 0.1805	0.1374 0.1840	0.1391 0.1642	0.1553 0.2154	
		$d = 10$	0.1556 0.2023	0.1588 0.2046	0.1546 0.1863	0.1640 0.2082	

The bold values correspond to the lowest KS scores for training and test set for each configuration of  $\tau_{ij}$ .

A bipartite dynamic network has been constructed from a subset of NetFlow data collected at Imperial College London (ICL). The network consists of 1,951,067 arrival times recorded to the nearest millisecond, observed between January 20, 2020, and 9th February 2020, recorded from  $n_1 = 173$  clients hosted within the Department of Mathematics at ICL, connecting to  $n_2 = 6083$  internet servers connecting on ports 80 and 443 (corresponding to unencrypted and encrypted web traffic), forming a total of 156,186 unique edges. The periodic and automated activity has been filtered by considering only edges such that the percentage of arrival times observed between 7 a.m. and 12 a.m. is larger than 99%, corresponding to the building opening hours. To learn connectivity patterns, the MEG model is trained on the first two weeks of data, corresponding to 1,299,372 events, and tested on 651,695 events observed in the final week. The number of unique edges observed in the training period is 115,600, and 70,408 in the test set; only 29,822 edges are observed in both time windows, which implies that 40,586 new edges are observed in the test set.

As discussed in Section 1, computer network data are observed in bursts and exhibit periodic behavior. Figure 6 gives an example of the connections from two of the clients to the ICL Virtual Learning Environment (VLE) server. Each session begins at an hour consistent with human behavior, while the frequency of subsequent connections within each session is likely to be due to automated activity and page refreshing.

The models have been initialized using a similar initialization scheme to Section 5.3, with learning rate  $\eta = 0.5$ . In particular, setting  $u_i = \frac{N_i(T)}{n_1 T}$  and  $u'_j = \frac{N'_j(T)}{n_2 T}$ , the chosen initial values are  $\alpha_i = \mu_i = u_i$ ,  $\phi_i = 3u_i$ ,  $\beta_j = \mu'_j = u'_j$ , and  $\phi'_j = 3u'_j$ ,  $\gamma_{i\ell} = (u_i)^{1/2}$ ,  $\gamma'_{j\ell} = (u'_j)^{1/2}$ ,  $v_{i\ell} = v'_{j\ell} = 10^{-4}$ , and  $\theta_{i\ell} = \theta'_{j\ell} = 5 \cdot 10^{-4}$ . As before, Gaussian noise is added to the interaction parameters if  $d > 1$ .

The likelihood for the Hawkes process is highly multimodal, and more sensitive to the initial values of the parameters than the Markov process with  $r = 1$ . Therefore, the parameters for the Hawkes process models are initialized with the optimal values obtained from the corresponding Markov process models, which seems to lead to fast convergence. The Kolmogorov-Smirnov scores calculated on the training and test set arrival times for different MEG models are reported in Table 2. The parameter  $\tau_{ij}$  is set according to option 3 from Section 5.3, which was observed to have the best performance on the Enron data.

The best performance (KS score 0.0728) is achieved by a Markov process with  $r = 1$  for both the main effects and interactions, and latent dimensionality  $d = 5$  for the parameters of the interaction term. Corresponding Q-Q plots for some of the models are plotted in Figure 7. Overall, the table and plots demonstrate that correctly modeling the arrival times requires inclusion within the model of an interaction term with a self-exciting component. Because of the extremely bursty

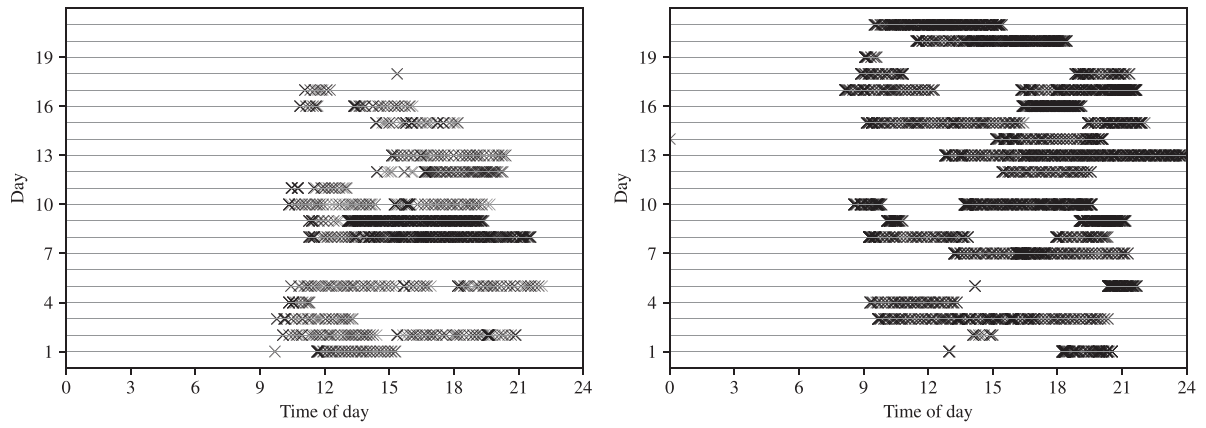


Figure 6. Connections to the ICL Virtual Learning Environment from two clients.

Table 2. KS scores on the ICL NetFlow data for different configurations of the MEG model.

KS scores (train and test)		Main effects $\alpha_i(\cdot)$ and $\beta_j(\cdot)$ ↓			
Interactions $\gamma_{ij}(\cdot)$ ↓		Absent	Poisson ( $r = 0$ )	Markov ( $r = 1$ )	Hawkes ( $r = \infty$ )
Absent		–			
Poisson ( $r = 0$ )	$d = 1$	0.7328 0.7157	0.7351 0.7148	0.6678 0.6489	0.7312 0.6950
	$d = 5$	0.7295 0.7167	0.7325 0.7150	0.6672 0.6480	0.7316 0.6960
	$d = 10$	0.7260 0.7174	0.7313 0.7123	0.6673 0.6487	0.7275 0.6967
Markov ( $r = 1$ )	$d = 1$	0.2194 0.1723	0.2242 0.1657	0.2038 0.1440	0.1645 0.1281
	$d = 5$	0.1024 0.1080	0.0896 0.0805	<b>0.0728 0.0738</b>	0.1041 0.0899
	$d = 10$	0.0843 0.0764	0.0871 0.0761	0.0850 0.0843	0.1100 0.0883
Hawkes ( $r = \infty$ )	$d = 1$	0.1080 0.0802	0.0747 0.1182	0.1082 0.0794	0.0884 0.1262
	$d = 5$	0.1576 0.1819	0.1532 0.2126	0.1677 0.2143	0.2307 0.2383
	$d = 10$	0.1584 0.1935	0.1546 0.2112	0.1619 0.2206	0.2388 0.2503

The bold values correspond to the lowest KS scores for training and test set.

(a) Training set

(b) Test set

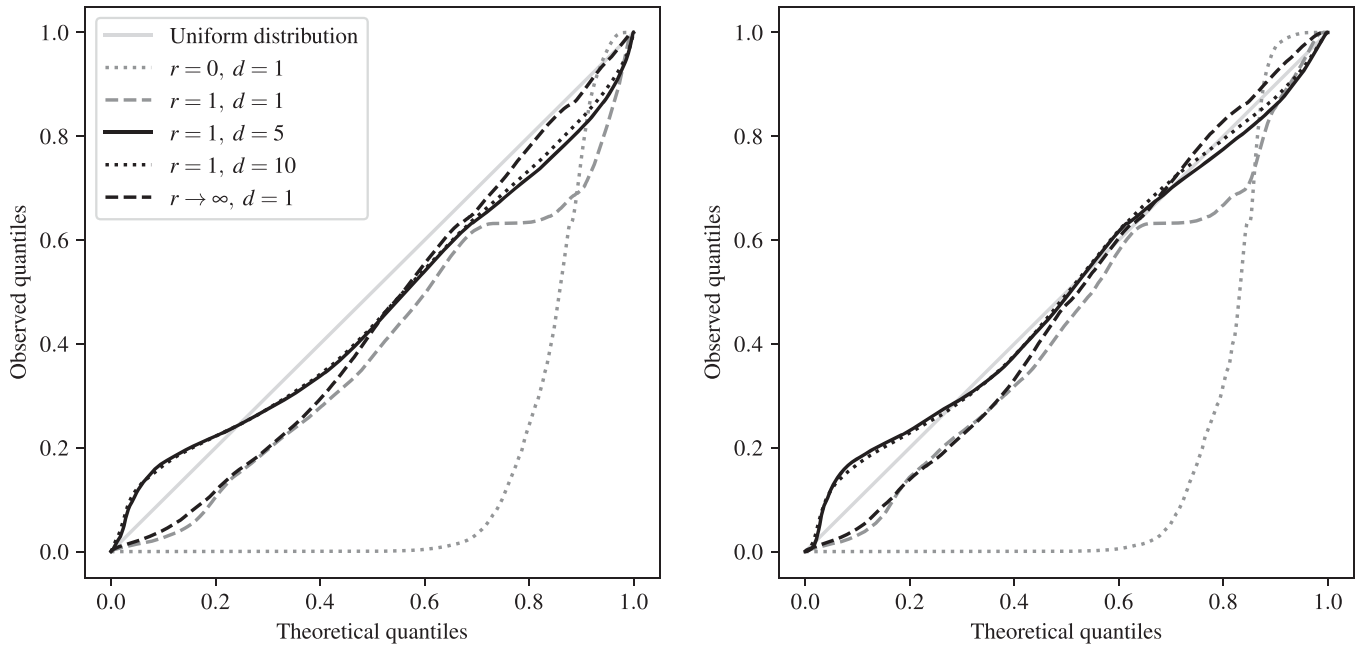
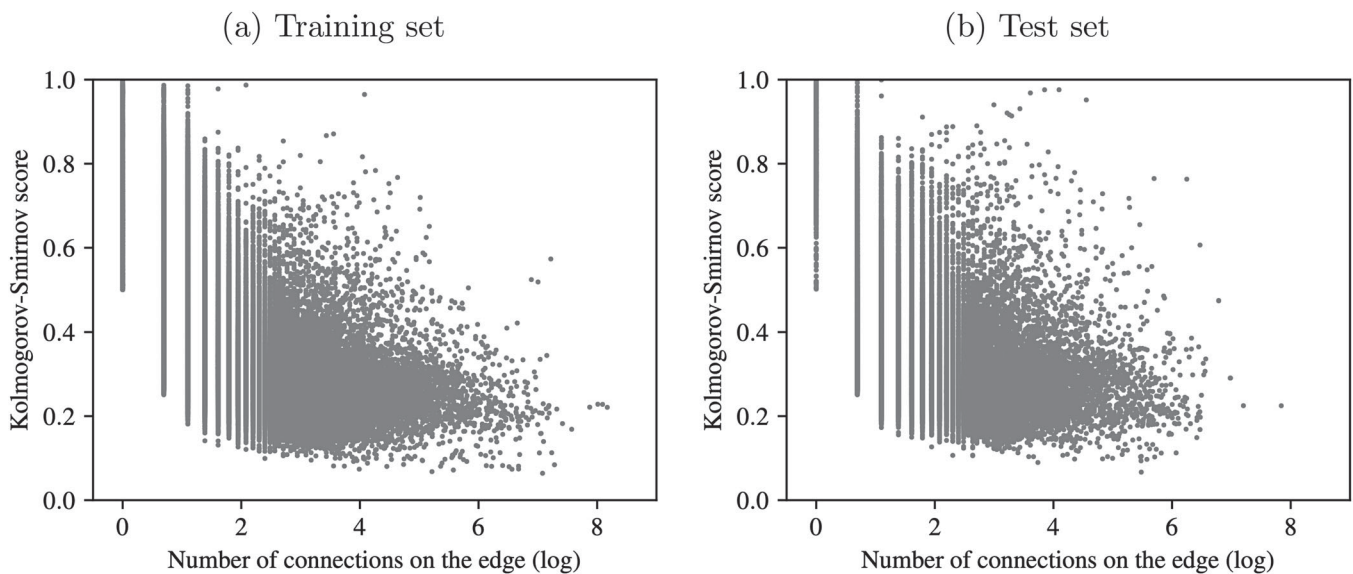


Figure 7. Q–Q plots for the training and test  $p$ -values obtained from different MEG models, with main effects  $\alpha_i(t)$  and  $\beta_j(t)$  with  $r = 1$ , and different parameters for the interaction term  $\gamma_{ij}(t)$ , specified in the legend.

behavior of NetFlow arrival times, the Markov process model for main effects and interactions intuitively appears to be a suitable choice.

Finally, for the best performing model the corresponding KS scores are calculated individually for each edge, and plotted in Figure 8 as a function of the number of connections on the edge.



**Figure 8.** Scatterplot of the KS scores, calculated for each edge, versus the logarithm of the total number of connections on the edge, for the best performing model in Table 2.

Clearly, the model has a better performance at scoring arrival times on more active edges.

## 6. Conclusion

The mutually exciting graph (MEG), a novel network-wide model for point processes with dyadic marks has been proposed. MEG uses mutually exciting point processes to model intensity functions, and borrows ideas from latent space models to infer relationships between the nodes. Edge-specific intensities are obtained only via node-specific parameters, which is useful for large and sparse graphs. Importantly, the proposed model is able to predict events observed on *new* edges. Inference is performed via maximum likelihood estimation, optimized using the EM algorithm, or numerically via modern gradient ascent methods. The model has been tested on simulated data and on two data sources related to computer networking: ICL NetFlow data and the Enron e-mail network. MEG appears to have excellent goodness-of-fit on training and testing data, resulting in low Kolmogorov-Smirnov scores even on very large and heterogeneous data like network flows. Furthermore, for the Enron e-mail network, MEG greatly outperforms results previously obtained in the literature on the same data. The model has been specifically motivated by cyber-security applications, where scoring observations on new links is particularly important for network security. Within this context, MEG might be used to complement existing techniques for modeling sequences of edges on dynamic networks (Sanna Passino and Heard 2019), providing a network-wide method for scoring arrival times.

The model could potentially be extended to admit an increasing number of nodes. Node-specific parameter values could be assigned after clustering similar nodes into groups, and allocating new nodes to one such community. For example, the initial parameter values for new nodes could correspond to the centroid of the corresponding community-specific parameter values. The initial cluster structure could be established from the connectivity patterns in the adjacency matrix (via spectral

clustering or modularity maximization), or from additional labels available for the nodes (for example, in cyber-security applications, subnets, geographical location, or machine type).

## Supplementary Materials

The Supplementary Material for this article contains details about the calculation of the log-likelihood and its gradient in MEG models. A *python* library to reproduce the results in this article, and a *bash* script to obtain the Enron e-mail network data, are available in the *GitHub* repository `fraspas/meg`.

## Acknowledgments

The authors thank Dr Melissa J. M. Turcotte for helpful discussions and comments.

## Funding

This work is funded by the Microsoft Security AI research grant “*Understanding the enterprise: Host-based event prediction for automatic defence in cyber-security.*”

## References

- Athreya, A., Fishkind, D. E., Tang, M., Priebe, C. E., Park, Y., Vogelstein, J. T., Levin, K., Lyzinski, V., Qin, Y., and Sussman, D. L. (2018), “Statistical Inference on Random Dot Product Graphs: A Survey,” *Journal of Machine Learning Research*, 18, 1–92. [3]
- Blundell, C., Beck, J., and Heller, K. (2012), “Modelling Reciprocating Relationships with Hawkes Processes,” in *Advances in Neural Information Processing Systems 25*, eds. F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Curran Associates. [2]
- Bowsher, C. G. (2007), “Modelling Security Market Events in Continuous Time: Intensity Based, Multivariate Point Process Models,” *Journal of Econometrics*, 141, 876–912. [1]
- Brown, E., Barbieri, R., Ventura, V., Kass, R., and Frank, L. (2002), “The Time-Rescaling Theorem and its Application to Neural Spike Train Data Analysis,” *Neural computation*, 14, 325–346. [7]

- Chen, F., and Tan, W. H. (2018), “Marked Self-exciting Point Process Modelling of Information Diffusion on Twitter,” *Annals of Applied Statistics*, 12, 2175–2196. [1]
- Chen, X., Liu, S., Sun, R., and Hong, M. (2019), “On the Convergence of A Class of Adam-Type Algorithms for Non-Convex Optimization,” in *International Conference on Learning Representations*. [7]
- Daley, D., and Vere-Jones, D. (2002), *An Introduction to the Theory of Point Processes – Volume I: Elementary Theory and Methods*, Probability and Its Applications, New York: Springer. [4]
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977), “Maximum Likelihood from Incomplete Data Via the EM Algorithm,” *Journal of the Royal Statistical Society, Series B*, 39, 1–22. [5]
- Durante, D., and Dunson, D. B. (2016), “Locally Adaptive Dynamic Networks,” *Annals of Applied Statistics*, 10, 2203–2232. [2]
- Eichler, M., Dahlhaus, R., and Dueck, J. (2017), “Graphical Modeling for Multivariate Hawkes Processes with Nonparametric Link Functions,” *Journal of Time Series Analysis*, 38, 225–242. [2]
- Etesami, J., Kiyavash, N., Zhang, K., and Singhal, K. (2016), “Learning Network of Multivariate Hawkes Processes: A Time Series Approach,” in *Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence*, UAI’16, AUAI Press. [2]
- Fox, E., Short, M., Schoenberg, F., Coronges, K., and Bertozzi, A. (2016), “Modeling E-mail Networks and Inferring Leadership Using Self-Exciting Point Processes,” *Journal of the American Statistical Association*, 111, 564–584. [2,5,10,11]
- Hawkes, A. (1971), “Spectra of some Self-exciting and Mutually Exciting Point Processes,” *Biometrika*, 58, 83–90. [1]
- Heard, N. A., Rubin-Delanchy, P. T. G., and Lawson, D. J. (2014), “Filtering Automated Polling Traffic in Computer Network Flow Data,” in *Proceedings – 2014 IEEE Joint Intelligence and Security Informatics Conference, JISIC 2014*, pp. 268–271. [11]
- Hoff, P. (2021), “Additive and Multiplicative Effects Network Models,” *Statistical Science*, 36, 34–50. [2]
- Hoff, P. D., Raftery, A. E., and Handcock, M. S. (2002), “Latent Space Approaches to Social Network Analysis,” *Journal of the American Statistical Association*, 97, 1090–1098. [2,3]
- Hofstede, R., Čeleda, P., Trammell, B., Drago, I., Sadre, R., Sperotto, A., and Pras, A. (2014), “Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX,” *IEEE Communications Surveys Tutorials*, 16, 2037–2064. [11]
- Kingma, D. P., and Ba, J. (2015), “Adam: A Method for Stochastic Optimization,” in *3rd International Conference on Learning Representations, ICLR*, eds. Y. Bengio and Y. LeCun, San Diego, CA, USA. [5,6]
- Krivitsky, P. N., and Handcock, M. S. (2014), “A Separable Model for Dynamic Networks,” *Journal of the Royal Statistical Society, Series B*, 76, 29–46. [2]
- Lee, W., McCormick, T. H., Neil, J., Sodja, C., and Cui, Y. (2021), “Anomaly Detection in Large Scale Networks with Latent Space Models,” *Technometrics*, 64, 241–252. [2]
- Linderman, S. W., and Adams, R. P. (2014), “Discovering Latent Network Structure in Point Process Data,” in *Proceedings of the 31st International Conference on Machine Learning - Volume 32, ICML’14*. [2]
- Metelli, S., and Heard, N. A. (2019), “On Bayesian New Edge Prediction and Anomaly Detection in Computer Networks,” *Annals of Applied Statistics*, 13, 2586–2610. [11]
- Miscouridou, X., Caron, F., and Teh, Y. W. (2018), “Modelling Sparsity, Heterogeneity, Reciprocity and Community Structure in Temporal Interaction Data,” in *Proceedings of the 32nd International Conference on Neural Information Processing Systems, NIPS’18*. [2]
- Mohler, G. O., Short, M. B., Brantingham, P. J., Schoenberg, F. P., and Tita, G. E. (2011), “Self-Exciting Point Process Modeling of Crime,” *Journal of the American Statistical Association*, 106, 100–108. [1]
- Neil, J., Hash, C., Brugh, A., Fisk, M., and Storlie, C. B. (2013), “Scan Statistics for the Online Detection of Locally Anomalous Subgraphs,” *Technometrics*, 55, 403–414. [11]
- Ogata, Y. (1978), “The Asymptotic Behaviour of Maximum Likelihood Estimators for Stationary Point Processes,” *Annals of the Institute of Statistical Mathematics*, 30, 243–261. [4]
- (1981), “On Lewis’ Simulation Method for Point Processes,” *IEEE Transactions on Information Theory*, 27, 23–31. [7]
- (1988), “Statistical Models for Earthquake Occurrences and Residual Analysis for Point Processes,” *Journal of the American Statistical Association*, 83, 9–27. [1]
- Ozaki, T. (1979), “Maximum Likelihood Estimation of Hawkes’ Self-exciting Point Processes,” *Annals of the Institute of Statistical Mathematics*, 31, 145–155. [1]
- Perry, P., and Wolfe, P. (2013), “Point Process Modelling for Directed Interaction Networks,” *Journal of the Royal Statistical Society, Series B*, 75, 821–849. [2]
- Price-Williams, M., and Heard, N. A. (2020), “Nonparametric Self-exciting Models for Computer Network Traffic,” *Statistics and Computing*, 30, 209–220. [1,10]
- Priebe, C. E., Conroy, J. M., Marchette, D. J., and Park, Y. (2005), “Scan Statistics on Enron Graphs,” *Computational & Mathematical Organization Theory*, 11, 229–247. [10]
- Reddi, S. J., Kale, S., and Kumar, S. (2018), “On the Convergence of Adam and Beyond,” in *International Conference on Learning Representations*. [7]
- Ruder, S. (2016), “An Overview of Gradient Descent Optimization Algorithms,” arXiv. [7]
- Sanna Passino, F., and Heard, N. A. (2019), “Modelling Dynamic Network Evolution as a Pitman-Yor Process,” *Foundations of Data Science*, 1, 293–306. [14]
- Sarkar, P., and Moore, A. W. (2006), “Dynamic Social Network Analysis using Latent Space Models,” in *Advances in Neural Information Processing Systems (Vol. 18)*, 1145–1152. [2]
- Sewell, D. K., and Chen, Y. (2015), “Latent Space Models for Dynamic Networks,” *Journal of the American Statistical Association*, 110, 1646–1657. [2]
- Stomakhin, A., Short, M. B., and Bertozzi, A. L. (2011), “Reconstruction of Missing Data in Social Networks Based on Temporal Patterns of Interactions,” *Inverse Problems*, 27, 115013, DOI: 10.1088/0266-5611/27/11/115013/meta. [1]
- Zhao, Q., Erdogdu, M. A., He, H. Y., Rajaraman, A., and Leskovec, J. (2015), “SEISMIC: A Self-Exciting Point Process Model for Predicting Tweet Popularity,” in *Proceedings of the 21th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. [1]
- Zou, F., Shen, L., Jie, Z., Zhang, W., and Liu, W. (2019), “A Sufficient Condition for Convergences of Adam and RMSProp,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE Computer Society. [7]