The Multivariate Community Hawkes Model for Dependent Relational Events in Continuous-time Networks

Hadeel Soliman [1] **Lingfei Zhao** [2] **Zhipeng Huang** [1] **Subhadeep Paul** [2] **Kevin S. Xu** [1]

Abstract

The stochastic block model (SBM) is one of the most widely used generative models for network data. Many continuous-time dynamic network models are built upon the same assumption as the SBM: edges or events between all pairs of nodes are conditionally independent given the block or community memberships, which prevents them from reproducing higher-order motifs such as triangles that are commonly observed in real networks. We propose the *multivariate community Hawkes (MULCH) model*, an extremely flexible community-based model for continuous-time networks that introduces *dependence between node pairs* using structured multivariate Hawkes processes. We fit the model using a spectral clustering and likelihood-based local refinement procedure. We find that our proposed MULCH model is far more accurate than existing models both for predictive and generative tasks.

1. Introduction

Networks are often used to represent data in the form of relations (edges) between a set of entities (nodes). In many settings, the nodes and edges change over time, resulting in dynamic or temporal networks. We consider networks observed through *timestamped relational events*, where each event is a triplet (i, j, t) denoting events from node i (sender) to node j (receiver) at timestamp t. Application settings involving timestamped relational events include interactions (messages, likes, shares, etc.) between users on social media, financial transactions between buyers and sellers, and military actions between countries in diplomatic conflicts.

[1]Department of Electrical Engineering and Computer Science, University of Toledo, Toledo, OH, USA [2]Department of Statistics, The Ohio State University, Columbus, OH, USA. Correspondence to: Subhadeep Paul <paul.963@osu.edu>, Kevin S. Xu <kevin.xu@utoledo.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

There has been significant recent interest in generative models for timestamped relational event data. Such models typically combine a Temporal Point Process (TPP) model such as a Hawkes process (Laub et al., 2021) for event times with a latent variable network model such as a Stochastic Block Model (SBM) (Nowicki & Snijders, 2001) for the sender and receiver of the event (Blundell et al., 2012; DuBois et al., 2013; Yang et al., 2017; Miscouridou et al., 2018; Matias et al., 2018; Junuthula et al., 2019; Arastuie et al., 2020; Huang et al., 2022). We call such models *continuous-time network models* because they provide probabilities of observing events between nodes at arbitrary times.

Continuous-time network models typically assume that the probability of an event occurring between a pair of nodes (i, j) at some time t is conditionally independent of all other node pairs given the latent variable representation of the network. The conditional independence between node pairs allows them to be modeled separately using univariate or bivariate TPPs. Such an approach makes the model more tractable but prevents it from replicating higher-order structures including triangles and other network motifs (Benson et al., 2016). Of particular interest in the dynamic network setting are *temporal motifs*, which require multiple events to be formed between different nodes within a certain time window (Paranjape et al., 2017).

We propose the *multivariate community Hawkes (MULCH) model*, a highly flexible continuous-time network model that introduces dependence between node pairs in a controlled manner. We jointly model all node pairs using a multivariate Hawkes process where an event between a node pair (x, y) can increase the probability of an event between a different node pair (i, j). To keep the model tractable, we impose an SBM-inspired structure on the excitation matrix $\boldsymbol{\alpha}$ of the multivariate Hawkes process. We consider several different types of excitation to encourage formation of temporal motifs, inspired by the notion of participation shifts (Gibson, 2003; 2005) from sociology.

Our main contributions are as follows: (1) We propose the highly flexible MULCH model for continuous-time networks using multivariate Hawkes processes that incorporate many types of dependence between node pairs including reciprocity and participation shifts. (2) We develop an

efficient estimation approach for MULCH using spectral clustering and likelihood-based local refinement. (3) We demonstrate the MULCH fits to several real data sets are superior to existing continuous-time network models both at predicting future events and at replicating temporal motifs. (4) We present a case study using MULCH to analyze military dispute data, revealing how groups of countries act and respond to other actions in such disputes.

2. Background

2.1. Multivariate Hawkes Process

Temporal point processes (TPPs) are used to model events that occur randomly (i.e. not at regularly spaced intervals) over time. A univariate TPP is characterized by a conditional intensity function $\lambda(t|\mathcal{H}_t)$ such that the expected number of events in an infinitesimal interval dt around t is given by $\lambda(t|\mathcal{H}_t)\,dt$ (Rasmussen, 2018). $\mathcal{H}_t$ denotes the history of the TPP (all event times up to time t); we drop it for ease of notation and simply use $\lambda(t)$. A univariate Hawkes process is a self-exciting TPP where the occurrence of an event increases the probability of another event occurring shortly afterwards (Laub et al., 2021). Given a sequence of timestamps $\{t_1, t_2, \ldots, t_l\}$ for l events, the conditional intensity function takes the form $\lambda(t) = \mu + \alpha \sum_{s:t_s<t} \gamma(t - t_s)$, where μ is the background or base intensity, α is the jump size or excitation, and $\gamma(\cdot)$ is the kernel function.

A multivariate TPP is characterized by a set of conditional intensity functions $\{\lambda_j(t)\}_{j=1}^d$ for the d different variables. A multivariate Hawkes process is both self and mutually exciting, so that an event in one variable can also increase the probability of an event in another variable (Zhou et al., 2013b; Hawkes, 2018). The conditional intensity function of the jth dimension is given by $\lambda_j(t) = \mu_j + \sum_{s:t_s<t} \alpha_{ij}\gamma_{ij}(t - t_s)$, where the background intensity μ_j can differ for each dimension, α_{ij} denotes the jump size that an event in dimension i causes to dimension j, and $\gamma_{ij}(\cdot)$ denotes the kernel, which may differ across dimensions. Notice that the background intensities are now characterized by a d-dimensional vector $\boldsymbol{\mu}$, while the jump sizes and kernels are characterized by $d \times d$ matrices $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$, respectively.

2.2. Stochastic Block Model

The *stochastic block model* (SBM), first formalized by Holland et al. (1983), is one of the most widely used generative models for network data. The SBM was designed for a static network, but many dynamic extensions have since been proposed, which we discuss in the related work section. In a (static) SBM with n nodes, every node i is assigned to one and only *block* $Z_i \in \{1, \ldots, K\}$, where K denotes the number of blocks. For a directed SBM, given the node membership vector $\boldsymbol{Z} = [Z_i]_{i=1}^n$, all off-diagonal entries of the adjacency matrix A_{ij} are independent Bernoulli random variables with parameter p_{Z_i,Z_j}, where p is a $K \times K$ matrix of probabilities which is not symmetric. Thus, the probability of forming an edge between nodes i and j depends only on the node memberships Z_i and Z_j.

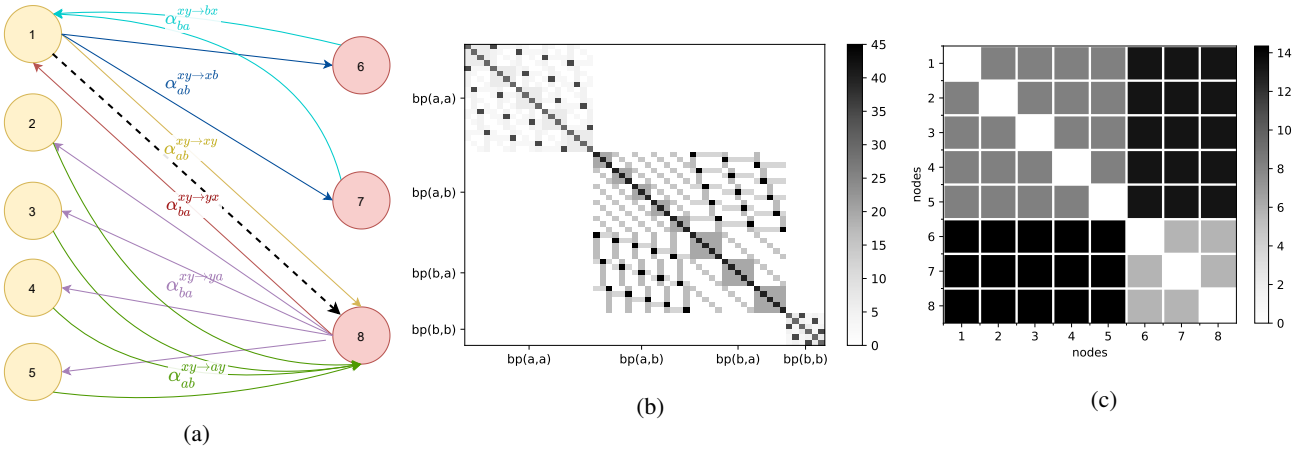
Real networks often have many reciprocated edges ($A_{ij} = 1 \Rightarrow A_{ji} = 1$) and triangles ($A_{ij} = 1, A_{jm} = 1 \Rightarrow A_{im} = 1$). These are not replicated by the SBM due to the independence of the adjacency matrix entries. Replicating triangles usually requires an additional generative process on top of the SBM (Peixoto, 2022) or generative models which induce dependence (Paul et al., 2018; Bollobás et al., 2011).

Fitting an SBM to data involves estimating both the node memberships $\boldsymbol{Z}$ and the edge probabilities p_{Z_i,Z_j} between all pairs of blocks (Nowicki & Snijders, 2001). Several variants of spectral clustering, including regularized versions (Chaudhuri et al., 2012; Amini et al., 2013), have been shown to be consistent estimators of the node memberships in the SBM and various extensions in several asymptotic settings (Rohe et al., 2011; Sussman et al., 2012; Qin & Rohe, 2013; Lei & Rinaldo, 2015; Chin et al., 2015; Han et al., 2015; Gao et al., 2017). Spectral clustering scales to large networks with hundreds of thousands of nodes and is generally not sensitive to initialization, so it is also a practically useful estimator.

2.3. Related Work

Most prior models for continuous-time networks utilize low-dimensional latent variable representations of the networks to parameterize univariate or bivariate TPPs, typically Hawkes processes, for the node pairs. The self excitation in Hawkes processes has been found to be a good model for conversation event sequences (Masuda et al., 2013) among other temporal relational event data. The latent variable representations are often inspired by generative models for static networks such as latent space models (Hoff et al., 2002) and stochastic block models. Continuous-time network models have been built with continuous latent space representations (Yang et al., 2017; Huang et al., 2022) and latent block or community representations (Blundell et al., 2012; DuBois et al., 2013; Xin et al., 2017; Matias et al., 2018; Miscouridou et al., 2018; Corneli et al., 2018; Junuthula et al., 2019; Arastuie et al., 2020).

The CHIP (Arastuie et al., 2020) model uses a univariate Hawkes process to model self excitation for each node pair, with node pairs in the same community pair sharing parameters. Bivariate Hawkes process models (Blundell et al., 2012; Yang et al., 2017; Miscouridou et al., 2018; Huang et al., 2022) allow events $i \to j$, which we denote by the directed pair (i, j), to influence the probability of events (j, i). This encourages reciprocal events, which are commonly



(a) (b) (c)

Figure 1. Toy example illustrating the different excitations in the proposed multivariate block Hawkes model. Nodes 1-5 are in block a, while nodes 6-8 are in block b. (a) The event $(1, 8)$ (dashed line) excites the processes for the other node pairs shown in solid lines by the specified jump size α_{ab} or α_{ba}. (b) Block diagonal structure of the excitation matrix $\boldsymbol{\alpha}$ for this toy example. (c) Block structure of the expected count matrix (expected number of events between each node pair) for this toy example.

seen in email and messaging networks, where a reciprocal event typically denotes a user replying to a message. A weakness of such models is that they still have no mechanism to encourage the formation of higher-order motifs such as triangles given their bivariate nature. One way to accomplish this would be to move to higher-dimensional Hawkes processes, as we propose in this paper.

High-dimensional multivariate Hawkes processes are frequently used for estimating the structure of a latent or unobserved network from observed events at the nodes (Zhou et al., 2013b; Linderman & Adams, 2014; Farajtabar et al., 2015; Tran et al., 2015). These models are often used to estimate *static* networks of diffusion from information cascades. The information cascade is modeled using an n-dimensional Hawkes process with each dimension corresponding to a node, and the objective is to estimate the $n \times n$ excitation matrix $\boldsymbol{\alpha}$ of influence strengths between nodes.

3. Multivariate Community Hawkes Model

In order to replicate higher-order motifs such as triangles, we must move beyond univariate and bivariate Hawkes process-based models. One approach would be to model all (ordered) node pairs (i, j) using an $n(n-1)$-dimensional Hawkes process where an event for any node pair can excite any other node pair, similar to the models used for information cascades (Zhou et al., 2013b; Tran et al., 2015). Such a model would have an $n(n-1) \times n(n-1)$ excitation matrix $\boldsymbol{\alpha}$, which would be computationally intractable even for relatively small networks with a few hundred nodes.

We propose the *multivariate community Hawkes (MULCH)* model, an extremely flexible continuous-time network model based on the SBM where each node i belongs to a block $a \in \{1, \dots, K\}$, which we denote by $i \in a$ or $Z_i = a$. Each node pair (i, j) then belongs to a block pair (a, b), which we denote by $(i, j) \in \text{bp}(a, b)$. We assume a block diagonal structure on $\boldsymbol{\alpha}$ so that events for node pair $(x, y) \in \text{bp}(a, b)$ can only excite node pairs $(i, j) \in \{\text{bp}(a, b), \text{bp}(b, a)\}$. This block diagonal structure is shown in Figure 1b. This structure allows for higher-order motifs to form in the same (a, b) and reciprocal (b, a) block pairs of the initial event for node pair (x, y).

Within the non-zero diagonal blocks of $\boldsymbol{\alpha}$, we introduce dependence between node pairs in a controlled manner using different types of excitations. Consider a node pair $(i, j) \in \text{bp}(a, b)$, which has conditional intensity function

$$\lambda_{ij}(t) = \mu_{ab} + \sum_{\substack{(x,y)\in\text{bp}(a,b)\\(x,y)\in\text{bp}(b,a)}} \alpha_{ab}^{xy\to ij} \sum_{t_s\in T_{xy}} \gamma_{xy\to ij}(t - t_s),$$

where T_{xy} denotes the set of times that event (x, y) happens. The parameters $\alpha_{ab}^{xy\to ij}$ control the types of excitations in the model by denoting which node pairs (x, y) can increase the probability of an event for node pair (i, j) occurring shortly after time t.

3.1. Excitation Parameters

Many variants of our proposed multivariate block Hawkes model are possible depending on the structure of the parameters $\alpha_{ab}^{xy\to ij}$ corresponding to different types of excitations. We consider 6 types of excitations, listed in Table 1 and illustrated for a sample network in Figure 1a. If we had only the excitation parameter $\alpha_{ab}^{xy\to xy}$, then our model would only incorporate self excitation and reduces to the CHIP model

Table 1. Descriptions of the 6 types of excitations (illustrated in Figure 1a) we consider following an event from node x in block a to node y in block b. Excitations that trigger events from block b back to block a are modeled by α_{ba}, not α_{ab}.

Parameter	Excitation Type
$\alpha_{ab}^{xy\to xy}$	*Self excitation*: continuation of event (x, y)
$\alpha_{ba}^{xy\to yx}$	*Reciprocal excitation*: event (y, x) taken in response to event (x, y)
$\alpha_{ab}^{xy\to xb}$	*Turn continuation*: (x, b) following (x, y) to other nodes except for y in the same block b
$\alpha_{ba}^{xy\to ya}$	*Generalized reciprocity*: (y, a) following (x, y) to other nodes except x in block a
$\alpha_{ab}^{xy\to ay}$	*Allied continuation*: event (a, y) following (x, y) from other nodes except x in block a
$\alpha_{ba}^{xy\to bx}$	*Allied reciprocity*: event (b, x) following (x, y) from other nodes except y in block b

Algorithm 1 MULCH Model Generative Process

Input: Block probabilities $\boldsymbol{\pi}$, Hawkes process parameters $\boldsymbol{\Theta}$, Duration T
Output: Node memberships $\boldsymbol{Z}$, Array of event triplets E

for $i = 1$ to n **do**
  $Z_i \sim$ Categorical($\boldsymbol{\pi}$)
end for
Split $n \times (n-1)$ node pairs into $K \times K$ block pairs
for each diagonal block pair (a, a) **do**
  $E_{aa} \leftarrow$ MultivariateHawkesProcess($\boldsymbol{\theta}_{aa}, T$)
end for
for each off-diagonal block pairs (a, b) and (b, a) **do**
  $E_{ab}, E_{ba} \leftarrow$ MultivariateHawkesProcess($\boldsymbol{\theta}_{ab}, \boldsymbol{\theta}_{ba}, T$)
end for
Concatenate all E_{aa} and E_{ab} into E
return $\boldsymbol{Z}, E$

(Arastuie et al., 2020). The remaining 5 parameters denote mutual excitation. These different types of mutual excitation correspond to the notion of participation shifts (Gibson, 2003; 2005) from sociology. The terms "turn continuation" (Gibson, 2003) and "generalized reciprocity" (Yamagishi & Kiyonari, 2000) also have origins from sociology.

Each excitation is associated with its own jump size parameter that controls probability of such an event following the initial event (x, y). The structure of a non-zero diagonal block of $\boldsymbol{\alpha}$ under these 6 excitations is shown in Figure 1b. By using specific types of excitations in a multivariate Hawkes process model, we not only reduce the number of parameters, but also improve interpretability through the set of excitation parameters for each block pair.

While other models have incorporated self excitation (Arastuie et al., 2020; Junuthula et al., 2019) and reciprocal excitation (Blundell et al., 2012; Yang et al., 2017; Miscouridou et al., 2018), our model is the first to incorporate the 4 other types of excitations. These newly-introduced excitations incorporate higher-order dependencies beyond a node pair to encourage the formation of motifs such as triangles.

3.2. Generative Process

To generate events between node pairs in the network, we first sample the membership of each node Z_i from a categorical distribution with block probability $\boldsymbol{\pi} = [\pi_1, \cdots, \pi_K]$. There is a total of $n(n-1)$ node pairs, which are split into $K \times K$ block pairs. All node pairs within one block pair (a, b) share the same base intensity μ_{ab} and same set of 6 excitation parameters from Table 1. We denote these 7 parameters for a block pair by $\boldsymbol{\theta}_{ab}$.

By ordering node pairs by block pairs, we can form the excitation matrix $\boldsymbol{\alpha}$ for a high-dimensional multivariate Hawkes process as shown in Figure 1b. As a result of MULCH's block diagonal structure on $\boldsymbol{\alpha}$, we can simulate events between $(i, j) \in \{\text{bp}(a, b), \text{bp}(b, a)\}$ as separate multivariate Hawkes processes. We use a variant of Ogata's thinning algorithm proposed by Xu et al. (2020) to simulate the multivariate Hawkes processes. Given the block probabilities $\boldsymbol{\pi}$, the Hawkes process parameters for all block pairs $\boldsymbol{\Theta} = [\boldsymbol{\theta}_{ab}]$, and simulation duration T, the generative process is summarized in Algorithm 1. The generated events are sets of triplets (i, j, t) denoting nodes and timestamps concatenated into a single array E.

3.3. Hawkes Process Kernel Selection

A frequent choice of kernel for univariate Hawkes processes is an exponential kernel $\gamma(t) = e^{-\beta(t)}$. Estimation of β is difficult, and estimators are typically not well-behaved (Santos et al., 2021). In many applications involving multivariate Hawkes processes, the exponential kernel is normalized so that $\gamma_{xy\to ij}(t) = \beta e^{-\beta(t)}$, and β is typically assumed to be a fixed rather than estimated parameter (Zhou et al., 2013b; Bacry et al., 2015; 2017). While assuming β to be fixed greatly simplifies estimation, a poor choice of β may result in a much worse fit compared to estimating β.

One way to mitigate this possibility is to use a weighted sum of different kernels (Zhou et al., 2013a). Inspired by Yang et al. (2017), we use a sum of multiple exponential kernels with different decay rates. Let $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_Q)$ denote a set of Q fixed decays shared among all block pairs. For each $\text{bp}(a, b)$, we introduce a block pair-specific kernel scaling parameter $\boldsymbol{C}_{ab} = \left(C_{ab}^1, \cdots, C_{ab}^Q\right)$ that is estimated simultaneously with $\boldsymbol{\Theta}$. For identifiability, we assume $\sum_{q=1}^{Q} C_{ab}^q = 1$, and $C_{ab}^q \in [0, 1]$. The full expression for the MULCH conditional intensity function with the sum of exponential kernels is provided in Appendix A.1.

4. Estimation Procedure

Fitting the MULCH model involves both estimation of node membership vector $\boldsymbol{Z}$ and Hawkes process parameters $\boldsymbol{\theta}_{ab}$ for all block pairs (a, b). We first aggregate the number of events between each node pair (i, j) to form entry $N_{ij}(T)$ of the count adjacency matrix $\boldsymbol{N}(T)$. We then apply spectral clustering to the count matrix $\boldsymbol{N}(T)$ to obtain an initial estimate of the node memberships $\boldsymbol{Z}$. Since the network is directed, we use singular vectors for spectral clustering according to the algorithm of Sussman et al. (2012) with the added step of row normalization of the singular vectors (Rohe et al., 2016), which is beneficial in the presence of degree heterogeneity.

The expectation of the count matrix has a block structure induced by MULCH, as shown in Figure 1c. The expected count matrix is simply $E[\boldsymbol{N}(T)] = \boldsymbol{\lambda}T$, where $\boldsymbol{\lambda}$ denotes the expected intensity function. When the process is stationary, we have the following theorem to ensure that the expected count matrix has the block structure.

Theorem 4.1. *For any $(i, j) \in (a, b)$ and $(i', j') \in (a, b)$, $\boldsymbol{\lambda}_{(i,j)} = \boldsymbol{\lambda}_{(i',j')} = g_{ab}\mu_{ab} + g_{ba}\mu_{ba}$, where g_{ab} and g_{ba} are real valued functions which depend on model parameters.*

Proof of this theorem can be found in Appendix A.3. As the duration $T \to \infty$, the count matrix $\boldsymbol{N}(T)$ should approach the expected count matrix, and spectral clustering should succeed in recovering the correct node memberships. For finite T, the estimated node memberships for spectral clustering may not be optimal, so we run an iterative refinement procedure using the likelihood to improve the estimated node memberships, which we describe in Section 4.1.

We do not have a theoretical guarantee for spectral clustering estimation accuracy due to the *dependent* adjacency matrix entries in MULCH resulting from the different excitations in Table 1. This is unlike CHIP (Arastuie et al., 2020), which had independent adjacency matrix entries and used proof techniques that assumed independence.

After we get the node membership $\boldsymbol{Z}$, we then estimate $\boldsymbol{\Theta}$, using a maximum likelihood estimation (MLE) approach. Under the MULCH assumptions, the model's log-likelihood can be expressed as a sum over block pair log-likelihoods

$$\ell(\boldsymbol{\Theta}|\boldsymbol{Z}, \mathcal{H}_t) = \sum_{a=1}^{K}\sum_{b=1}^{K} \ell_{ab}(\boldsymbol{\theta}_{ab}|\boldsymbol{Z}, \mathcal{H}_t) \tag{1}$$

The detailed form for $\ell_{ab}(\boldsymbol{\theta}_{ab}|\boldsymbol{Z}, \mathcal{H}_t)$ is provided in Appendix A.2. Each block pair log-likelihood function $\ell_{ab}(\boldsymbol{\theta}_{ab}|\boldsymbol{Z}, \mathcal{H}_t)$ can be maximized by standard non-linear optimization methods. We use the L-BFGS-B (Byrd et al., 1995) optimizer implemented in SciPy. Parameters are initialized to small random numbers, and we set bounds to be between $(10^{-7}, +\infty)$ to ensure that they are positive.

4.1. Likelihood Refinement Procedure

Next, we propose a likelihood refinement procedure using the MULCH model likelihood to improve the estimation of the node memberships starting from a spectral clustering initial solution. The algorithm is motivated by leave-one-out likelihood maximization procedures that are commonly employed in conjunction with spectral clustering to achieve minimax optimal error rates of community detection in the context of stochastic block models and extensions (Gao et al., 2017; 2018; Chen et al., 2020). However, in contrast to the previous literature, we do not use the likelihood of the count matrix elements, but instead use the likelihood of the event times to obtain the refined node memberships.

Our refinement procedure has 3 steps. First, we perform spectral clustering on the count matrix to get the initial node memberships $\boldsymbol{Z}^0$, and then estimate the parameters through maximum likelihood estimation as described in the previous section. Then for each node i, we assign it to the block which can maximize the log-likelihood (1) given the estimated node memberships of all other nodes and the estimated parameters. Finally, we re-estimate the model parameters using the new block assignment.

The refinement step involves computing a likelihood over significantly less number of events and is thus computationally efficient. To get better results, we run the refinement multiple times, each time using the latest refined estimates as the initial values. We end the refinement when no node memberships change or at a maximum of 15 iterations.

4.2. Model Selection

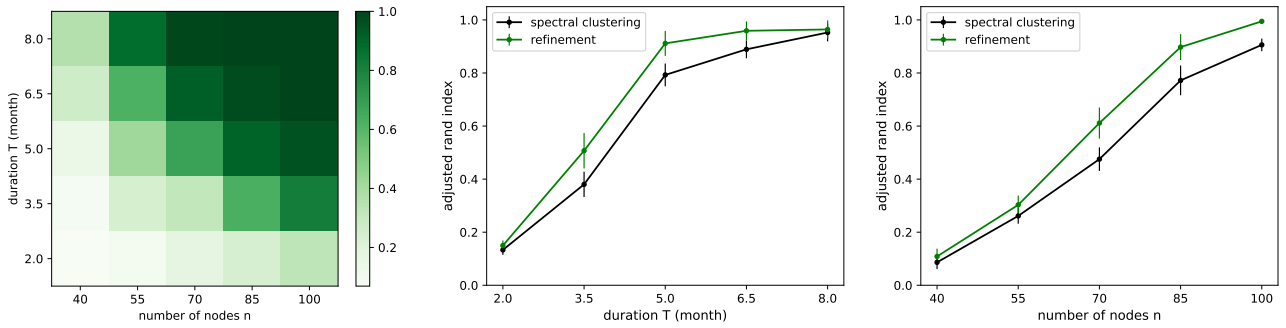
Up to this point, we have assumed that both the decay values β_q used in the sum of exponential kernels and the number of blocks K are fixed and known. The β_q values should be chosen appropriately for the type of network. For example, for networks with rapid dynamics such as instant messaging networks, suitable β_q values will likely be on the order of minutes, hours, or days; whereas, for networks with slower dynamics, suitable β_q values may be in the order of weeks or months. We choose three β_q values in our experiments.

In most application settings, the number of blocks K is usually also unknown. We use task-based model selection in our experiments; that is, we choose the value of K that maximizes the evaluation metric we use for predictive accuracy (Section 5.2.1) or generative accuracy (Section 5.2.2)

5. Experiments

5.1. Simulated Networks

We first test the ability of both spectral clustering and our likelihood refinement procedure to recover true node memberships on networks simulated from MULCH. In Appendix



(a) (b) (c)

Figure 2. Block estimation accuracy averaged over 10 simulated networks. (a) Heat map of adjusted Rand index for spectral clustering at $K = 4$ while varying T, n. Comparison between adjusted Rand index achieved by spectral clustering and refinement algorithm for (b) fixed $n = 70$, varying T and (c) fixed $T = 3.5$ months, varying n ($\pm$ standard error over 10 runs).

B.2, we present an additional experiment evaluating parameter estimation accuracy.

Spectral Clustering Accuracy We simulate networks at $K = 4$ while varying both n and T. For each (n, T) value, we simulate a network from the MULCH model, run spectral clustering on the count matrix, and calculate the adjusted Rand index (Hubert & Arabie, 1985) between the true and estimated node memberships, where a score of 1 indicates perfect clustering and 0 is the expected score for a random estimate. As shown in Figure 2a, the accuracy of estimated node memberships improves as both n, T increase. The average score over 10 simulations is shown and indicates that spectral clustering can recover true node memberships for large n and T. Parameter values used to simulate the networks are listed in Appendix B.1.

Likelihood Refinement Accuracy Figure 2b shows the adjusted Rand index from spectral clustering and from applying our refinement algorithm to networks simulated at $n = 70$ and varying T. Similarly, Figure 2c shows the adjusted Rand index at $T = 3.5$ months and varying n. Each point is averaged over 10 simulations. Notice that the adjusted Rand index always improves after applying our refinement algorithm, and in some cases, allows for perfect estimation of node memberships for cases where spectral clustering still makes errors.

5.2. Real Networks

We perform benchmark experiments on 4 real network datasets to evaluate the predictive and generative accuracy of our proposed MULCH model against several other models[1]. Summary statistics for the datasets are shown in Table 2, with additional details in Appendix B.3. Each dataset consists of a set of events where each event is denoted by a sender, a receiver, and a timestamp.

Table 2. Summary statistics of real network datasets

Dataset	Nodes	Total Events	Test Events
Reality	70	2,161	661
Enron	142	4,000	1,000
MID	147	5,117	1,078
Facebook	43,953	852,833	170,567

Models for Comparison We compare against several other TPP models for continuous-time networks: REM (DuBois et al., 2013), BHM (Junuthula et al., 2019), CHIP (Arastuie et al., 2020), DLS (Yang et al., 2017), and ADM4 (Zhou et al., 2013b). Each of these models can be fit to a network and used to evaluate test log-likelihood on future events and to simulate networks from the fit. REM, BHM, CHIP, and DLS are continuous-time network models, while ADM4 is a sparse and low-rank regularized model for general multivariate Hawkes processes. Additional details on these models is provided in Appendix B.4. We present also experiments on scalability and on other parameterizations for MULCH in Appendices B.5 and B.6.

5.2.1. Predictive Accuracy

Experiment Set-up We first evaluate the ability of our proposed MULCH model to predict future events between nodes. To do this, we split the data into training and test sets, with the first l_{train} events being used to fit the model and the remaining l_{test} events (shown in Table 2) being used to evaluate the model's predictive ability. We assign all new nodes present in the test set but not the training set to the largest block in the training set, consistent with Arastuie

[1]Python code is available at https://github.com/IdeasLabUT/Multivariate-Community-Hawkes

Table 3. Mean test log-likelihood per event for each real network dataset across all models. Larger (less negative) values indicate better predictive ability. Bold entry denotes highest accuracy for a dataset. Results for REM are reported values from DuBois et al. (2013), so results on MID and Facebook are not available. DLS does not scale to Facebook; ADM4 does not scale beyond Reality.

Model	Reality	Enron	MID	Facebook
MULCH	**−3.82**	**−5.13**	**−3.53**	**−6.82**
CHIP	−4.83	−5.61	−3.67	−9.46
BHM	−5.37	−7.49	−5.33	−14.4
DLS	−5.74	−7.75	−5.52	
REM	−6.11	−6.84		
ADM4	−8.52			

Table 4. Dynamic link prediction AUC for each real network dataset across all models. Mean (standard deviation) of AUC over 100 random short time windows is shown. Bold entry denotes highest mean link prediction AUC for a dataset.

Model	Reality	Enron	MID
MULCH	**0.954(.036)**	0.852(.006)	0.968(.023)
CHIP	0.931(.033)	0.792(.005)	0.966(.030)
BHM	0.951(.035)	0.846(.005)	0.973(.022)
DLS	0.935(.034)	**0.872(.001)**	**0.981(.013)**

et al. (2020). For the DLS model, we randomly sample latent positions for new nodes from a multivariate Gaussian.

We consider two evaluation metrics previously established in the literature. The first is the mean test data log-likelihood per event (DuBois et al., 2013; Arastuie et al., 2020). We use the same train and test splits as in DuBois et al. (2013) for the Reality and Enron datasets, which allows us to compare against their reported results. The second evaluation metric is the area under the receiver operating characteristic curve (AUC) for dynamic link prediction. Specifically, we adopt the dynamic link prediction setting proposed by Yang et al. (2017). We divide the test set into 100 random short time windows and compute the mean and standard deviation of the link prediction AUC over the 100 windows.

Results and Discussion The predictive abilities of the different models are summarized in Table 3. Notice that MULCH achieves the highest test log-likelihood on all 4 datasets, and by a large margin on the Reality and Facebook data. CHIP performed second best on all of the datasets, indicating the importance of self excitation.

MULCH also includes reciprocal and other excitations, which is partially responsible for the improved predictive ability. We find that using a sum of exponential kernels in MULCH also helps to improve predictive log-likelihood compared to a single exponential kernel in CHIP and BHM.

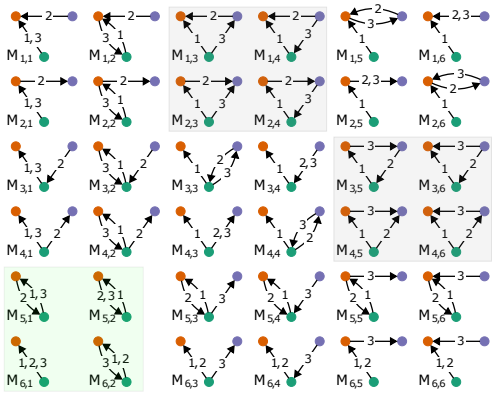


Figure 3. All possible 3-edge temporal motifs. Green and grey shaded boxes denotes 2-node and triangle motifs, respectively. All other motifs are stars. Figure credit: Paranjape et al. (2017).

The importance of the SBM-structured excitation matrix $\boldsymbol{\alpha}$ in MULCH is clearly visible when comparing it to ADM4, which estimates a sparse and low rank $\boldsymbol{\alpha}$ without additional network structure and is not competitive.

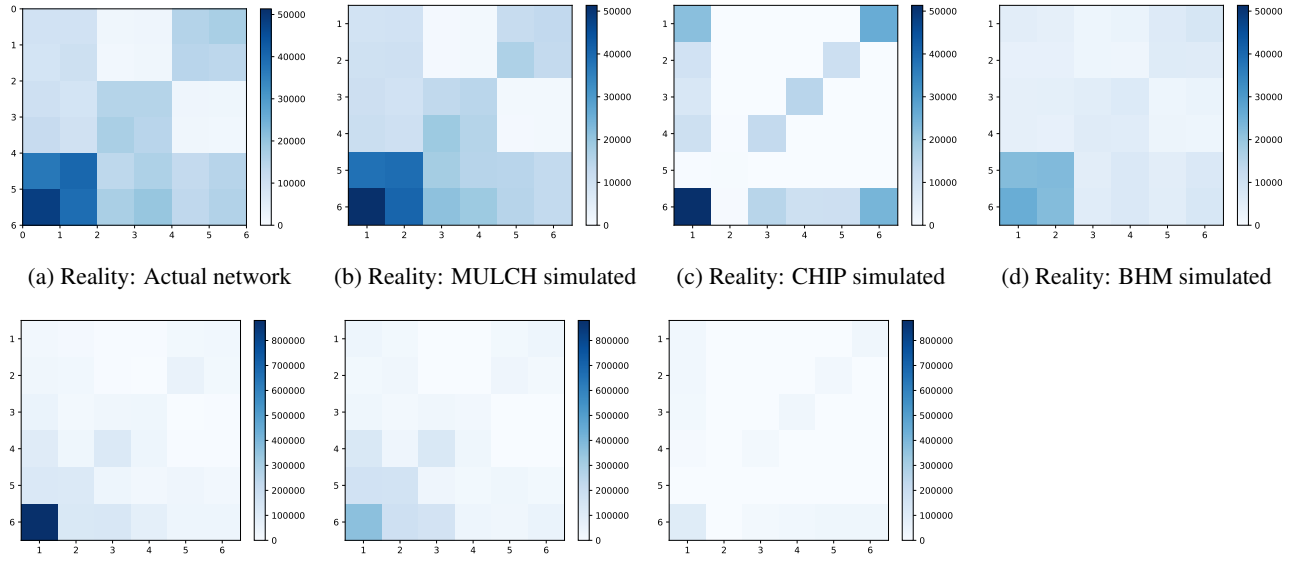
Table 4 shows the dynamic link prediction AUC values[2]. MULCH performs the best on Reality and is competitive on the other two datasets. Notice that the BHM performs better than CHIP in dynamic link prediction AUC, while it was substantially worse than CHIP in test log-likelihood in Table 3. This is due to a difference in the evaluation metrics—test log-likelihood is evaluated on all events, so that repeated events between node pairs are counted multiple times. Conversely, dynamic link prediction considers only whether a pair of nodes had at least a single event within a short time window, so each node pair is only counted once.

5.2.2. Generative Accuracy

Experiment Set-up We evaluate the generative accuracy of our proposed MULCH model by simulating networks from the fitted model and comparing the counts of temporal motifs in the simulated networks to those in the actual network. We consider all 36 possible temporal motifs with 2 or 3 nodes and 3 edges arranged in the same 6×6 matrix as defined by Paranjape et al. (2017). The matrix of different motifs is shown in Figure 3. One would expect a good generative model to replicate the number of temporal motifs observed in the actual network. We consider temporal motifs over 1 week for the Reality and Enron datasets and 1 month for the MID data[3].

[2]We exclude the Facebook data because the dynamic link prediction experiment does not scale to a network of its size.

[3]We do not evaluate generative accuracy on the Facebook data due to its size. We also do not include DLS in this comparison because it resulted in unstable models that cannot be used to generate new networks, also been noted by Huang et al. (2022).



(a) Reality: Actual network (b) Reality: MULCH simulated (c) Reality: CHIP simulated (d) Reality: BHM simulated

(e) Enron: Actual network (f) Enron: MULCH simulated (g) Enron: CHIP simulated

Figure 4. Average temporal motif counts for time window $\delta = 1$ week on 10 simulated networks from MULCH, CHIP, and BHM models fitted on the (a)-(d) Reality and (e)-(g) Enron datasets. The simulations are unstable for the BHM model fit to Enron. Our proposed MULCH model best replicates the temporal motif counts from the actual networks.

Table 5. Mean absolute percentage error (MAPE) on temporal motif counts for each real network dataset across all models. Smaller values indicate better generative ability. Bold entry denotes best fit for a dataset. The BHM fit to the Enron data results in an unstable Hawkes process that cannot simulate networks.

Model	Reality	Enron	MID
MULCH	**16.5**	**32.0**	92.3
CHIP	79.3	74.5	**91.0**
BHM	51.1		97.6

To provide a quantitative assessment of generative ability, we compute the mean absolute percentage error (MAPE) on the temporal motif count matrix. The MAPE is defined by

$$\text{MAPE} = \frac{100}{36} \sum_{i=1}^{6} \sum_{j=1}^{6} \left| \frac{M_{i,j}^{A} - M_{i,j}^{S}}{M_{i,j}^{A}} \right|,$$

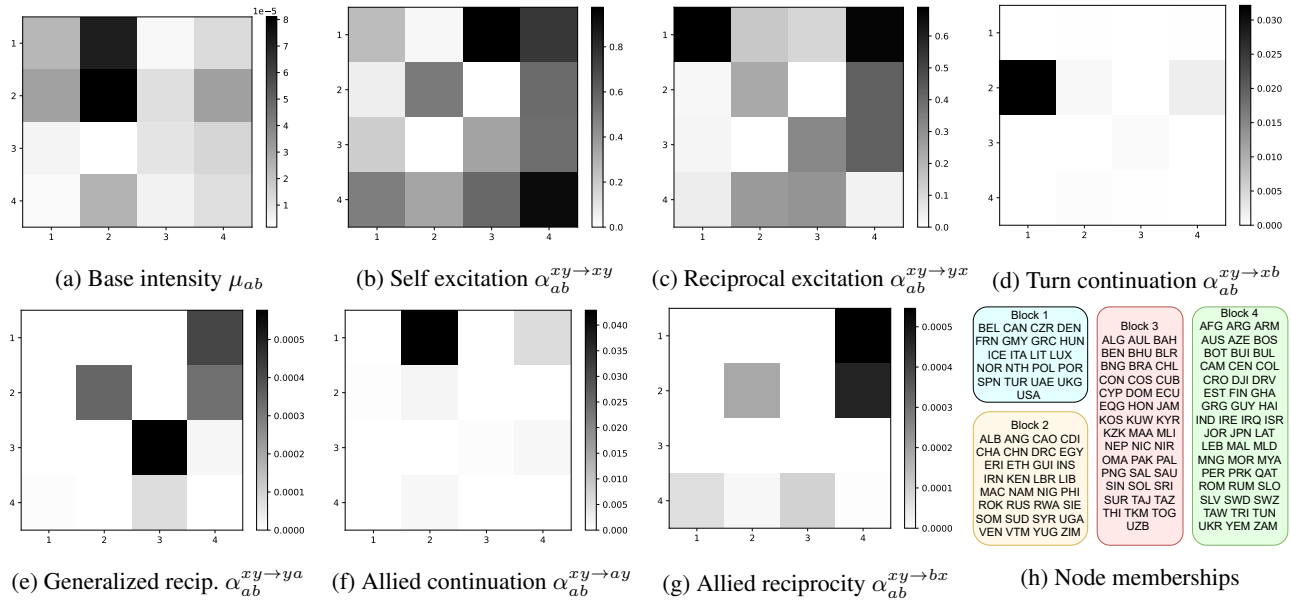
where $M_{i,j}^{A}$ denotes the number of occurrences of motif $M_{i,j}$ in the actual network, and $M_{i,j}^{S}$ denotes the mean number of occurrences of the motif over 10 simulated networks.

Results and Discussion The MAPE values for the different models and datasets are shown in Table 5. MULCH is by far the best at replicating temporal motif counts on Reality and Enron, while all of the models struggle on the MID data. Do & Xu (2021) found that the majority of motifs occurred during several major international conflicts in the years 1999 and 2000, which are not accurately replicated by any of the models. More sophisticated models that incorporate change points may be required to capture these dynamics.

In Figure 4, we show a comparison of the average temporal motif counts on simulated networks from the MULCH, CHIP, and BHM model fits to the Reality and Enron datasets to the actual counts. Notice that CHIP, which uses only self excitation, does not generate motifs with reciprocated edges with any appreciable frequency. The BHM uses self excitation at the block level and then randomly assigns events to node pairs, which results in the uniform-like distribution of 3-node temporal motifs seen in Figure 4d. It generates a wide variety of motifs, but not at frequencies similar to the actual network. On the other hand, our proposed MULCH model generates both a variety of different temporal motifs and at relative frequencies similar to the actual network, as shown by the similarities of the checkerboard patterns between Figures 4a and 4b as well as Figures 4e and 4f for the Enron data. This indicates that the added excitations in MULCH that create dependence between node pairs can indeed replicate higher-order motifs found in real networks.

6. Case Study

We now present a case study on the Militarized Interstate Disputes (MID) dataset. A node denotes a (sovereign) state. An edge denotes a threat, display, or use of force one state directs towards another. We apply MULCH to perform model-based exploratory analysis using the different excitations to reveal insights into behaviors of different states.



(a) Base intensity μ_{ab} (b) Self excitation $\alpha_{ab}^{xy\to xy}$ (c) Reciprocal excitation $\alpha_{ab}^{xy\to yx}$ (d) Turn continuation $\alpha_{ab}^{xy\to xb}$

(e) Generalized recip. $\alpha_{ab}^{xy\to ya}$ (f) Allied continuation $\alpha_{ab}^{xy\to ay}$ (g) Allied reciprocity $\alpha_{ab}^{xy\to bx}$ (h) Node memberships

Figure 5. Estimated values for MULCH parameter estimates and node memberships fit to the Military Interstate Disputes data. Nodes are represented by their 3-letter country codes from The Correlates of War Project (2021).

Our parameter and block estimates are shown in Figure 5. From examining the magnitudes of the different excitations, one can see that self and reciprocal excitation are the strongest, with highest α values around 0.9 and 0.7, respectively. Next strongest are allied continuation and turn continuation, with highest α values around 0.04 and 0.03, respectively, which is one order of magnitude weaker. Generalized reciprocity and allied reciprocity have much lower α values, with the highest around 0.0005 for both, two orders of magnitude weaker than allied and turn continuation.

Examining the combination of node memberships and parameter estimates also reveals some interesting insights. Events in this network correspond to disputes between states, and we find that most events occur between blocks. Every member of block 1 is a member state of the North Atlantic Treaty Organization (NATO) except the UAE, which has partnered with NATO in several disputes, including the intervention in Libya in 2011.

We observe that block pair $(1, 2)$ has by far the highest excitation for allied continuation, indicating that nodes in block 1 (NATO members) tend to jointly engage states in block 2. By consulting the narratives that accompany the MID dataset, indeed we find that two of the most prominent disputes involving NATO correspond to the NATO bombing of Yugoslavia in 1999 and the Libya intervention in 2011. Both Yugoslavia (YUG) and Libya (LIB) are in block 2, so the high allied continuation $\alpha_{1,2}^{xy\to ay}$ is correctly modeling these incidents. Note also that block pair $(2, 1)$ has the highest excitation for turn continuation. This is dominated again by the dispute between NATO and Yugoslavia, with Yugoslavia threatening multiple NATO states in rapid succession.

We present this case study to illustrate the type of analysis that MULCH can be used for. Our analysis is exploratory rather than confirmatory, and we caution against jumping to conclusions about the behaviors of states from our results.

7. Conclusion

We proposed the multivariate community Hawkes (MULCH) model for continuous-time dynamic networks and demonstrated that it is superior to existing models both in terms of predictive and generative abilities on several real network datasets. The main innovation in our model is introducing dependence between node pairs in a tractable manner by using multivariate Hawkes processes with a structured excitation matrix $\boldsymbol{\alpha}$ inspired by the SBM. In addition to self and reciprocal excitation, we also incorporated excitations motivated by sociological concepts of turn continuing and generalized reciprocity, which can replicate higher-order temporal motifs. We emphasize that these are not the only types of excitations that can be incorporated into our modeling framework—an investigation of other potential excitations would be a useful avenue for future research.

Acknowledgements

This material is based upon work supported by the National Science Foundation grants IIS-1755824, DMS-1830412, IIS-2047955, and DMS-1830547.

References

Amini, A. A., Chen, A., Bickel, P. J., and Levina, E. Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics*, 41(4):2097–2122, 2013.

Arastuie, M., Paul, S., and Xu, K. S. CHIP: A Hawkes process model for continuous-time networks with scalable and consistent estimation. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Bacry, E., Mastromatteo, I., and Muzy, J.-F. Hawkes processes in finance. *Market Microstructure and Liquidity*, 1 (01):1550005, 2015.

Bacry, E., Bompaire, M., Deegan, P., Gaïffas, S., and Poulsen, S. tick: a Python library for statistical learning, with an emphasis on Hawkes processes and time-dependent models. *Journal of Machine Learning Research*, 18(1):7937–7941, 2017.

Benson, A. R., Gleich, D. F., and Leskovec, J. Higher-order organization of complex networks. *Science*, 353(6295): 163–166, 2016.

Blundell, C., Beck, J., and Heller, K. A. Modelling reciprocating relationships with Hawkes processes. In *Advances in Neural Information Processing Systems 25*, pp. 2600–2608, 2012.

Bollobás, B., Janson, S., and Riordan, O. Sparse random graphs with clustering. *Random Structures & Algorithms*, 38(3):269–323, 2011.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

Chaudhuri, K., Chung, F., and Tsiatas, A. Spectral clustering of graphs with general degrees in the extended planted partition model. In *Proceedings of the 25th Annual Conference on Learning Theory*, pp. 35.1–35.23, 2012.

Chen, S., Liu, S., and Ma, Z. Global and individualized community detection in inhomogeneous multilayer networks. *arXiv preprint arXiv:2012.00933*, 2020.

Chin, P., Rao, A., and Vu, V. Stochastic block model and community detection in sparse graphs: A spectral algorithm with optimal rate of recovery. In *Proceedings of the 28th Conference on Learning Theory*, pp. 391–423, 2015.

Corneli, M., Latouche, P., and Rossi, F. Multiple change points detection and clustering in dynamic networks. *Statistics and Computing*, 28(5):989–1007, 2018.

Do, H. N. and Xu, K. S. Analyzing escalations in militarized interstate disputes using motifs in temporal networks. In *Proceedings of the 10th International Conference on Complex Networks and Their Applications*, pp. 527–538, 2021.

DuBois, C., Butts, C. T., and Smyth, P. Stochastic blockmodeling of relational event dynamics. In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics*, pp. 238–246, 2013.

Eagle, N., Pentland, A. S., and Lazer, D. Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36): 15274–15278, 2009.

Farajtabar, M., Wang, Y., Rodriguez, M. G., Li, S., Zha, H., and Song, L. COEVOLVE: A joint point process model for information diffusion and network co-evolution. In *Advances in Neural Information Processing Systems 28*, pp. 1945–1953, 2015.

Gao, C., Ma, Z., Zhang, A. Y., and Zhou, H. H. Achieving optimal misclassification proportion in stochastic block models. *The Journal of Machine Learning Research*, 18 (1):1980–2024, 2017.

Gao, C., Ma, Z., Zhang, A. Y., and Zhou, H. H. Community detection in degree-corrected block models. *The Annals of Statistics*, 46(5):2153–2185, 2018.

Gibson, D. R. Participation shifts: Order and differentiation in group conversation. *Social Forces*, 81(4):1335–1380, 2003.

Gibson, D. R. Taking turns and talking ties: Networks and conversational interaction. *American Journal of Sociology*, 110(6):1561–1597, 2005.

Goldenberg, A., Zheng, A. X., Fienberg, S. E., and Airoldi, E. M. A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2(2):129–233, 2010.

Han, Q., Xu, K. S., and Airoldi, E. Consistent estimation of dynamic and multi-layer block models. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 1511–1520, 2015.

Hawkes, A. G. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

Hawkes, A. G. Hawkes processes and their applications to finance: a review. *Quantitative Finance*, 18(2):193–198, 2018.

Hoff, P. D., Raftery, A. E., and Handcock, M. S. Latent space approaches to social network analysis. *Journal of*

the American Statistical Association, 97(460):1090–1098, 2002.

Holland, P. W., Laskey, K. B., and Leinhardt, S. Stochastic blockmodels: First steps. *Social Networks*, 5(2):109–137, 1983.

Huang, Z., Soliman, H., Paul, S., and Xu, K. S. A mutually exciting latent space Hawkes process model for continuous-time networks. *arXiv preprint arXiv:2205.09263*, 2022.

Hubert, L. and Arabie, P. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

Junuthula, R., Haghdan, M., Xu, K. S., and Devabhaktuni, V. The block point process model for continuous-time event-based dynamic networks. In *Proceedings of the World Wide Web Conference*, pp. 829–839, 2019.

Klimt, B. and Yang, Y. The Enron corpus: A new dataset for email classification research. In *Proceedings of the 15th European Conference on Machine Learning*, pp. 217–226, 2004.

Laub, P. J., Lee, Y., and Taimre, T. *The Elements of Hawkes Processes*. Springer Nature, 2021.

Lei, J. and Rinaldo, A. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1): 215–237, 2015.

Linderman, S. W. and Adams, R. P. Discovering latent network structure in point process data. In *Proceedings of the 31st International Conference on Machine Learning*, pp. 1413–1421, 2014.

Masuda, N., Takaguchi, T., Sato, N., and Yano, K. Self-exciting point process modeling of conversation event sequences. In *Temporal Networks*, pp. 245–264. Springer, 2013.

Matias, C., Rebafka, T., and Villers, F. A semiparametric extension of the stochastic block model for longitudinal networks. *Biometrika*, 105(3):665–680, 2018.

Miscouridou, X., Caron, F., and Teh, Y. W. Modelling sparsity, heterogeneity, reciprocity and community structure in temporal interaction data. In *Advances in Neural Information Processing Systems*, volume 31, pp. 2343–2352, 2018.

Nowicki, K. and Snijders, T. A. B. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

Palmer, G., McManus, R. W., D'Orazio, V., Kenwick, M. R., Karstens, M., Bloch, C., Dietrich, N., Kahn, K., Ritter, K., and Soules, M. J. The MID5 dataset, 2011–2014: Procedures, coding rules, and description. *Conflict Management and Peace Science*, 39(4):470–482, 2022.

Paranjape, A., Benson, A. R., and Leskovec, J. Motifs in temporal networks. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*, pp. 601–610, 2017.

Paul, S., Milenkovic, O., and Chen, Y. Higher-order spectral clustering under superimposed stochastic block model. *arXiv preprint arXiv:1812.06515*, 2018.

Peixoto, T. P. Disentangling homophily, community structure, and triadic closure in networks. *Physical Review X*, 12:011004, 2022.

Qin, T. and Rohe, K. Regularized spectral clustering under the degree-corrected stochastic blockmodel. In *Advances in Neural Information Processing Systems 26*, pp. 3120–3128, 2013.

Rasmussen, J. G. Lecture notes: Temporal point processes and the conditional intensity function. *arXiv preprint arXiv:1806.00221*, 2018.

Rohe, K., Chatterjee, S., and Yu, B. Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4):1878–1915, 2011.

Rohe, K., Qin, T., and Yu, B. Co-clustering directed graphs to discover asymmetries and directional communities. *Proceedings of the National Academy of Sciences*, 113 (45):12679–12684, 2016.

Santos, T., Lemmerich, F., and Helic, D. Surfacing estimation uncertainty in the decay parameters of Hawkes processes with exponential kernels. *arXiv preprint arXiv:2104.01029*, 2021.

Sussman, D. L., Tang, M., Fishkind, D. E., and Priebe, C. E. A consistent adjacency spectral embedding for stochastic blockmodel graphs. *Journal of the American Statistical Association*, 107(499):1119–1128, 2012.

The Correlates of War Project. COW country codes, 2021. URL https://correlatesofwar.org/data-sets/cow-country-codes.

Tran, L., Farajtabar, M., Song, L., and Zha, H. NetCodec: Community detection from individual activities. In *Proceedings of the SIAM International Conference on Data Mining*, pp. 91–99, 2015.

Viswanath, B., Mislove, A., Cha, M., and Gummadi, K. P. On the evolution of user interaction in Facebook. In *Proceedings of the 2nd ACM Workshop on Online Social Networks*, pp. 37–42, 2009.

Xin, L., Zhu, M., and Chipman, H. A continuous-time stochastic block model for basketball networks. *The Annals of Applied Statistics*, 11(2):553–597, 2017.

Xu, S., Morse, S., and González, M. C. Modeling human dynamics and lifestyle using digital traces. *arXiv preprint arXiv:2005.06542*, 2020.

Yamagishi, T. and Kiyonari, T. The group as the container of generalized reciprocity. *Social Psychology Quarterly*, 63(2):116–132, 2000.

Yang, J., Rao, V., and Neville, J. Decoupling homophily and reciprocity with latent space network models. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2017.

Zhou, K., Zha, H., and Song, L. Learning triggering kernels for multi-dimensional Hawkes processes. In *Proceedings of the 30th International Conference on Machine Learning*, pp. 1301–1309, 2013a.

Zhou, K., Zha, H., and Song, L. Learning social infectivity in sparse low-rank networks using multi-dimensional Hawkes processes. In *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics*, pp. 641–649, 2013b.

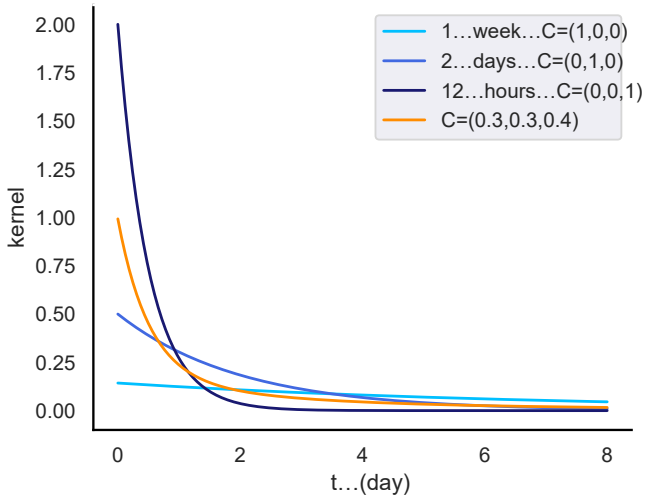


Figure 6. Sum of exponential kernels shapes at $\boldsymbol{\beta} = (1 \text{ week}, 2 \text{ days}, 12 \text{ hours})$ and different $\boldsymbol{C}^q_{ab}$ values.

A. Additional Details on MULCH Model and Estimation Procedure

A.1. Sum of Kernels Hawkes Process Intensity

By allowing different block pairs to have different scaling parameters C^q_{ab}, their intensities can decay at different rates, as in the CHIP (Arastuie et al., 2020) and BHM (Junuthula et al., 2019) models where a single β_{ab} value was estimated for each block pair. As shown in Figure 6, $\boldsymbol{C}^q_{ab}$ controls decay rate, and also, allows every block pair to have distinct kernel shape. The intensity of $(i, j) \in \text{bp}(a, b)$ becomes

$$\lambda_{ij}(t) = \mu_{ab} + \sum_{\substack{(x,y)\in \text{bp}(a,b) \\ (x,y)\in \text{bp}(b,a)}} \alpha^{xy\to ij}_{ab} \sum_{t_s \in T_{xy}} \sum_{q=1}^{Q} C^q_{ab} \beta_q e^{-\beta_q (t-t_s)}. \tag{2}$$

A.2. Full Log-likelihood Function

Based on our intensity function (2) and the log-likelihood function for exponential Hawkes processes, we can derive our block pair wise log-likelihood function to be

$$\ell_{ab}(\boldsymbol{\theta}_{ab}|\boldsymbol{Z}, \mathcal{H}_t) = \sum_{Z_i=a, Z_j=b, i\neq j} \left\{ -\mu_{ab} T - \sum_{\substack{(x,y)\in \text{bp}(a,b) \\ (x,y)\in \text{bp}(b,a)}} \alpha^{xy\to ij}_{ab} \sum_{t_s \in T_{xy}} \sum_{q=1}^{Q} C^q_{ab} \left[\left(1 - e^{-\beta_q (T-t_s)} \right) \right] \right.$$
$$\left. + \sum_{t_s \in T_{ij}} \ln \left[\mu_{ab} + \sum_{\substack{(x,y)\in \text{bp}(a,b) \\ (x,y)\in \text{bp}(b,a)}} \alpha^{xy\to ij}_{ab} \sum_{q=1}^{Q} C^q_{ab} \beta_q R^q_{xy\to ij}(t_s) \right] \right\}$$

where

$$R^q_{xy\to ij}(t_s) = \sum_{\substack{t_r \in T_{xy} \\ t_r < t_s}} e^{-\beta_q (t_s - t_r)}$$

can be computed recursively (Arastuie et al., 2020).

A.3. Block Structure in Expected Count Matrix

As shown in Figure 1c, the expected count matrix $E[\boldsymbol{N}(T)]$ follows a block structure. This block structure is stated explicitly in Theorem 4.1, and the proof is below.

Proof of Theorem 4.1. When the process is stationary, we can derive the vectorized version of the expected intensity as $vec(\boldsymbol{\lambda}) = (\boldsymbol{I} - \boldsymbol{\Gamma})^{-1}\boldsymbol{\mu}$, where all node pairs in a block pair (a, b) are kept together and they are ordered such that block pairs (a, b), (b, a) occupy consecutive positions. Then $\boldsymbol{\Gamma}$ is a $n(n-1) \times n(n-1)$ matrix whose elements are $\Gamma_{(i,j)\to(x,y)} = \alpha_{ab}^{ij\to xy}$ (Hawkes, 1971) and it has a diagonal block structure, i.e.,

$$\boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{\Gamma}_{(a,b)} & & \\ & \boldsymbol{\Gamma}_{(a',b')} & \\ & & \dots \end{pmatrix}$$

where $\boldsymbol{\Gamma}_{(a,b)}$ contains the rows $(i, j) \to (x, y)$ such that $(i, j) \in (a, b)$ or (b, a) and $(x, y) \in (a, b)$ or (b, a). This is because $\Gamma_{(i,j)\to(x,y)} = 0$ for any $(i, j) \in (a, b)$ and $(x, y) \notin \{(a, b), (b, a)\}$. Denote $\boldsymbol{G}_{(a,b)} = \boldsymbol{I} - \boldsymbol{\Gamma}_{(a,b)}$, where $\boldsymbol{I}$ is the identity matrix. Then,

$$\boldsymbol{I} - \boldsymbol{\Gamma} = \begin{pmatrix} \boldsymbol{G}_{(a,b)} & & \\ & \boldsymbol{G}_{(a',b')} & \\ & & \dots \end{pmatrix}$$

also have the diagonal block structure, and so does

$$(\boldsymbol{I} - \boldsymbol{\Gamma})^{-1} = \begin{pmatrix} \boldsymbol{G}_{(a,b)}^{-1} & & \\ & \boldsymbol{G}_{(a',b')}^{-1} & \\ & & \dots \end{pmatrix}$$

By our construction of the $\boldsymbol{\Gamma}$, we can write the $\boldsymbol{G}_{(a,b)}$ as a block matrix:

$$\boldsymbol{G}_{(a,b)} = \begin{pmatrix} \boldsymbol{G}_{ab\to ab} & \boldsymbol{G}_{ab\to ba} \\ \boldsymbol{G}_{ba\to ab} & \boldsymbol{G}_{ba\to ba} \end{pmatrix}$$

and in the same block matrix, the rows have the same row sum.

Note for a block matrix, the inversion is

$$\begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{pmatrix}^{-1} = \begin{pmatrix} \left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1} & -\left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1}\mathbf{B}\mathbf{D}^{-1} \\ -\mathbf{D}^{-1}\mathbf{C}\left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1} & \mathbf{D}^{-1} + \mathbf{D}^{-1}\mathbf{C}\left(\mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1}\mathbf{B}\mathbf{D}^{-1} \end{pmatrix}$$

We summarize a few simple observations in the following proposition.

Proposition A.1. *For any square matrix $\boldsymbol{A}$ the following holds.*

1. *If $\boldsymbol{A}\mathbf{1} = a\mathbf{1}$, i.e., if the row sum of $\boldsymbol{A}$ are identical, and if $\boldsymbol{A}^{-1}$ exists, then $\boldsymbol{A}^{-1}\mathbf{1} = a^{-1}\mathbf{1}$, i.e., the row sum of $\boldsymbol{A}^{-1}$ are also identical.*
2. *If $\boldsymbol{A}\mathbf{1} = a\mathbf{1}$ and $\boldsymbol{B}\mathbf{1} = b\mathbf{1}$, then $\boldsymbol{A}\boldsymbol{B}\mathbf{1} = ab\mathbf{1}$*
3. *If $\boldsymbol{A}\mathbf{1} = a\mathbf{1}$ and $\boldsymbol{B}\mathbf{1} = b\mathbf{1}$, then $(\boldsymbol{A} - \boldsymbol{B})\mathbf{1} = (a - b)\mathbf{1}$*

Thus, since $\boldsymbol{G}_{ab\to ab}, \boldsymbol{G}_{ab\to ba}, \boldsymbol{G}_{ba\to ab}, \boldsymbol{G}_{ba\to ba}$ have the same row sum, and if all of them and $\boldsymbol{G}_{(a,b)}$ are invertible, then using the proposition above, we have

$$\begin{aligned} \boldsymbol{G}_{(a,b)}^{-1}\begin{pmatrix} \mathbf{1}_{n_a n_b} \\ \mathbf{0}_{n_a n_b} \end{pmatrix} &= \begin{pmatrix} \left(\mathbf{G}_{ab\to ab} - \mathbf{G}_{ab\to ba}{\mathbf{G}_{ba\to ba}}^{-1}\mathbf{G}_{ba\to ab}\right)^{-1}\mathbf{1}_{n_a n_b} \\ -\mathbf{G}_{ba\to ba}^{-1}\mathbf{G}_{ba\to ab}\left(\mathbf{G}_{ab\to ab} - \mathbf{G}_{ab\to ba}{\mathbf{G}_{ba\to ba}}^{-1}\mathbf{G}_{ba\to ab}\right)^{-1}\mathbf{1}_{n_a n_b} \end{pmatrix} \\ &= \begin{pmatrix} g_{ab}^1\mathbf{1}_{n_a n_b} \\ g_{ab}^2\mathbf{1}_{n_a n_b} \end{pmatrix} \end{aligned}$$

where g^1_{ab}, g^2_{ab} are some real valued functions of the parameters $\boldsymbol{\alpha}_{ab}, \boldsymbol{\alpha}_{ba}$, and n_a denotes the number of nodes in block a. Similarly,

$$\boldsymbol{G}^{-1}_{(a,b)} \begin{pmatrix} \mathbf{0}_{n_a n_b} \\ \mathbf{1}_{n_a n_b} \end{pmatrix} = \begin{pmatrix} g^1_{ba}\mathbf{1}_{n_a n_b} \\ g^2_{ba}\mathbf{1}_{n_a n_b} \end{pmatrix}$$

Thus, by our construction,

$$\text{vec}(\boldsymbol{\lambda}_{(i,j)|i\in a, j\in b \text{ or } i\in b, j\in a}) = \boldsymbol{G}^{-1}_{(a,b)} \begin{pmatrix} \mu_{ab} \\ \vdots \\ \mu_{ab} \\ \mu_{ba} \\ \vdots \\ \mu_{ba} \end{pmatrix} \tag{3}$$

$$= \boldsymbol{G}^{-1}_{(a,b)} \left(\begin{pmatrix} \mathbf{1}_{n_a n_b} \\ \mathbf{0}_{n_a n_b} \end{pmatrix} \mu_{ab} + \begin{pmatrix} \mathbf{0}_{n_a n_b} \\ \mathbf{1}_{n_a n_b} \end{pmatrix} \mu_{ba} \right) \tag{4}$$

$$= \begin{pmatrix} (g^1_{ab}\mu_{ab} + g^1_{ba}\mu_{ba})\mathbf{1}_{n_a n_b} \\ (g^2_{ab}\mu_{ab} + g^2_{ba}\mu_{ba})\mathbf{1}_{n_a n_b} \end{pmatrix} \tag{5}$$

which means for any $(i,j) \in (a,b)$, $\lambda_{(i,j)} = (g^1_{ab}\mu_{ab} + g^1_{ba}\mu_{ba})$, and for any $(i',j') \in (b,a)$, $\lambda_{(i',j')} = (g^2_{ab}\mu_{ab} + g^2_{ba}\mu_{ba})\mathbf{1}_{n_a n_b}$. □

B. Additional Experiment Details and Results

B.1. Spectral Clustering and Refinement Accuracy

For these experiments, we generate data from the MULCH model with $K = 4$ and assume parameters of the four diagonal block pairs are equal, and similarly, parameters of the off-diagonal block pairs are equal.

We denote the set of Hawkes process parameters for a block pair (a, b) by

$$\boldsymbol{\theta}_{ab} = \left(\mu_{ab}, \alpha^{xy\to xy}_{ab}, \alpha^{xy\to yx}_{ab}, \alpha^{xy\to xb}_{ab}, \alpha^{xy\to ya}_{ab}, \alpha^{xy\to ay}_{ab}, \alpha^{xy\to bx}_{ab}\right).$$

In these experiments, we use the following parameters:

$$\begin{aligned} \boldsymbol{\theta}_{aa} = \boldsymbol{\theta}_{bb} &= (0.008, 0.3, 0.3, 0.002, 0.0005, 0.001, 0.0005) \\ \boldsymbol{\theta}_{ab} = \boldsymbol{\theta}_{ba} &= (0.008, 0.1, 0.1, 0.001, 0.0001, 0.001, 0.0001) \\ \boldsymbol{C}_{aa} = \boldsymbol{C}_{ab} = \boldsymbol{C}_{ba} = \boldsymbol{C}_{bb} &= (0.33, 0.33, 0.34) \\ \boldsymbol{\beta} &= (2 \text{ weeks}, 1 \text{ day}, 2 \text{ hours}) = (1/14, 1, 24/2) \end{aligned}$$

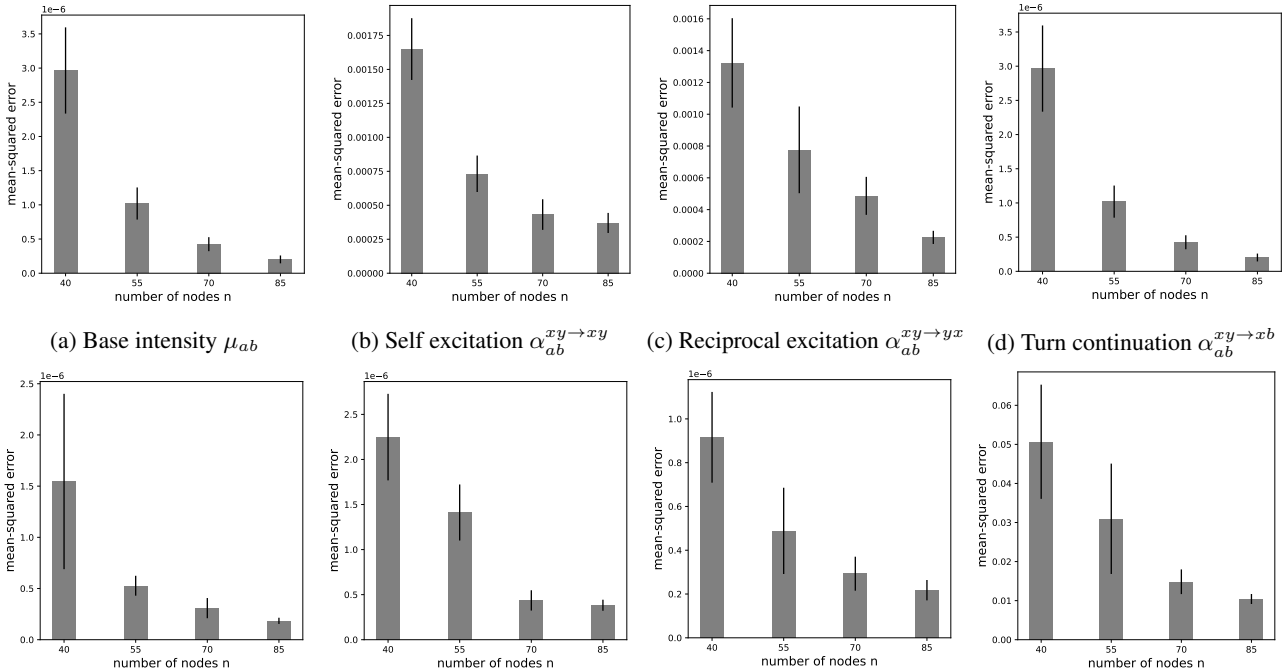
Note that, simulated networks have assortative mixing, and we assumed timestamps are in unit of days.

B.2. Parameter Estimation Accuracy

We test the accuracy of our MLE in this experiment. We generate data from the MULCH model with $K = 2$ and assume same structured parameters as in B.1. However, we switch diagonal and off-diagonal block pairs parameters so simulated networks are disassortative:

$$\begin{aligned} \boldsymbol{\theta}_{aa} = \boldsymbol{\theta}_{bb} &= (0.008, 0.1, 0.1, 0.001, 0.0001, 0.001, 0.0001) \\ \boldsymbol{\theta}_{ab} = \boldsymbol{\theta}_{ba} &= (0.008, 0.3, 0.3, 0.002, 0.0005, 0.001, 0.0005) \end{aligned}$$

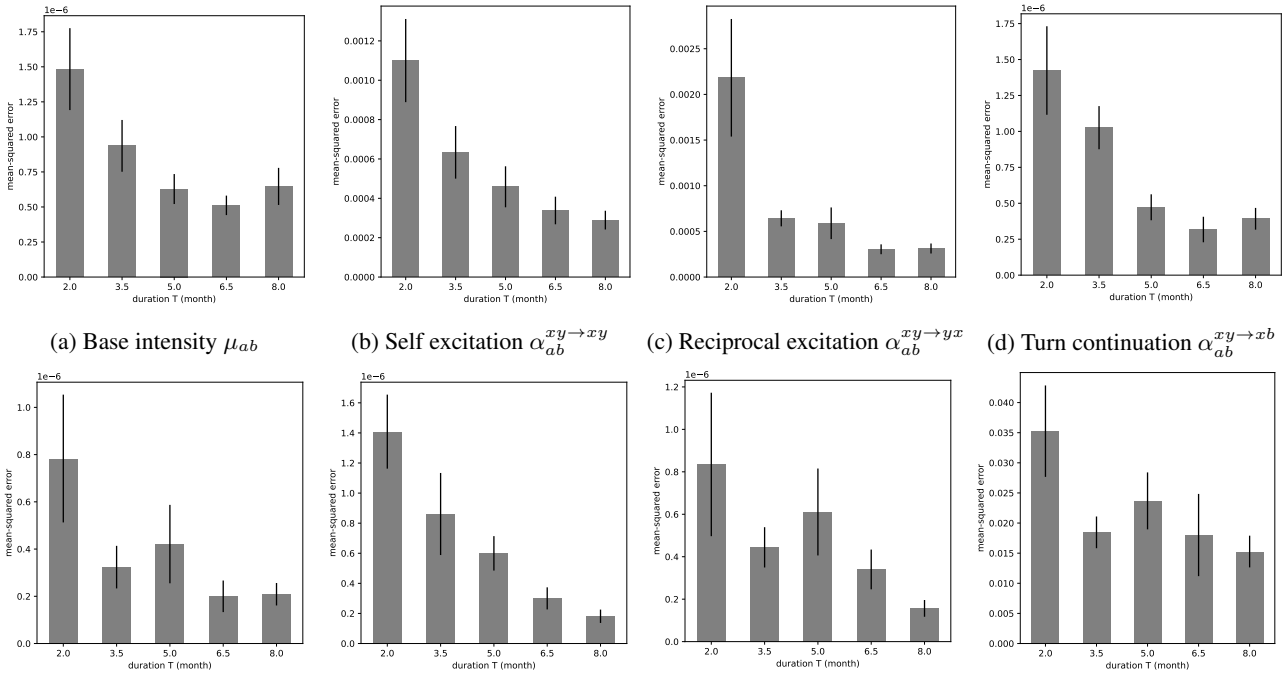
At a fixed duration $T = 5$ months, and while varying number of nodes n, we simulate and fit our model to evaluate the accuracy of the different Hawkes process parameters. As shown in Figure 7, the mean-squared error (MSE) for each estimated parameter decreases with increasing n as expected. Similarly, for a fixed number of nodes $n = 70$, we vary the duration T. Figure 8 shows that the MSE decreases with increasing T as expected. For each pair values of (n, T), we run 10 simulations.



(a) Base intensity μ_{ab} (b) Self excitation $\alpha_{ab}^{xy\to xy}$ (c) Reciprocal excitation $\alpha_{ab}^{xy\to yx}$ (d) Turn continuation $\alpha_{ab}^{xy\to xb}$

(e) Generalized recip. $\alpha_{ab}^{xy\to ya}$ (f) Allied continuation $\alpha_{ab}^{xy\to ay}$ (g) Allied reciprocity $\alpha_{ab}^{xy\to bx}$ (h) Kernel scaling C_{ab}

Figure 7. Mean squared error for Hawkes process parameters on simulated networks at $T = 5$ months ($\pm$ standard error over 10 runs).



(a) Base intensity μ_{ab} (b) Self excitation $\alpha_{ab}^{xy\to xy}$ (c) Reciprocal excitation $\alpha_{ab}^{xy\to yx}$ (d) Turn continuation $\alpha_{ab}^{xy\to xb}$

(e) Generalized recip. $\alpha_{ab}^{xy\to ya}$ (f) Allied continuation $\alpha_{ab}^{xy\to ay}$ (g) Allied reciprocity $\alpha_{ab}^{xy\to bx}$ (h) Kernel scaling C_{ab}

Figure 8. Mean squared error for Hawkes process parameters on simulated networks at $n = 70$ ($\pm$ standard error over 10 runs).

B.3. Real Dataset Descriptions

Each dataset consists of a set of events where each event is denoted by a sender, a receiver, and a timestamp:

- MIT Reality Mining (Eagle et al., 2009): We use the phone call data, where the start time of each call was used as the event timestamp. We consider calls between pairs of the core 70 callers and recipients. We use β values of 1 week, 2 days, and 1 hour.
- Enron Emails (Klimt & Yang, 2004): We use the same subset of the Enron email corpus as in DuBois et al. (2013). We use β values of 1 week, 2 days, and 6 hours.
- Militarized Interstate Disputes (MID) (Palmer et al., 2022): We use the MID 5.01 dataset compiled by the Correlates of War project. Nodes denote (sovereign) states. Each edge denotes an incident, which is a threat, display, or use of force one state directs towards another. We remove 8 nodes that are disconnected from the largest connected component. Unlike the other networks, this is a conflict network, so we expect to see *disassortative* rather than assortative mixing between nodes. We use β values of 2 months, 2 weeks, and 12 hours.
- Facebook Wall Posts (Viswanath et al., 2009): We consider only posts from a user to another user's wall so that there are no self-edges. We analyze the largest connected component of the network excluding self loops. We use β values of 2 months, 1 week, and 2 hours.

The MIT Reality Mining, Enron, and Facebook datasets were loaded and preprocessed identically to Arastuie et al. (2020). Timestamps in all datasets were rescaled to be in the range $[0, 1000]$, the same as in Arastuie et al. (2020) and DuBois et al. (2013) so that log-likelihoods are comparable with their reported figures.

B.4. Descriptions of Other Models for Comparison

We compare against several other TPP models for continuous-time networks:

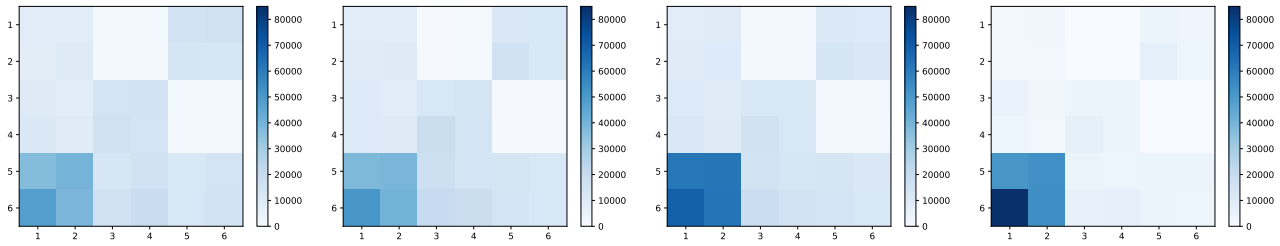
- Community Hawkes Independent Pairs (CHIP) (Arastuie et al., 2020): Univariate Hawkes process network model with block structure where each node pair is independent of all others. We use the implementation at `https://github.com/IdeasLabUT/CHIP-Network-Model`
- Block Hawkes Model (BHM) (Junuthula et al., 2019): Univariate Hawkes process network model with block structure where an event between a node pair equally excites all node pairs in the same block pair. We use the implementation at `https://github.com/IdeasLabUT/CHIP-Network-Model`
- Relational Event Model (REM) (DuBois et al., 2013): Inhomogeneous Poisson process network model with piecewise constant intensities and block structure. The instantaneous intensity for a node pair depends on several network summary statistics in a manner similar to an exponential random graph model (Goldenberg et al., 2010). We were not able to locate a working implementation so we only compare against reported results.
- Dual Latent Space (DLS) (Yang et al., 2017): Bivariate Hawkes process network model with separate continuous latent spaces for the base intensities and for reciprocal excitations. We use the implementation at `https://github.com/jiaseny/lspp`
- ADM4 (Zhou et al., 2013b): Multivariate Hawkes process network model with penalties to encourage sparsity and low rank for the excitation matrix $\boldsymbol{\alpha}$. We use the implementation in the Python package `tick` (Bacry et al., 2017).

B.5. Scalability of MULCH

We compare scalability of MULCH against the CHIP and BHM models. We fit all 3 models to the Facebook ($n = 43,953$) and MID ($n = 147$) datasets (see Table 2 for other dataset statistics). We run the models without likelihood refinement over a range of number of blocks K and report the wall clock time required to fit to the model at K corresponding to the best test log-likelihood score. The wall clock times and test log-likelihoods are both shown in Table 6. Both CHIP and BHM, which utilize univariate Hawkes processes, are much faster to fit than our proposed MULCH model that uses multivariate Hawkes processes. However, MULCH is able to achieve better fits, as indicated by the higher test log-likelihood.

Table 6. Wall clock times to fit CHIP, BHM, and MULCH on the Facebook and MID datasets. Test ℓ denotes the mean test log-likelihood per event as defined in Section 5.2.1.

	CHIP			BHM			MULCH		
Dataset	Test ℓ	K	Time	Test ℓ	K	Time	Test ℓ	K	Time
Facebook	-9.48	9	3.8 minutes	-14.3	15	3.5 minutes	-6.82	1	16 hours
MID	-3.67	2	0.48 seconds	-5.18	91	3.5 seconds	-3.53	2	31 seconds



(a) Actual network (b) Six-α (MAPE $= 16.5$) (c) Four-α (MAPE $= 16.1$) (d) Two-α (MAPE $= 69.8$)

Figure 9. Average temporal motif counts for time window $\delta = 1$ week on 10 simulated networks from six-α, four-α, and two-α MULCH models fitted on the Reality Mining dataset.

B.6. Ablation Experiment on Number of Excitation Types

The excitation parameters $\alpha_{ab}^{xy \to ij}$ control which node pairs can mutually excite each other. For the full MULCH model, we consider 6 types of excitations, listed in Table 1. In this experiment, we perform an ablation study by removing some of the excitation types (i.e. set to 0), then refit the model. We test out two additional variants of our MULCH model: two α's (self and reciprocal excitation only) and four α's (add turn continuation and generalized reciprocity). We use the Reality Mining dataset in this experiment.

We find that the full six-α model has slightly better predictive accuracy. Compared to the 0.954 link prediction AUC of the full six-α model, the four-α and two-α models had AUCs of 0.950 and 0.951, respectively. All 3 models had similar test-log likelihood (± 0.01).

We find that the six-α and four-α models significantly improve generative accuracy compared to the two-α model, as shown in Figure 9. The improvement is because the two-α model has no mechanism to generate 3-node motifs, e.g. triangles, and needs to generate way too many 2-node motifs (see Figure 9d, MAPE $= 69.8$) to generate 3-node motif counts close to the actual network in Figure 9a. The full six-α model (Figure 9b) can accurately generate all motif counts due to the additional excitations and achieves a much better MAPE $= 16.5$ score. Note that the four-α model is able to perform well and generate motifs close to actual dataset at MAPE $= 16.1$.