

Spatiotemporal Interactive Modeling of Event-Based Dynamic Networks

Di Wang, Xiaochen Xian & Haidong Li

To cite this article: Di Wang, Xiaochen Xian & Haidong Li (2025) Spatiotemporal Interactive Modeling of Event-Based Dynamic Networks, *Technometrics*, 67:2, 293-310, DOI: [10.1080/00401706.2024.2441679](https://doi.org/10.1080/00401706.2024.2441679)

To link to this article: <https://doi.org/10.1080/00401706.2024.2441679>



[View supplementary material](#)



Published online: 30 Jan 2025.



[Submit your article to this journal](#)



Article views: 280



[View related articles](#)



[View Crossmark data](#)



Spatiotemporal Interactive Modeling of Event-Based Dynamic Networks

Di Wang^a, Xiaochen Xian^b, and Haidong Li^c

^aDepartment of Industrial Engineering and Management, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China; ^bH. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA; ^cDepartment of Management Science, School of Economics and Management, University of Chinese Academy of Sciences, Beijing, China

ABSTRACT

Event-based dynamic networks exist in a wide range of areas, including traffic, biological, and social applications. Such a network consists of interaction event sequences over different locations, where each event may trigger or influence a series of subsequent events under certain intrinsic spatial structure because of their geographical and semantic proximities. Such influence patterns and triggering motivations reflect the nature and semantics of human/object behaviors in the network. Thus, modeling event-based dynamic networks properly is critically important. This article proposes a spatiotemporal interactive Hawkes process (SIHP) that describes how a series of events occurs and models the rate of interaction events between any pair of nodes on the network explicitly with the information from related historical events as well as geographical and semantic neighboring nodes. The proposed SIHP can not only learn the patterns of influence from historical interaction events on later ones, but can also understand the network dynamics by fully considering spatial structure knowledge. Specifically, we incorporate prior knowledge of spatial structure as a graph and design graph regularization in the SIHP, where model parameters are estimated by designing an alternating direction method of multiplier (ADMM) framework. Numerical experiments and a real case study on New York yellow taxi data validate the effectiveness of the proposed method.

ARTICLE HISTORY

Received May 2023
Revised December 2024

KEYWORDS

Event counts; Influence patterns and triggering motivations; Neighboring information; Spatial structure knowledge; Spatiotemporal dynamic network

1. Introduction

Network data have been accumulated and exploited for decision-making in a wide range of applications, including traffic demand systems (Chu, Lam, and Li 2020), bike sharing systems (Schuijbroek, Hampshire, and van Hoeve 2017), solar radiation systems (Zhang et al. 2021b), social systems (Dong and Wang 2022), and academic systems (Wang et al. 2019). Such network data can naturally be regarded as sequences of interaction events over different locations. Interaction events refer to the events that may trigger or influence a series of subsequent events under certain intrinsic spatial structure because of their geographical and semantic proximities (Chu, Lam, and Li 2020). These interaction events are often collected and recorded for different networks that vary dynamically with space and time, which we refer to as event-based dynamic networks. To fully understand an event-based dynamic network, influence patterns and triggering motivations for the interaction events are crucial to be considered. Such influence patterns and triggering motivations reflect the nature and semantics of human/object behaviors in the network; therefore, modeling an event-based dynamic network properly is critically important for the success of human/object-centric applications (Wang et al. 2017; Yan et al. 2017).

An event-based dynamic network is formed by nodes and edges within a network, and edges between nodes are generally

established by sequential events, which should be regarded as a dynamic process driven by sequential events among the network. In this sense, an event-based dynamic network corresponds to a spatiotemporal count system that is represented by counts of events that occur between two connected nodes in the spatial domain and evolves in time because of the fine granularity on the timestamps of events. The event-based dynamic network varies with space and time. Here, varying with space signifies the events that occur at adjacent edges within the network have spatial correlation, and varying with time signifies the events that occur at adjacent discretized time points have temporal correlation. Given that geographical and semantic proximities driven by intrinsic spatial interaction exist among the network, each event may trigger or influence a series of subsequent events, leading to interactions among events.

As a particular example, Figure 1 shows a toy example of a small origin–destination (OD) demand traffic network at some specific time interval with six locations (i.e., nodes) denoted by letters from *a* to *f*. One location is either an origin or a destination, and the event counts between two nodes represent the demands at some time interval from an origin to a destination. Such OD demand traffic networks widely exist, such as Didi and Uber, which aim to provide drivers and passengers with convenient ride services and improve the efficiency of public transportation by fully understanding the passenger demands

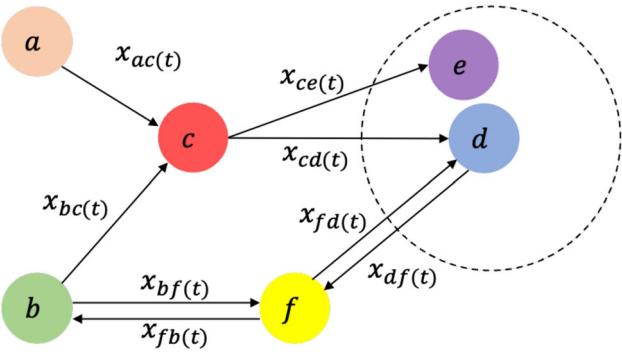


Figure 1. An illustrative example of a traffic demand network at specific time.

in real time (Li, Knoop, and Lint 2021). In this example, eight OD pairs (i.e., $a-c$, $b-c$, $c-d$, $c-e$, $b-f$, $f-b$, $d-f$, $f-d$) have corresponding demand counts $x_{ac}(t), \dots, x_{fd}(t)$ at some time t , where the demand counts of each OD vary with time t . From Figure 1, we have the following insights: *first*, for a specific location, its demands as an origin are probably caused by previous demands from arriving at the location as the destination. For example, a driver who completes a demand from a or b to c may continue another demand from c to d , and thereby a (or b), c and d are interacting due to semantic proximity. *Second*, the dashed circle centered at location d indicates that the locations in the circle are geographical neighbors. Being the geographical neighbor of d , e is more likely to have close densities of passenger demands and shares similar features with d . For example, if locations d and e are adjacent and located in densely populated areas, then demands to both of them (e.g., $c-d$, $c-e$) are likely to be higher. *Third*, unlike conventional spatiotemporal count data, the OD demand is directional. For example, the demands of $b-f$ and $f-b$ can be different (i.e., $x_{bf}(t) \neq x_{fb}(t)$). The directional OD demand system has more complex characteristics than conventional spatiotemporal count systems. This is because that the spatiotemporal count system focuses on the count at each node, whereas the directional OD system focuses on the count at each edge, where an edge refers to the directed connection between two nodes. Obviously, the directional OD system is more complex than the spatiotemporal count system, and thereby the demands of such a system are more challenging to predict. The observations above indicate that an event-based dynamic network contains rich information about the occurring counts of events, the time intervals between related events, as well as the influence patterns or triggering motivations for each interaction event due to semantic or geographical proximities. Therefore, developing methods to model the dynamic count process by fully considering these types of information for interaction events in a dynamic network is necessary.

Although sufficient count observations are provided to show great promise in capturing the characteristics of spatiotemporal event dynamics and interactions among events, modeling event-based dynamic networks still remains a challenging task. *First*, most existing event-based methods mainly focus on modeling temporal dependencies between events and do not consider spatial correlation (Wheatley, Filimonov, and Sornette 2016; Junuthula et al. 2019). These methods ignore the natural characteristics of the events that count data are correlated in

both temporal and spatial domains among the network, and thus they are not suitable for modeling event-based dynamic networks. Although some methods are developed in different application contexts for handling spatiotemporal correlations among count data (Conn et al. 2015; Flaxman et al. 2015), these methods still fail to be applied for modeling event-based dynamic networks because they do not consider the unique directionality between two nodes, thereby leading to an inappropriately captured spatiotemporal correlation structure. *Second*, incorporating the triggering motivations for interaction events into network modeling is crucial for fully understanding the network dynamics. The interaction events can be motivated by various reasons in reality, including periodic contact and reply to a former interaction event. However, previous studies mostly aggregate these dynamic interactions as simple edges over nodes in networks (Newman, Watts, and Strogatz 2002; Dong, Chen, and Wang 2020), which completely ignores the rich information about triggering motivations for each interaction event due to semantic and geographical proximities. In an event-based dynamic network, such as the traffic demand network in Figure 1, considering the interactive influence of related historical events on each current event based on their semantic and geographical proximities is desirable for modeling such networks. *Third*, multi-dimensional Hawkes processes (MHP) have drawn great attention for analyzing the influence patterns between sequential events because of its mutual-exciting properties (Embrechts, Liniger, and Lin 2011; Rasmussen 2013). However, the current methods generally assume no spatial structural knowledge across event dimensions (Zhou, Zha, and Song 2013). Spatial interaction serves as valuable knowledge to understand the intrinsic structure of spatiotemporal events more thoroughly and deserves serious consideration in modeling (Liu, Yan, and Chen 2018). For example, in a traffic network, the distances between ODs and the attractiveness of locations reflect the spatial structure among the network and indeed influence traffic demands (Xian et al. 2021). The ODs with short distance and the locations with more attractiveness usually have more demands that are demonstrated to play important roles in OD-trip generation and analysis.

To address the above issues, we develop a spatiotemporal interactive Hawkes process (SIHP) for the modeling of event-based dynamic networks, which aims to learn the influence patterns from historical interaction events on later ones and understand the dynamics on networks by incorporating spatial structure knowledge effectively. The developed SIHP describes how a series of events occurs and models the rate of interaction events between any pair of nodes on the network explicitly with the information from related historical events and related neighboring nodes, including semantic and geographical neighbors. To incorporate prior knowledge of spatial structure, we represent the spatial structure information as a graph and design graph regularization in the SIHP. Then, the influence patterns among event nodes can be learned by maximizing the likelihood of the SIHP with the constraints imposed by the graph. Specifically, inspired by multi-view learning, the graph and the influence pattern can be represented as matrices and regarded as two views. We align them by exploiting their subspace embeddings, that is, imposing the spatial graph constraint in learning influence pattern by obtaining a shared subspace for both matrices

through the nonnegative matrix factorization (NMF) method. We develop a parameter estimation algorithm based on an alternating direction method of multiplier (ADMM) framework. The estimation results are expected to enhance the prediction accuracy of the online count prediction for future epochs by fully exploring the complicated spatiotemporal influence patterns of the interactive events. Numerical experiments and a case study on real-world New York yellow taxi data demonstrate that the proposed method can model the event-based dynamic network well.

The remainder of this article is organized as follows. Section 2 provides a literature review on event-based network modeling. Section 3 introduces the proposed method, including the formulation of the proposed network, parameter estimation procedure, and online count prediction. Sections 4 and 5 present numerical experiments and a case study of a New York yellow taxi dataset, respectively, to thoroughly evaluate and compare the proposed method with existing benchmarks. Section 6 provides a conclusion and outlines the future directions.

2. Related Work

The goal of this study is to model the influence and evolution patterns of a dynamic network system with interactive events. In this section, we review relevant studies about spatiotemporal network modeling and point processes.

2.1. Spatiotemporal Network Modeling

Extensive research has been performed to model spatiotemporal networks for a wide range of applications, in which various methods have been employed to solve corresponding unique problems. For example, real-time road traffic prediction (Kamarianakis, Shen, and Wynter 2012), individual and population mobility prediction on diverse spatial scales (Yan et al. 2017), and solar ramping event prediction (Zhang et al. 2021b). The aforementioned studies are mainly application-driven without considering the universal characteristics of interaction events thoroughly when employing spatiotemporal modeling. However, our objective in this article is to solve a more general problem, which is to model the interactions and influence patterns of spatiotemporal events in dynamic networks.

As for event-based spatiotemporal characteristics, some traffic prediction methods work with traffic flow data at specific locations (Ashok and Ben-Akiva 2002; Moreira-Matias et al. 2013). These methods capture the temporal evolution at specific locations independently, but ignore natural characteristics of demands that count data are correlated in the spatial domain. Achcar et al. (2011) applied a Poisson spatiotemporal regression model for the Brazilian Amazon forest based on Malaria count data. Dong, Chen, and Wang (2020) proposed a multilayer weighted stochastic block model to characterize the correlated multilayer interactions of individuals on networks. Although these methods can characterize spatiotemporal correlations based on count data, they are still inapplicable for modeling the unique directionality of different node pairs. Bera and Rao (2011) and Perrakis et al. (2012) considered the

directionality of different node pairs for traffic demands. However, they modeled the traffic demand count data separately for different OD pairs without considering spatial correlations. Instead, Buddhavarapu, Bansal, and Prozzi (2021) proposed a spatial count data model that integrated a negative binomial process and dynamic linear models with time-varying parameters to incorporate spatiotemporal correlations of OD traffic demands. Xian et al. (2021) investigated a multivariate Poisson log-normal model with a block-diagonal covariance matrix to capture spatiotemporal correlations for the prediction of OD traffic demands. However, Buddhavarapu, Bansal, and Prozzi (2021) and Xian et al. (2021) focused on the spatiotemporal modeling of various routes (or OD pairs), thereby ignoring the triggering motivations or traffic flows from historical events and events from nearby locations. In addition, when the locations in the network become large, the considered routes will increase exponentially, hence, leading to huge computational burdens.

Deep learning models have been investigated to event-based network prediction recently. For example, Zhang et al. (2021a) established a dynamic node-edge attention network for demand prediction to capture the temporal evolution of OD node topologies. Li, Knoop, and Lint (2021) forecasted traffic states by capturing real-time spatial dependencies via a dynamic graph convolution model. Liu et al. (2019) and Hu et al. (2022) introduced an attention-based long short-term memory (LSTM) to predict traffic flows. Most of existing deep learning models characterize spatiotemporal correlations of OD pairs. Few studies consider triggering motivations among interactive nodes and directionality between two nodes.

2.2. Point Processes

A point process can be represented as a stochastic process that describes the rate of events occurring along the temporal or spatial domain. The rate of events in such point processes is characterized by the conditional intensity function (CIF). The point process has been widely applied to model event sequences, such as social events (Dong and Wang 2022), communication activities (Lerner et al. 2013), patient flows (Xu et al. 2017), and device failures (Xu et al. 2017) because of the prevalence of event data. The point process can be broadly categorized as various types according to the form of CIF, including homogeneous Poisson process, nonhomogeneous Poisson process, and the Hawkes process (Daley and Vere-Jones 2003), among which the Hawkes process is the most popular. Hawkes process is a self-exciting point process that characterizes the positive exciting influences of historical events on later events (Rasmussen 2013) and fits well in a large number of practical scenarios in a wide range of fields from traffic, financial to social networks. Existing Hawkes process models are continuous-time network models as they provide probabilities of observing events between nodes at arbitrary times, which can be categorized as relational event models (Butts et al. 2023).

Numerous variants of the Hawkes process have been used in recent years (Marsan and Lengline 2008; Liu et al. 2016; Xu et al. 2017). For multivariate Hawkes, Farajtabar et al. (2015), Hall and Willett (2016), and Junuthula et al. (2019) mainly captured

the temporal evaluation without fully considering the spatial correlation among nodes of the network. The spatiotemporal Hawkes process has drawn great attention for analyzing the influence patterns between sequential events generated from different locations because of its mutual-exciting properties and aims to learn an influence matrix that captures the mutual triggering weight across spatial and temporal domains (i.e., event sources of different locations and a time decay function) (Ilhan and Kozat 2020). For example, Du et al. (2021) exploited a graph neural Hawkes process to predict event propagation with spatiotemporal characteristics. DuBois, Butts, and Smyth (2013) decomposed the network into a number of blocks, and described relational event dynamics by modeling the interactions between two nodes within the same block. This method can only model the relational event dynamics at each node, failing to model node pairs. To fill the research gap, Soliman et al. (2022) assumed each node pair ij belonged to a block pair (a, b) . The node pair ij is assumed to be excited by node pairs that belongs to the block pair (a, b) . However, this method does not explicitly capture the interaction patterns by considering geographical and semantic neighboring nodes, and the block modeling assumption cannot be satisfied in practice. In addition, when learning the influence matrix, the above methods do not usually consider the spatial information, that is, they generally assume no structural knowledge across the spatial domain, and the influence matrix is estimated such that the arrival time of events is best fitted at each location. Zhou, Zha, and Song (2013) provided a pioneer work to incorporate prior knowledge of social network structure as constraints to characterize the influence patterns between users. Liu, Yan, and Chen (2018) exploited a graph regularized MHP for modeling events with spatiotemporal characteristics by designing the spatial information as graph matrix and set it as a constraint when solving the Hawkes process. This method models the influence patterns by considering spatial and temporal dependencies at each node but cannot characterize the influence patterns for the edges or node pairs that commonly exist in practice, such as traffic OD demand networks.

To fill in the research gap, the contributions of the proposed SIHP method are detailed as follows: *First*, considering temporal evolution from previous events and influence patterns from related geographical and semantic neighboring nodes among the spatial domain, the proposed method can model the rate of interaction events explicitly by fully capturing spatiotemporal correlation structure and noticing the unique directionality between two nodes. *Second*, compared with previous studies that model event dynamics at each node but ignore triggering motivations among node pairs due to semantic and geographical proximities, the SIHP contains rich information about influence patterns of interaction events at each node pair, and thus the event dynamics on the network can be explained well by the rates of interaction events over all node pairs on the network. *Third*, we design a graph-regularized Hawkes process to incorporate prior knowledge of spatial structure as a graph and learn influence patterns between event nodes by maximizing the likelihood of the Hawkes process with constraints imposed by the graph. Therefore, valuable knowledge of spatial interaction is served to better understand the intrinsic structure of spatiotemporal events.

3. Methodology

We consider an event-based dynamic network that has n nodes in the spatial domain, and focus on time-varying event counts at each node. In a network, we represent the indices of spatiotemporal event counts as $\{i, j, t\}$ with $i, j = 1, \dots, n$, where each $\{i, j, t\}$ denotes events that occur at the node pair ij and time t during a period $[0, T]$. We aim to model the dynamics of interaction events for any node pairs in the network by expressing the rate of events explicitly through a network point process. The network point process is a stochastic process that can be represented by a number of counting processes $N(t) = \{N_{ij}(t), i, j = 1, \dots, n\}$ at all node pairs $\{i, j, \forall i, j = 1, \dots, n\}$ in the network, where $N_{ij}(t)$ denotes the number of events at node pair ij occurring in the time interval $[0, t]$. For example, the OD traffic demand network is composed of processes from each node i (origin) to another node j (destination). For each process $N_{ij}(t)$, with $i, j = 1, \dots, n$, we characterize the immediate occurrence rate at time t as $\lambda_{ij}(t)$ by a CIF given the set of historical events $\mathcal{H}(t)$ in the network happening before t , as follows:

$$\lambda_{ij}(t) = \lim_{\Delta t \rightarrow 0} \frac{E(N_{ij}(t + \Delta t) - N_{ij}(t) | \mathcal{H}(t))}{\Delta t}. \quad (1)$$

The CIF depicts the expected rate of events at time t for node pair ij , with $i, j = 1, \dots, n$, conditioned on the past events. A network point process can be uniquely determined by specific CIFs of all node pairs under rather mild conditions.

3.1. Spatiotemporal Interactive Hawkes Process

In our proposed model, we model the CIF of the network point process $\lambda_{ij}(t)$ from one node i to another node j on the network by borrowing the framework of the Hawkes process, which is a point process with a self-exciting property by triggering the occurrences of upcoming events from previous events (Daley and Vere-Jones 2003; Rasmussen 2013), as follows:

$$\lambda_{ij}(t) = \theta_{ij}^{(0)}(t) + \mu_{ij}(t). \quad (2)$$

Here, $\theta_{ij}^{(0)}(t)$ is the time-varying background rate base term and $\mu_{ij}(t)$ captures the triggering effects of historical network events on the occurrences of upcoming events at node pair ij . Considering that there are temporal patterns among node pairs in the network and upcoming events are triggered by historical network events, we admit the time-varying background rate $\theta_{ij}^{(0)}(t)$ and the triggering effect term $\mu_{ij}(t)$ for each node pair ij to capture the temporal evolution. The self-exciting property is reflected by $\mu_{ij}(t)$, demonstrating that the previous events before time t can trigger the occurrences of related events at time t . Specifically, we consider the triggering effects of previous network events from spatial correlation and temporal evolution interactively, which we will introduce in detail in next subsection.

3.1.1. Interactive Influence Patterns

For the modeling of $\mu_{ij}(t)$ at time t during a period $[0, T]$, where a period contains a number of time points, we consider the influence patterns from the neighbors of node pair ij in the

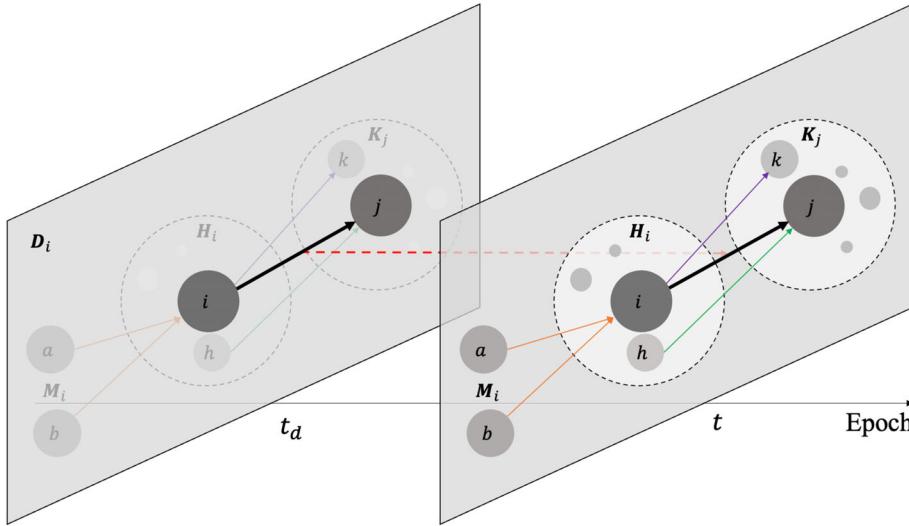


Figure 2. Illustration of the interactions of events.

current period and the node pair ij itself in previous periods. Specifically, we propose two kinds of neighborhood functions, namely, geographical and semantic neighbors, which are used to measure the intrinsic closeness between one node and its neighbors as well as capture the semantic strength related to a node pair, respectively. Therefore, $\mu_{ij}(t)$ is modeled by three decomposed parts, as follows:

$$\mu_{ij}(t) = g_{ij}(t) + s_{ij}(t) + d_{ij}(t) \quad (3)$$

where $g_{ij}(t)$ and $s_{ij}(t)$ represent the influences from historical events at all time points before t in the current period that have similar patterns to node pair ij by considering its geographical neighbors and semantic neighbors, respectively, and $d_{ij}(t)$ represents the influence from historical events of the node pair ij itself at all time points before t in previous periods. We introduce $g_{ij}(t)$, $s_{ij}(t)$ and $d_{ij}(t)$ in detail as follows.

The geographical neighbors of one node are those nodes that are geographically adjacent to it. One node and its geographical neighbors are more likely to send or receive similar densities of event counts. As shown in Figure 2, taking node i as an example, the nodes within its surrounding dashed circle are its geographical neighbors (e.g., h). Being the geographical neighbors of i , h is more likely to generate similar numbers of events and share similar features with i . For example, when i and h are adjacent and located in a sparsely populated suburb, both of them are likely to have fewer event counts. In this figure, hj and ik are geographically neighboring node pairs of ij . These geographical neighbors of node pair ij (i.e., have the same source node and geographically neighboring target nodes as the current node pair, and have geographically neighboring source nodes and the same target node as the current node pair) share similar patterns, thus, with which we represent the influence from historical events. Therefore, $g_{ij}(t)$ is modeled as

$$g_{ij}(t) = \sum_{h \in H_i} \beta_{ij,hj}^{(1)} \sum_{t_h^c < t} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c) + \sum_{k \in K_j} \beta_{ij,ik}^{(1)} \sum_{t_k^c < t} e^{-\frac{(t-t_k^c)}{\delta}} n_{ik}(t_k^c) \quad (4)$$

where we illustrate two terms on the right side of the equation as follows:

- 1) For the source node i , we define the set of its geographical neighbors as

$$H_i = \{h | dis(i, h) \leq G_i\}, \quad (5)$$

where $dis(\cdot)$ denotes the distance between two nodes. G_i is a threshold that is the range of neighborhood. It should be noted that H_i includes the node i , because node i can be considered as the geographical neighbor of itself satisfying (5). Therefore, the first term in (4), which is $\sum_{h \in H_i} \beta_{ij,hj}^{(1)} \sum_{t_h^c < t} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c)$, depicts the influence from all the historical events that have geographically neighboring source node h with $h \in H_i$ and the same target node j . Here, $\beta_{ij,hj}^{(1)}$ denotes the jump size to the CIF of ij excited by hj . Consider that the events in recent time points are more likely to have more triggering effects on the upcoming events. Thus, we employ an exponential kernel $e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c)$ to describe the time decay, where t_h^c denotes time prior to t in the current period, $n_{hj}(t_h^c)$ denotes the event counts at node pair hj that occurred at time t_h^c , and δ is the bandwidth with respect to time that is regarded as a hyperparameter. Furthermore, we represent $\beta_{ij,hj}^{(1)}$ as the product of the influence base term $\alpha_{ij}^{(1)}$ from node pair ij , the influence exciting term $\theta_{hj}^{(1)}$ from node pair hj , and a weight w_{ih} measuring the similarity between the source node i and its geographical neighbor h

$$\beta_{ij,hj}^{(1)} = \alpha_{ij}^{(1)} \cdot \theta_{hj}^{(1)} \cdot w_{ih} \quad (6)$$

Intuitively, the affinity between the neighboring node h and the target node j should be interacted by the source node i because i and h are geographical neighbors. Therefore, incorporating such characteristics of the source nodes in modeling jump size $\beta_{ij,hj}^{(1)}$ is desirable, and thus we design the weight w_{ih} to measure the similarity between the source node i and its geographical neighbor h . Inspired by recent attention-based

models (Bahdanau, Cho, and Bengio 2014), we define w_{ih} using a SoftMax function based on their distance

$$w_{ih} = \frac{e^{-\text{dis}(i,h)}}{\sum_{h \in H_i} e^{-\text{dis}(i,h)}}. \quad (7)$$

The closer the nodes i and h , the larger weight they have, thereby indicating a high similarity of the two nodes.

- 2) For the target node j , its geographical neighbors are defined following the same procedure as that of the source node i , as

$$K_j = \{k | \text{dis}(k,j) \leq G_j\}. \quad (8)$$

Notably, we consider the node j as a geographical neighbor of itself, that is, K_j includes the node j . The second term in (4),

that is, $\sum_{k \in K_j} \beta_{ij,ik}^{(1)} \sum_{t_k^c < t} e^{-\frac{(t-t_k^c)}{\delta}} n_{ik}(t_k^c)$, depicts the influence from all the historical events that have the same source node i and geographically neighboring target node k with $k \in K_j$, where $\beta_{ij,ik}^{(1)}$ denotes the jump size to the CIF of ij excited by ik , and t_k^c denotes time prior to t in the current period, and $n_{ik}(t_k^c)$ denotes the event counts at node pair ik that occurred at time t_k^c . Similarly, we represent $\beta_{ij,ik}^{(1)}$ as

$$\beta_{ij,ik}^{(1)} = \alpha_{ij}^{(1)} \cdot \theta_{ik}^{(1)} \cdot w_{kj}, \quad (9)$$

which is the product of $\alpha_{ij}^{(1)}$, the influence exciting term $\theta_{ik}^{(1)}$ from node pair ik , and the weight w_{kj} , that is, $w_{kj} = \frac{e^{-\text{dis}(kj)}}{\sum_{k \in K_j} e^{-\text{dis}(kj)}}$, measuring the similarity between the target node j and its geographical neighbor k .

Semantic neighbors are defined as follows—if there exist a large number of events between two nodes at more than one time slots during a period, the two nodes are semantic neighbors of each other. Exploring semantic neighbors can better discover more complex temporal interaction patterns among different nodes. Only if node i has been a target, it can then be a source to the next target j . Thus, we consider the nodes that acts as the source nodes of the target node i as the semantic neighbors of node i . As shown in Figure 2, for example, the past events at ai and bi in which i was a target node have influence on the current events at ij . We define the set of semantic neighbors of node i as the source nodes for target node i :

$$M_i = \{\forall m \text{ that acts as source node for all node pairs } mi\}. \quad (10)$$

Then, the influence of the semantic neighbors from all the historical event counts is represented as

$$s_{ij}(t) = \sum_{m \in M_i} \beta_{ij,mi}^{(1)} \sum_{t_m^c < t} e^{-\frac{(t-t_m^c)}{\delta}} n_{mi}(t_m^c), \quad (11)$$

where $\beta_{ij,mi}^{(1)}$ denotes the jump size to the CIF of ij excited by mi , t_m^c denotes time prior to t in the current period, and $n_{mi}(t_m^c)$ denotes the event counts at node pair mi that occurred at time t_m^c . Specifically, we represent $\beta_{ij,mi}^{(1)}$ as

$$\beta_{ij,mi}^{(1)} = \frac{1}{|M_i|} \alpha_{ij}^{(1)} \cdot \theta_{mi}^{(1)},$$

where $|M_i|$ denotes the number of elements in M_i . To keep consistence with w_{ih} and w_{kj} in (7) and (9) that can normalize the influence of geographical neighbors, $1/|M_i|$ is designed here to normalize the influence of semantic neighbors.

In addition, we find the event counts in a period have similar patterns to the ones in previous periods. For example, the traffic demands within a day have similar daily or weekly patterns to the ones on previous days. Therefore, we consider the influence from all the historical event counts of the node pair ij itself in previous periods, which is represented as

$$d_{ij}(t) = \frac{1}{|D_i|} \sum_{d \in D_i} \alpha_{ij}^{(2)} \sum_{t_d^p < t} e^{-\frac{(t-t_d^p)}{\delta}} n_{ij}^{(d)}(t_d^p), \quad (12)$$

where D_i denotes the set of previous periods, $\alpha_{ij}^{(2)}$ denotes the influence base term from node pair ij in previous periods, t_d^p denotes time prior to t in the previous period d , and $n_{ij}^{(d)}(t_d^p)$ denotes the number of event counts at node pair ij occurred at time t_d^p in the previous period d . $|D_i|$ is the number of elements in D_i , and $1/|D_i|$ is designed here for normalization.

From the above detailed introduction of the proposed SIHP, its CIF $\lambda_{ij}(t)$ from one node i to another node j on the network, with $i, j = 1, \dots, n$, is summarized as

$$\begin{aligned} \lambda_{ij}(t) = & \theta_{ij}^{(0)}(t) + \alpha_{ij}^{(1)} \left(\sum_{h \in H_i} \theta_{hj}^{(1)} \sum_{t_h^c < t} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c) w_{ih} \right. \\ & + \sum_{k \in K_j} \theta_{ik}^{(1)} \sum_{t_k^c < t} e^{-\frac{(t-t_k^c)}{\delta}} n_{ik}(t_k^c) w_{kj} \\ & + \frac{1}{|M_i|} \sum_{m \in M_i} \theta_{mi}^{(1)} \sum_{t_m^c < t} e^{-\frac{(t-t_m^c)}{\delta}} n_{mi}(t_m^c) \\ & \left. + \frac{1}{|D_i|} \sum_{d \in D_i} \alpha_{ij}^{(2)} \sum_{t_d^p < t} e^{-\frac{(t-t_d^p)}{\delta}} n_{ij}^{(d)}(t_d^p) \right). \end{aligned} \quad (13)$$

3.1.2. Consideration of Spatial Structural Prior Knowledge

Spatial structural knowledge of the network, such as location information and the relationship/similarity of a node pair, pervasively exists in practice. As a network is composed of a number of nodes, such spatial structure knowledge can naturally be represented as a graph. We introduce a spatial information matrix $\mathbf{B} = \{B_{ij}\} \in \mathbb{R}^{n \times n}$ to represent the intrinsic spatial structural knowledge of the network, which can be designated for specific considerations. For instance, if the network connection information is considered, then \mathbf{B} can be designed as 0–1 weighting matrix, where $B_{ij} = 1$ if and only if the nodes i and j are connected by an edge, and $B_{ij} = 0$, if otherwise. If needed, B_{ij} can be designed as weight values based on prior knowledge about the relationship between nodes (e.g., distance, similarity). Moreover, the structural relationship in complex systems can be represented by the connection of the node pair ij and their related functions (Cai et al. 2011). As a particular example, the attractiveness of node pairs provides useful information for traffic OD demand network, that is, more attractiveness of a node pair will bring more traffic demands. Inspired by Yan

et al. (2017), we define the attractiveness B_{ij} from i to j in the traffic demand network as $B_{ij} = D_{ij}/\text{dis}(i,j)$, where D_{ij} is the log transformation of monthly average number of demands. Its attractiveness is proportional to its demand, but inversely proportional to its distance.

3.2. Parameter Estimation with Graph Regularization

As formulated in (13) of the SIHP, we write the model parameters to be estimated in a matrix form, including the background rate base term $\Theta_0 = \{\theta_{ij}^{(0)}(t)\}$, the influence exciting term $\Theta_1 = \{\theta_{ij}^{(1)}\}$, the influence base term $\alpha_1 = \{\alpha_{ij}^{(1)}\}$ in the current period, and the influence base term in previous periods $\alpha_2 = \{\alpha_{ij}^{(2)}\}$, with $i, j = 1, \dots, n$. Overall, we have $\Theta_0, \Theta_1, \alpha_1, \alpha_2 \in \mathbb{R}^{n \times n}$ to be estimated. Given L available samples with count event data of the network at time points t_1, \dots, t_T during a time period $[0, T]$, the log-likelihood function of the SIHP can be expressed in the following form

$$\mathcal{L}(\Theta_0, \Theta_1, \alpha_1, \alpha_2) = \sum_{l=1}^L \sum_{i=1}^n \sum_{j=1}^n \left[\sum_{t=t_1}^{t_T} n_{ij,l}(t) \log \lambda_{ij,l}(t) - \int_0^T \lambda_{ij,l}(t) dt \right], \quad (14)$$

where the integral term can be computed analytically. The detailed proof of (14) can be found in Appendix A of the supplementary materials.

When estimating model parameters, we aim to integrate spatial structural knowledge with the log-likelihood function to obtain more accurate results. Among the model parameters, the influence base term α_1 captures the correlation of each node pair among the network, and thus is more related to the spatial structure knowledge of the network than other parameters. Therefore, we impose the spatial information matrix \mathbf{B} as graph regularization in learning α_1 . Given that \mathbf{B} and α_1 may be measured in different scales and domains with various underlying physical meanings, \mathbf{B} and α_1 cannot be directly comparable if their differences are minimized. Instead, we exploit their subspace embeddings by a multi-view subspace learning framework (Xu, Tao, and Xu 2013), which regards \mathbf{B} and α_1 as different information sources. We attempt to obtain the best alignment between them. In particular, we adopt the NMF method to transfer \mathbf{B} and α_1 to a shared subspace \mathbf{Q} (Zhu et al. 2019) and represent the embeddings of α_1 and \mathbf{B} as \mathbf{U}_1 and \mathbf{U}_2 , respectively. The goal of the NMF method is to minimize the difference between \mathbf{U}_1 and \mathbf{U}_2 . To summarize, we propose the objective function of SIHP with graph regularization from spatial structural knowledge as follows

$$\begin{aligned} & \min -\mathcal{L}(\Theta_0, \Theta_1, \alpha_1, \alpha_2) + \lambda_1 (\|\alpha_1 - \mathbf{Q}\mathbf{U}_1\|^2 + \|\mathbf{B} - \mathbf{Q}\mathbf{U}_2\|^2) \\ & + \lambda_2 \|\mathbf{U}_1 - \mathbf{U}_2\|^2, \\ & \text{s.t. } \mathbf{Q}, \mathbf{U}_1, \mathbf{U}_2, \lambda_1, \lambda_2 \geq 0. \end{aligned} \quad (15)$$

In (15), the first term is to minimize the negative log-likelihood of SIHP, and the remaining terms are to embed \mathbf{B}

and α_1 to a shared subspace \mathbf{Q} accurately and obtain the best alignment of their views \mathbf{U}_1 and \mathbf{U}_2 based on the measure of l_2 -norm. λ_1 and λ_2 are tuning coefficients to control the strength of the regularization terms.

Solving the objective function (15) directly to estimate Θ_0 , Θ_1 , α_1 , α_2 , \mathbf{Q} , \mathbf{U}_1 , and \mathbf{U}_2 is difficult. Instead, we propose to use ADMM (Wei and Ozdaglar 2012) to solve the objective function by considering the augmented Lagrangian function given by

$$\begin{aligned} \mathbf{L}_p = & -\mathcal{L}(\Theta_0, \Theta_1, \alpha_1, \alpha_2) + \lambda_1 (\|\mathbf{Z} - \mathbf{Q}\mathbf{U}_1\|^2 + \|\mathbf{B} - \mathbf{Q}\mathbf{U}_2\|^2) \\ & + \lambda_2 \|\mathbf{U}_1 - \mathbf{U}_2\|^2 + \rho \text{tr}(\mathbf{W}^T(\alpha_1 - \mathbf{Z})) + \frac{\rho}{2} \|\alpha_1 - \mathbf{Z}\|^2, \end{aligned} \quad (16)$$

where we induce an auxiliary variable \mathbf{Z} to rewrite the objective function, ρ is the penalty coefficient, $\text{tr}(\cdot)$ is the trace of matrix, and \mathbf{W} is the dual variable associated with the constraint $\alpha_1 = \mathbf{Z}$. Then, we develop an iterative algorithm, as presented in Algorithm 1, to iteratively update Θ_0 , Θ_1 , α_1 , α_2 , \mathbf{Q} , \mathbf{U}_1 , \mathbf{U}_2 , \mathbf{W} , and \mathbf{Z} by the following four steps:

- Step 1: Updating $\Theta_0, \Theta_1, \alpha_2$

Given the estimated parameters $\alpha_1, \mathbf{Q}, \mathbf{U}_1, \mathbf{U}_2, \mathbf{W}, \mathbf{Z}$ at iteration p , we update $\Theta_0, \Theta_1, \alpha_2$ for iteration $p+1$ through the following settings: $\Theta_0^{p+1}, \Theta_1^{p+1}, \alpha_2^{p+1} = \arg\min \mathbf{L}_p(\Theta_0, \Theta_1, \alpha_2, \alpha_1^p, \mathbf{Q}^p, \mathbf{U}_1^p, \mathbf{U}_2^p, \mathbf{W}^p, \mathbf{Z}^p)$, which is equivalent to minimize

$$-\mathcal{L}(\Theta_0, \Theta_1, \alpha_2, \alpha_1^p). \quad (17)$$

We update Θ_0, Θ_1 , and α_2 given α_1^p by adopting the expectation–maximization (EM) algorithm, where the expectation (E) step and the maximization (M) step are implemented iteratively until convergence is reached.

In E step, for sample l , with $l = 1, \dots, L$, we define $p_{ij,l}^0(t)$ as the probability that an event occurring at time t and node pair ij is a background event, and $p_{ij,l}^{g1}(t, t_h^c), p_{ij,l}^{g2}(t, t_k^c), p_{ij,l}^s(t, t_m^c)$ and $p_{ij,l}^d(t, t_d^p)$ as the probabilities of an event occurring at time t caused by the events at previous time t_h^c, t_k^c, t_m^c and t_d^p with respect to the geographical neighbors for source nodes and target nodes, semantic neighbors, and previous periods except the influence of α_1 , respectively. We approximate $p_{ij,l}^0(t), p_{ij,l}^{g1}(t, t_h^c), p_{ij,l}^{g2}(t, t_k^c), p_{ij,l}^s(t, t_m^c)$, and $p_{ij,l}^d(t, t_d^p)$ as

$$p_{ij,l}^0(t) = \frac{\theta_{ij}^{(0)}(t)}{\lambda_{ij,l}(t)},$$

$$p_{ij,l}^{g1}(t, t_h^c) = \frac{\sum_{h \in H_i} \theta_{hj}^{(1)} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj,l}(t_h^c) w_{ih}}{\lambda_{ij,l}(t)},$$

$$p_{ij,l}^{g2}(t, t_k^c) = \frac{\sum_{k \in K_j} \theta_{ik}^{(1)} e^{-\frac{(t-t_k^c)}{\delta}} n_{ik,l}(t_k^c) w_{kj}}{\lambda_{ij,l}(t)},$$

$$p_{ij,l}^s(t, t_m^c) = \frac{\sum_{m \in M_i} \theta_{mi}^{(1)} e^{-\frac{(t-t_m^c)}{\delta}} n_{mi}(t_m^c)}{|M_i| \lambda_{ij,l}(t)},$$

Algorithm 1 Parameter estimation algorithm

Input: Historical event counts of the dynamic networks

Output: Estimated parameters $\Theta_0, \Theta_1, \alpha_1, \alpha_2$

1: Procedure:

2: Initialize model parameters as $\Theta_0^0, \Theta_1^0, \alpha_1^0, \alpha_2^0, Z^0, Q^0, U_1^0, U_2^0, W^0$ and set the iteration $p = 0$.

3: **While**

4: Step 1: update to obtain $\Theta_0^{p+1}, \Theta_1^{p+1}$, and α_2^{p+1} by (17)–(21) through integrating the EM algorithm and least squares method.

Step 2: update to obtain α_1^{p+1} by (22), (23).

5: Step 3: update to obtain Z^{p+1} by (24), (25).

6: Step 4: update to obtain W^{p+1} by (26).

7: Step 5: update to obtain Q^{p+1}, U_1^{p+1} , and U_2^{p+1} by (27), (28) through inducing Lagrange multipliers.

8: **If** convergence is achieved

9: **Stop** the loop

10: **Else**

11: $p = p + 1$

$$p_{ij,l}^d(t, t_d^p) = \frac{\sum_{d \in D_i} \alpha_{ij}^{(2)} e^{-\frac{(t-t_d^p)}{\delta}} n_{ij,l}^{(d)}(t_d^p)}{|\mathcal{D}_i| \lambda_{ij,l}(t)}. \quad (18)$$

Then, we estimate the number of counts with respect to background events and events at the previous time with respect to geographical neighbors, semantic neighbors, and previous periods except the influence of α_1 by $\hat{n}_{ij,l}^0(t) = p_{ij,l}^0(t) n_{ij,l}(t)$, $\hat{n}_{ij,l}^{g1}(t, t_h^c) = p_{ij,l}^{g1}(t, t_h^c) n_{ij,l}(t)$, $\hat{n}_{ij,l}^{g2}(t, t_k^c) = p_{ij,l}^{g2}(t, t_k^c) n_{ij,l}(t)$, $\hat{n}_{ij,l}^s(t, t_m^c) = p_{ij,l}^s(t, t_m^c) n_{ij,l}(t)$ and $\hat{n}_{ij,l}^d(t, t_d^p) = p_{ij,l}^d(t, t_d^p) n_{ij,l}(t)$, with $i, j = 1, \dots, n, l = 1, \dots, L$.

In M step, our objective is to update Θ_0, Θ_1 , and α_2 .

Proposition 1. Given $\hat{n}_{ij,l}^0(t), \hat{n}_{ij,l}^{g1}(t, t_h^c), \hat{n}_{ij,l}^{g2}(t, t_k^c), \hat{n}_{ij,l}^s(t, t_m^c)$, and $\hat{n}_{ij,l}^d(t, t_d^p), \theta_{ij}^{(0)}, \theta_{ij}^{(1)}$, and $\alpha_{ij}^{(2)}$ can be estimated by solving problem (17), and the solutions are given by

$$\theta_{ij}^{(0)}(t) = \frac{\sum_{l=1}^L \hat{n}_{ij,l}^0(t)}{L} \quad (19)$$

$$\begin{aligned} \theta_{ij}^{(1)} = & \left(\sum_{l=1}^L \sum_{h \in H_{ij}} \sum_{t=t_1}^{t_T} \sum_{t_h^c=t_1}^t \hat{n}_{ih,l}^{g1}(t, t_h^c) \right. \\ & + \sum_{l=1}^L \sum_{k \in K_{ij}} \sum_{t=t_1}^{t_T} \sum_{t_k^c=t_1}^t \hat{n}_{ik,l}^{g2}(t, t_k^c) \\ & + \sum_{l=1}^L \sum_{m \in M_{ij}} \sum_{t=t_1}^{t_T} \sum_{t_m^c=t_1}^t \hat{n}_{jm,l}^s(t, t_m^c) \Bigg) / \\ & \left(\sum_{l=1}^L \sum_{h \in H_{ij}} \sum_{t=t_1}^{t_T} \delta \left(n_{ij,l}(t_1) - n_{ij,l}(t) e^{-\frac{t_T-t}{\delta}} \right) w_{ih} \right. \\ & + \sum_{l=1}^L \sum_{k \in K_{ij}} \sum_{t=t_1}^{t_T} \delta \left(n_{ij,l}(t_1) - n_{ij,l}(t) e^{-\frac{t_T-t}{\delta}} \right) w_{kj} \\ & \left. + \sum_{l=1}^L \sum_{m \in M_{ij}} \sum_{t=t_1}^{t_T} \frac{\delta}{|\mathcal{M}_i|} \left(n_{ij,l}(t_1) - n_{ij,l}(t) e^{-\frac{t_T-t}{\delta}} \right) \right) \end{aligned} \quad (20)$$

and

$$\alpha_{ij}^{(2)} = \frac{|\mathcal{D}_i| \sum_{l=1}^L \sum_{d \in D_{ij}} \sum_{t=t_1}^{t_T} \sum_{t_d^p=t_1}^t \hat{n}_{ij,l}^d(t, t_d^p)}{\sum_{l=1}^L \sum_{d \in D_{ij}} \sum_{t=t_1}^{t_T} \delta \left(n_{ij,l}^{(d)}(t_1) - n_{ij,l}^{(d)}(t) e^{-\frac{t_T-t}{\delta}} \right)}. \quad (21)$$

Notably, $\theta_{ij}^{(1)}$ may occur at the first three terms in the summation part in (13) as the geographical neighbors and semantic neighbors of other node pairs. We consider all of the scenarios where $\theta_{ij}^{(1)}$ has occurred and denote H_{ij} , K_{ij} , and M_{ij} as the set of scenarios with respect to the source nodes, target nodes, and semantic neighbors, respectively. In a special case where $\theta_{ij}^{(0)}(t)$ remains a constant, that is, $\theta_{ij}^{(0)}(t) = \theta_{ij}^{(0)}, \forall t = t_1, \dots, t_T$, we have $\theta_{ij}^{(0)} = \frac{\sum_{l=1}^L \sum_{t=t_1}^{t_T} \hat{n}_{ij,l}^0(t)}{LT}$. The details of the EM algorithm for estimating Θ_0, Θ_1 , and α_2 with the proof of Proposition 1 are provided in Appendix B of the supplementary materials.

• **Step 2: Updating α_1**

To update α_1 , we aim to solve the following problem $\alpha_1^{p+1} = \text{argmin}_{L_p} (\alpha_1, \Theta_0^{p+1}, \Theta_1^{p+1}, \alpha_2^{p+1}, Q^p, U_1^p, U_2^p, W^p, Z^p)$. Given $\Theta_0^{p+1}, \Theta_1^{p+1}$, and α_2^{p+1} , (13) reduces to $y_{ij}(t) = \alpha_{ij}^{(1)} x_{ij}(t) + \varepsilon_{ij}(t)$, where $y_{ij}(t) = \lambda_{ij}(t) - \theta_{ij}^{(0)} - 1/|\mathcal{D}_i| \sum_{d \in D_i} \alpha_{ij}^{(2)} \sum_{t_d^p < t} e^{-\frac{(t-t_d^p)}{\delta}} n_{ij}^{(d)}(t_d^p)$ and $x_{ij}(t) = \sum_{h \in H_i} \theta_{hj}^{(1)} \sum_{t_h^c < t} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c)$. Given that $y_{ij}(t)$ cannot be exactly equal to $\alpha_{ij}^{(1)} x_{ij}(t)$ due to the existence of measurement errors, we add an error term $\varepsilon_{ij}(t)$ to make the equation hold. By providing the sample data $\mathbf{X}_{ij,l} = [x_{ij,l}(t_1), \dots, x_{ij,l}(t_T)]^\top$ and $\mathbf{Y}_{ij,l} = [y_{ij,l}(t_1), \dots, y_{ij,l}(t_T)]^\top$ with $l = 1, \dots, L$, our objective is equivalent to minimize

$$\sum_{l=1}^L \|\mathbf{Y}_{ij,l} - \alpha_{ij}^{(1)} \mathbf{X}_{ij,l}\|^2 + \frac{\rho}{2} (\alpha_{ij}^{(1)} - z_{ij}^p + w_{ij}^p)^2. \quad (22)$$

Therefore, $\alpha_{ij}^{(1)}$ is updated as

$$\alpha_{ij}^{(1)} = \frac{2 \sum_{l=1}^L \mathbf{X}_{ij,l}^T \mathbf{Y}_{ij,l} + \rho (z_{ij}^p - w_{ij}^p)}{2 \sum_{l=1}^L \mathbf{X}_{ij,l}^T \mathbf{X}_{ij,l} + \rho}. \quad (23)$$

The detailed estimation procedure of $\alpha_{ij}^{(1)}$ can be found in Appendix C of the supplementary materials.

- **Step 3: Updating Z**

To update \mathbf{Z} for iteration $p+1$, we design to solve the following problem $\mathbf{Z}^{p+1} = \operatorname{argmin} \mathbf{L}_p (\mathbf{Z}, \Theta_0^{p+1}, \Theta_1^{p+1}, \boldsymbol{\alpha}_1^{p+1}, \boldsymbol{\alpha}_2^{p+1}, \mathbf{Q}^p, \mathbf{U}_1^p, \mathbf{U}_2^p, \mathbf{W}^p)$. The relevant terms from \mathbf{L}_p is to minimize

$$\lambda_1 \|\mathbf{Z} - \mathbf{Q}^p \mathbf{U}_1^p\|^2 + \frac{\rho}{2} \|\boldsymbol{\alpha}_1^{p+1} - \mathbf{Z} + \mathbf{W}^p\|^2. \quad (24)$$

By setting the first derivative of (24) as zero, \mathbf{Z} is updated as

$$\mathbf{Z}^{p+1} = \frac{2\lambda_1 \mathbf{Q}^p \mathbf{U}_1^p + \rho (\boldsymbol{\alpha}_1^{p+1} + \mathbf{W}^p)}{2\lambda_1 + \rho}. \quad (25)$$

- **Step 4: Updating W**

\mathbf{W} for iteration $p+1$ is updated as

$$\mathbf{W}^{p+1} = \mathbf{W}^p - \eta (\boldsymbol{\alpha}_1^{p+1} - \mathbf{Z}^{p+1}), \quad (26)$$

where η is a tuning coefficient that denotes the updating rate of \mathbf{W} .

- **Step 5: Updating Q, U₁, and U₂**

To update \mathbf{Q} , \mathbf{U}_1 , and \mathbf{U}_2 for iteration $p+1$, we aim to solve $\mathbf{Q}^{p+1}, \mathbf{U}_1^{p+1}, \mathbf{U}_2^{p+1} = \operatorname{argmin} \mathbf{L}_p (\mathbf{Q}, \mathbf{U}_1, \mathbf{U}_2, \Theta_0^{p+1}, \Theta_1^{p+1}, \boldsymbol{\alpha}_1^{p+1}, \boldsymbol{\alpha}_2^{p+1}, \mathbf{Z}^{p+1}, \mathbf{W}^{p+1})$. The relevant terms from \mathbf{L}_p is to minimize

$$\begin{aligned} & \lambda_1 (\|\mathbf{Z}^{p+1} - \mathbf{Q} \mathbf{U}_1\|^2 + \|\mathbf{B} - \mathbf{Q} \mathbf{U}_2\|^2) \\ & + \lambda_2 \|\mathbf{U}_1 - \mathbf{U}_2\|^2, \text{s.t., } \mathbf{Q}, \mathbf{U}_1, \mathbf{U}_2 \geq 0. \end{aligned} \quad (27)$$

Here, we implement the Lagrange multiplier method to update \mathbf{Q} , \mathbf{U}_1 , and \mathbf{U}_2 .

Proposition 2. With the formulation of (27), its solution is given by

$$\begin{aligned} \mathbf{Q}^{p+1} &= \mathbf{Q}^p \cdot * \left(\mathbf{Z}^{p+1} \mathbf{U}_1^{p^T} + \mathbf{B} \mathbf{U}_2^{p^T} \right) ./ (\mathbf{Q}^p \mathbf{U}_1^p \mathbf{U}_1^{p^T} + \mathbf{Q}^p \mathbf{U}_2^p \mathbf{U}_2^{p^T}), \\ \mathbf{U}_1^{p+1} &= \mathbf{U}_1^p \cdot * \left(\lambda_1 \mathbf{Q}^{p^T} \mathbf{Z}^{p+1} + \lambda_2 \mathbf{U}_2^p \right) ./ (\lambda_1 \mathbf{Q}^{p^T} \mathbf{Q}^p \mathbf{U}_1^p + \lambda_2 \mathbf{U}_1^p), \\ \mathbf{U}_2^{p+1} &= \mathbf{U}_2^p \cdot * \left(\lambda_1 \mathbf{Q}^{p^T} \mathbf{B} - \lambda_2 \mathbf{U}_1^p \right) ./ (\lambda_1 \mathbf{Q}^{p^T} \mathbf{Q}^p \mathbf{U}_2^p - \lambda_2 \mathbf{U}_2^p), \end{aligned} \quad (28)$$

where $\cdot *$ denotes the Hadamard product. The detailed proof of **Proposition 2** is provided in Appendix D of the supplementary materials. In (28), if the denominator term is equal to zero (i.e., the elements in $\mathbf{Q}^p \mathbf{U}_1^p \mathbf{U}_1^{p^T} + \mathbf{Q}^p \mathbf{U}_2^p \mathbf{U}_2^{p^T}$, $\lambda_1 \mathbf{Q}^{p^T} \mathbf{Q}^p \mathbf{U}_1^p + \lambda_2 \mathbf{U}_1^p$ and $\lambda_1 \mathbf{Q}^{p^T} \mathbf{Q}^p \mathbf{U}_2^p - \lambda_2 \mathbf{U}_2^p$), the corresponding elements in \mathbf{Q} , \mathbf{U}_1 , and \mathbf{U}_2 remain the values at iteration p .

It should be noted that the proposed method has the potential for incorporating low rank and sparsity regularizations for large-scale networks. In practice, it is typically observed that an event generally affects its neighboring events but hardly affects distant events, particularly for large-scale networks (Tobler 1970). The spatial information matrix \mathbf{B} and the influence base term matrix $\boldsymbol{\alpha}_1$ can reflect that nodes tend to connect to only a fraction of all nodes, indicating the sparsity property of both matrices. 0-1 weighting for \mathbf{B} is one of the simplest and most used weighting methods for characterizing the sparsity, where the element $B_{ij} = 1$ if and only if nodes i and j are connected by an edge, and 0 otherwise. In addition, we can integrate the regularization term $\lambda_3 (\|\mathbf{U}_1\|_1 + \|\mathbf{U}_2\|_1)$ into the objective function (15), where the l_1 -norm of the views is employed to ensure the sparsity of the matrices \mathbf{B} and $\boldsymbol{\alpha}_1$. Moreover, events located in different nodes mutually influence each other, leading to the formation of communities. The communities imply a low-rank property of the network structure (Zhou, Zha, and Song 2013). In such scenarios, we can first learn the community structure based on event occurrences of node pairs, and subsequently capture interactions among events within each community. Given these insights, \mathbf{B} , $\boldsymbol{\alpha}_1$ and the community structure are capable of fulfilling the sparsity and low-rank properties when modeling events that exhibit spatio-temporal characteristics within large-scale networks.

For computational efficiency on parameter estimation, recall that in the prior works on spatiotemporal modeling in Buddhavarapu, Bansal, and Prozzi (2021) and Xian et al. (2021), the spatial correlations among OD pairs are captured by corresponding spatial covariance matrix with a scale of $n^2 \times n^2$. We can see from (14) that the overall log-likelihood function of SIHP for each sample l is composed of the summation of $\lambda_{ij,l}(t)$ values from n^2 node pairs. Compared with handling the inverse of $n^2 \times n^2$ covariance matrix, taking the summation of n^2 node pairs for $\lambda_{ij,l}(t)$ greatly reduces computational burdens, especially when n becomes large.

3.3. Online Event Count Prediction at Future Time

After parameter estimation, the function of CIF at any node pair ij , with $i, j = 1, \dots, n$, is learned. At future time t , given the set of historical events $\mathcal{H}(t)$ in the network happening before t , we predict the expected event count at node ij as the CIF $\lambda_{ij}(t)$ by (13), with $i, j = 1, \dots, n$.

4. Numerical Experiments

We simulate an event-based dynamic network to present the results of the numerical experiments. We assume $n = 10$ nodes are in the network and record the counts of events that occur at 15 consecutive time epochs, and thus $T = 15$ epochs. We generate the count data from the Poisson distribution at each time epoch, in which the model parameter is designed as the CIF formulated in (13). The values of the parameters in (13) are set as follows. We first design an arbitrary variable v_i , with $i = 1, \dots, n$, which represents node information, and generate the value of node information from uniform distribution Uniform (0, 1) randomly for each node. We set $\theta_{ij}^{(0)} =$

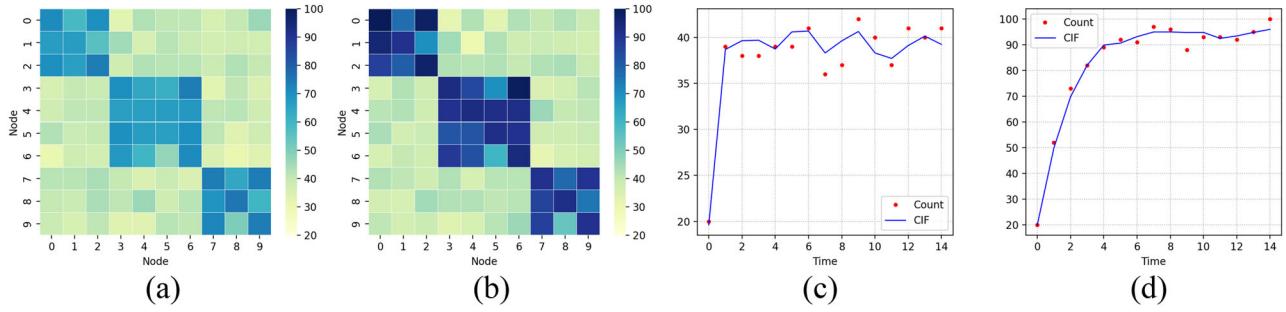


Figure 3. Examples of simulated event-based dynamic network data. (a) Event counts of the network on epoch 3, (b) event counts of the network on epoch 13, (c) event counts and CIF at the node pair (2, 8), and (d) event counts and CIF at the node pair (5, 5).

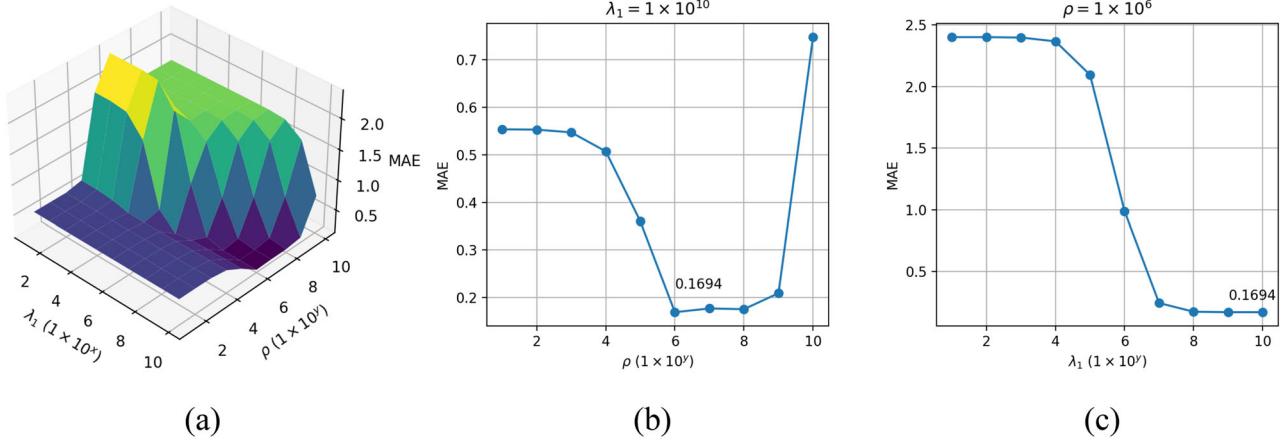


Figure 4. Model performance by setting various values of tuning coefficients.

$20e^{-0.1(v_i-v_j)^2}$, $\theta_{ij}^{(1)} = 2e^{-0.5(v_i-v_j)^2}$, $\alpha_{ij}^{(2)} = 0.5e^{-0.25(v_i-v_j)^2}$, with $i, j = 1, \dots, n$, and $\delta = 3$. Then, we design the spatial information of the network as a binary matrix as

$$\mathbf{B} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad (29)$$

and ideally set $\alpha_1 = \mathbf{B}$. The geographical neighbors of each node i are the nodes that are geographically adjacent with it based on \mathbf{B} and the difference between v_i and v_j , and we consider three geographical neighbors for each node. After parameter setting, we generate the count data of the event-based network as follows: we obtain the CIF $\lambda_{ij}(t)$ of each node pair ij , with $i, j = 1, \dots, n$, at time t , with $t = 1, \dots, T$ by (13) and generate the event count $n_{ij}(t)$ by the mean values of data from a Poisson distribution, that is, $n_{ij}(t) = \frac{1}{R} \sum_{r=1}^R n_{ijr}(t)$, where $n_{ijr}(t) \sim \text{Poisson}(\lambda_{ij}(t))$. We take $\{\lambda_{ij}(t)\}$, with $i, j = 1, \dots, n$, and $t = 1, \dots, T\}$ or $\{n_{ij}(t)\}$, with $i, j = 1, \dots, n$, and $t = 1, \dots, T\}$ as one sample. Finally, we generate 100 samples of CIF and event count data for model training, 100 samples for model validation, and another 100

samples for model testing. Figure 3 presents examples of the simulated event-based dynamic network data, including the event counts and CIF values in the space and time domain.

We first select the values of tuning coefficients using validation data. The tuning coefficients to be selected include λ_1 , λ_2 and ρ . Recall from (15) that λ_1 and λ_2 both control the strength of the regularization terms, we set $\lambda_2 = 2\lambda_1$ without loss of generality. We consider various values of λ_1 and ρ , and estimate model parameters using training data under each pair values of λ_1 and ρ . We predict the expected event count at node ij by (13), and compare the predicted CIF with its true CIF values using validation data. To measure the count prediction accuracy, we define the mean absolute error (MAE) of CIF as the mean value of absolute differences between the predicted CIFs and true CIFs at all node pairs and all time. Figure 4 displays the MAEs of the proposed method under various values of λ_1 and ρ . In Figure 4(a), it is observed that the MAE becomes smaller as λ_1 and ρ increase, but will be inversely larger if ρ is too large. Specifically, we present the effect of ρ on MAEs when $\lambda_1 = 1 \times 10^{10}$ in Figure 4(b), and the effect of λ_1 on MAEs when $\rho = 1 \times 10^6$ in Figure 4(c). From these two figures, we finally select $\lambda_1 = 1 \times 10^{10}$ and $\rho = 1 \times 10^6$ that has the smallest MAE (i.e., 0.1694).

For the choice of the bandwidth δ , we propose a wide- and deep-integrated search strategy based on validation data as a guideline. We first implement the wide search strategy, that is, we obtain the MAEs of the proposed method by attempting a number of bandwidth values within a wide range, and focus on the bandwidth values within a narrow range that result in

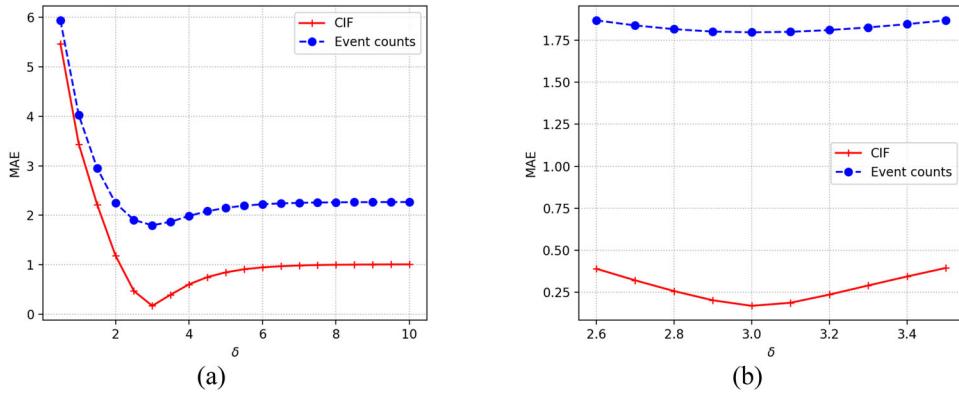


Figure 5. The wide- and deep-integrated search strategy for the choice of the bandwidth δ . (a) Wide search, and (b) deep search.

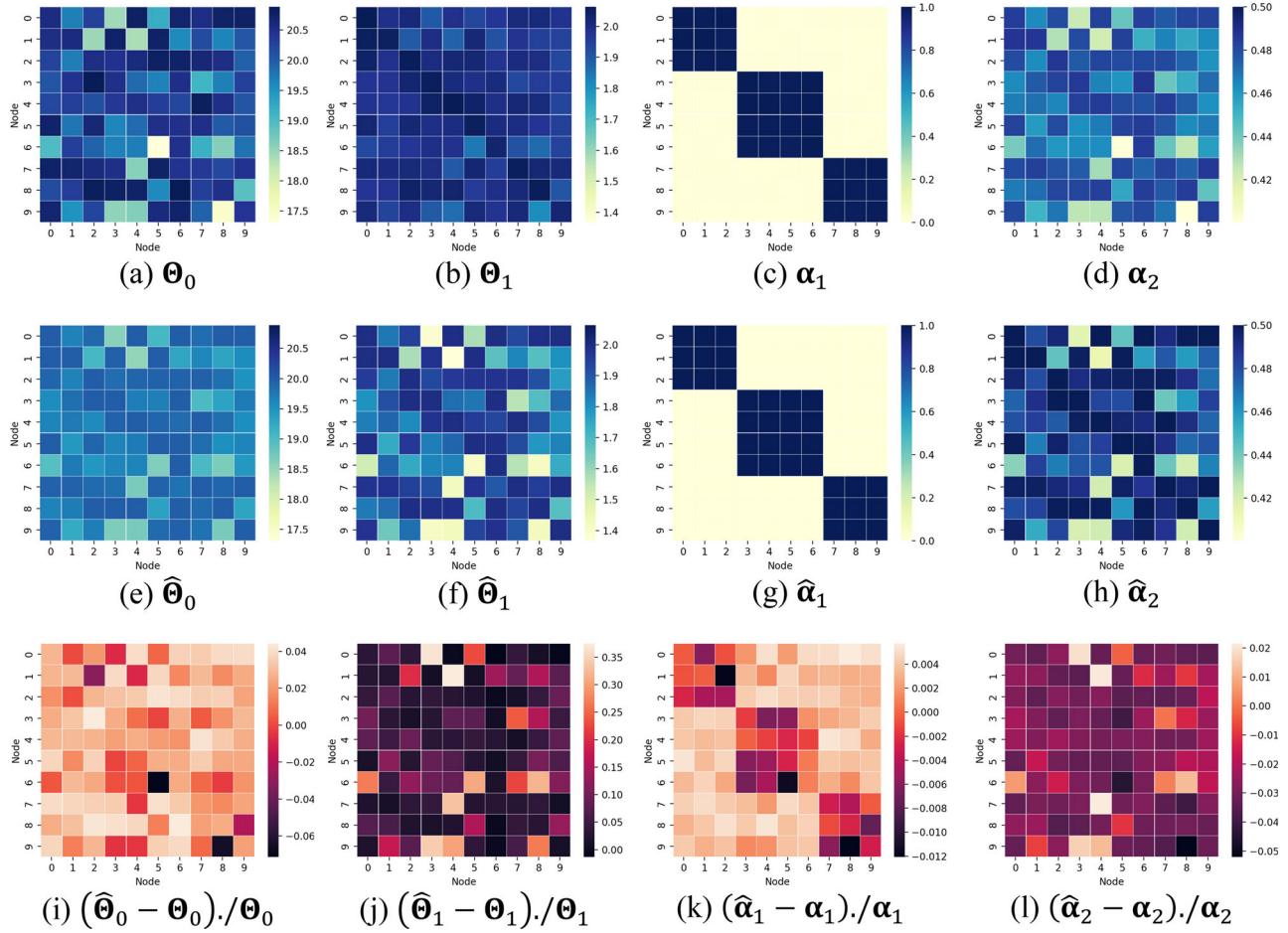


Figure 6. True and estimated values of model parameters in the simulation study.

relatively small MAEs values. Then, we employ the deep search strategy, that is, we arrange various bandwidth values within the narrow range, and select the bandwidth value that has the minimal MAE value. Figure 5 displays the results of wide- and deep-integrated search strategy. In the wide search strategy, we attempt the bandwidth values $\delta = 1, 1.5, 2, \dots, 10$, and obtain the MAEs using the proposed method. In Figure 5(a), the MAEs of counts and CIFs change obviously in the wide search strategy, and their relatively small values fall in the narrow range $\delta \in [2.5, 3.5]$. In the deep search strategy, we further set the bandwidth values $\delta = 2.5, 2.6, \dots, 3.5$ within the narrow range.

In Figure 5(b), the MAEs of counts change slightly because noises among the count data lead to the insensitivity to the bandwidth values within the narrow range. We select $\delta = 3$ that brings about the smallest MAE of CIFs.

Then, we employ the proposed ADMM algorithm to estimate the model parameters using training data. Model parameters include the background rate base term Θ_0 , the influence exciting term Θ_1 , the influence base term α_1 in the current period, and the influence base term in previous periods α_2 . Figure 6 presents the true values of Θ_0 , Θ_1 , α_1 , α_2 , the corresponding estimated values, as well as the relative differences between the true and

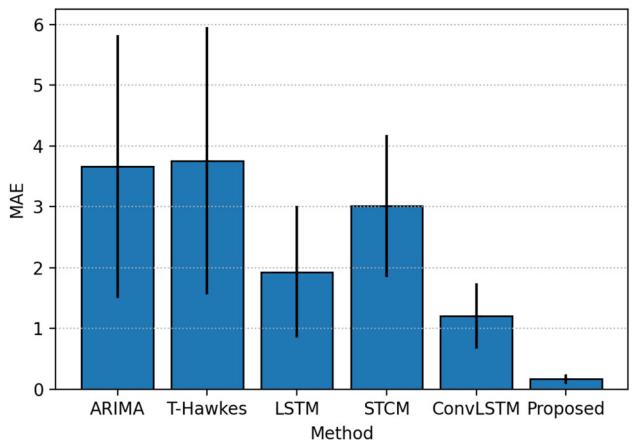
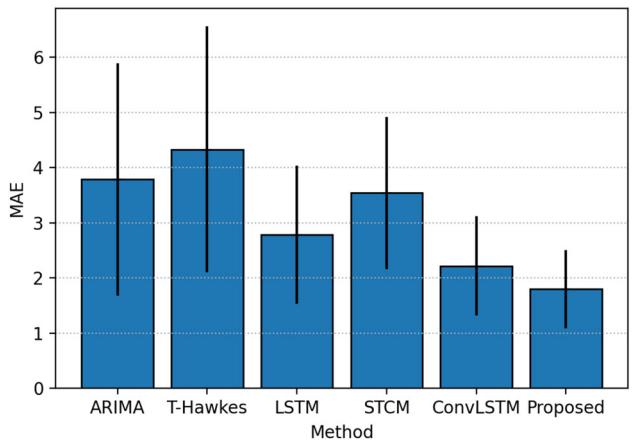
Table 1. RelErrs of estimated model parameters in the simulation study.

Model parameters	$\hat{\Theta}_0$	$\hat{\Theta}_1$	$\hat{\alpha}_1$	$\hat{\alpha}_2$
RelErr	0.0256	0.0813	0.0038	0.0269

estimated values. The estimated values of Θ_0 in Figure 6(e) are intuitively close to the true values in Figure 6(a), indicating that the background rate base term is captured accurately. The estimated values of α_1 are demonstrated in the heat map in Figure 6(g) and is very close to the true value shown in Figure 6(c), suggesting that α_1 is learned very well by fully considering the network spatial information (i.e., B). Θ_1 and α_2 are also well estimated by comparing Figure 6(b) (d) (f), and (h). To evaluate the accuracy of parameter estimation persuasively, we set the metric RelErr that is defined as the mean of relative errors between true values and estimated values for all elements in a parameter (i.e., $\text{RelErr} = \frac{1}{n^2} \sum_{i=1}^n \sum_{j=1}^n |\theta_{ij}^{(1)} - \hat{\theta}_{ij}^{(1)}| / \theta_{ij}^{(1)}$ if we take the evaluation of Θ_1 as an example). As presented in Table 1, the RelErrs for all parameters are small, and thus our proposed method can estimate parameters accurately.

We consider five competitive methods as benchmarks to compare their prediction performance for the event-based network: (a) ARIMA: autoregressive integrated moving average models for count prediction at each node, respectively; (b) Time-varying Hawkes (T-Hawkes): Hawkes processes that vary with time for each node, respectively (Daley and Vere-Jones 2003); (c) LSTM: LSTM models for count prediction at each node, respectively; (d) Spatio-temporal count model (STCM): a spatial count data model that captures the CIF of Poisson distribution via a vector autoregressive framework for spatiotemporal count prediction of the whole network (Buddhavarapu, Bansal, and Prozzi 2021); (e) ConvLSTM: a convolutional LSTM model for count prediction of the whole network (Chu, Lam, and Li 2020). ARIMA, T-Hawkes, and LSTM model time dynamics at each node without consideration of spatial correlation among nodes, whereas STCM and ConvLSTM consider spatiotemporal correlation among nodes in the network.

We evaluate the model performance using testing dataset, and measure the count prediction accuracy via the MAEs of CIF and counts. Specifically, we predict the event counts of the network as the CIF in our proposed model, and compare the predicted CIFs with the corresponding true CIF and count values of the testing dataset. As shown in Figure 7, we present the prediction errors of the proposed and benchmark methods for CIF, where the bar and error bar for each method represent the mean value and one half of the standard deviation of the MAEs, respectively. ARIMA, T-Hawkes and LSTM capture the evolution of events over time but ignores spatial interaction, and thus perform poorly compared with STCM, ConvLSTM and the proposed method. As a deep learning model, LSTM performs better than ARIMA and T-Hawkes because it has advantages to capture complex time-varying patterns of counts given sufficient available data. STCM further improves the model performance by considering the spatial influence among nodes in the network. ConvLSTM captures spatial influence among nodes as well and has superiority over STCM due to its advantages to capture complex spatiotemporal dynamic patterns as a deep learning model. Compared with all benchmarks, the proposed

**Figure 7.** MAEs of the proposed and benchmark methods for event counts in the simulation study.**Figure 8.** MAEs of the proposed and benchmark methods for CIF in the simulation study.

method yields the prediction results with the least errors. The proposed method considers spatiotemporal dynamics and interactive influence patterns based on spatial structural knowledge, and thus has superior model performance. In addition, for methods with small mean values of MAEs, the corresponding standard deviations have small values as well. The proposed method has the smallest standard deviation, which indicates its high accuracy and robustness. Figure 8 presents the MAEs of the proposed and benchmark methods for event counts. Given that the event counts are randomly generated by the Poisson distribution with the true CIF as the rate, the variances and uncertainties are intensified because noises exist in the count data. As a result, the means and standard deviations of the MAEs using the benchmarks and the proposed method increase. Evidently, the proposed method still has the smallest prediction deviations among all benchmarks. In particular, the MAE value of counts improves $(2.2191 - 1.7973)/2.2191 = 19.01\%$ using the proposed using the propose method than ConvLSTM that has the best performance among the benchmark methods. Examples of event count prediction results of the proposed and benchmark methods on some node pairs are provided in Appendix E.

As demonstrated in (13), there are a few triggering patterns in the proposed model. To analyze the triggering effect of each pattern in the proposed model, we present the importance of

the triggering patterns and provide the model performance by eliminating each triggering pattern. Table 3 shows the mean values and standard deviations of MAEs using the proposed model and the ones that eliminate each triggering pattern separately. The proposed model has the smallest MAE by fully considering triggering effects among the network. Eliminating any of the triggering patterns increases MAEs to some extent. Especially, eliminating $\theta_{ij}^{(0)}$ increases the MAE most. Recall that $\theta_{ij}^{(0)}$ represents the background rate base term. If $\theta_{ij}^{(0)}$ is eliminated, the trained model ignores the background information that actually exists, and the predicted CIFs and counts have poor accuracy. With $\alpha_{ij}^{(1)}$

and $\frac{1}{|D_i|} \sum_{d \in D_i} \alpha_{ij}^{(2)} \sum_{t_d^p < t} e^{-\frac{(t-t_d^p)}{\delta}} n_{ij}^{(d)}(t_d^p)$, the influence base terms in current and previous periods are captured, and thus it can be observed that model performance is largely reduced when eliminating them. $\sum_{h \in H_j} \theta_{hj}^{(1)} \sum_{t_h^c < t} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c) w_{ih}$, $\sum_{k \in K_j} \theta_{ik}^{(1)} \sum_{t_k^c < t} e^{-\frac{(t-t_k^c)}{\delta}} n_{ik}(t_k^c) w_{kj}$ and $\frac{1}{|M_i|} \sum_{m \in M_i} \theta_{mi}^{(1)} \sum_{t_m^c < t} e^{-\frac{(t-t_m^c)}{\delta}} n_{mi}(t_m^c)$ model the influence exciting terms. If one of them is eliminated, the MAE becomes large but not too much because the remaining terms can partially complement its effect. We also analyze the effect of spatial structure knowledge on the proposed model by eliminating the spatial information matrix \mathbf{B} as the regularization term when estimating model parameters. We can see that the MAEs of CIFs and counts are 0.4630 and 1.9041, which are larger than the ones of the proposed model (i.e., 0.1694, 1.7973), indicating the necessity and effectiveness of considering spatial structure knowledge. In addition, we provide model computational time in Tables 2 and 3. The computer we use has "Apple M1" CPU with 8 cores, 8 GB memory, integrated graphics card, and 512 GB solid state hard disk. Compared

Table 2. Model training time of the proposed and benchmark methods in the simulation study.

Models	ARIMA	T-Hawkes	LSTM	STCM	ConvLSTM	Proposed
Model training time /s	106	43	1155	<1	677	25

Table 3. Analysis of triggering effects in (13) of the proposed model.

Models	MAEs of CIFs	MAEs of counts	Model training time /s
Without $\theta_{ij}^{(0)}$	2.5168 (4.9278)	3.5695 (4.8062)	388
Without $\alpha_{ij}^{(1)}$	0.8117 (1.0790)	2.0886 (1.6303)	23
Without $\sum_{h \in H_j} \theta_{hj}^{(1)} \sum_{t_h^c < t} e^{-\frac{(t-t_h^c)}{\delta}} n_{hj}(t_h^c) w_{ih}$	0.4765 (0.4582)	1.8654 (1.4711)	149
Without $\sum_{k \in K_j} \theta_{ik}^{(1)} \sum_{t_k^c < t} e^{-\frac{(t-t_k^c)}{\delta}} n_{ik}(t_k^c) w_{kj}$	0.4712 (0.4507)	1.8632 (1.4693)	149
Without $\frac{1}{ M_i } \sum_{m \in M_i} \theta_{mi}^{(1)} \sum_{t_m^c < t} e^{-\frac{(t-t_m^c)}{\delta}} n_{mi}(t_m^c)$	0.4973 (0.4297)	1.8697 (1.4670)	135
Without $\frac{1}{ D_i } \sum_{d \in D_i} \alpha_{ij}^{(2)} \sum_{t_d^p < t} e^{-\frac{(t-t_d^p)}{\delta}} n_{ij}^{(d)}(t_d^p)$	2.1042 (2.1947)	2.9008 (2.4617)	163
Without spatial information \mathbf{B}	0.4630 (0.6596)	1.9041 (1.4931)	40
Proposed	0.1694 (0.1495)	1.7973 (1.4251)	25

NOTE: Values in parentheses denote the standard deviations of the MAEs.

with the benchmark methods, the proposed method implements model computation efficiently with acceptable training time guided by prior knowledge such as graph regularization.

In order to further analyze the effect of noises in count data on model performances, we supplement experiments under scenarios with different levels of noises. Recall that the counts are generated by the mean values of data from the Poisson distribution, that is, $n_{ij}(t) = \frac{1}{R} \sum_{r=1}^R n_{ijr}(t)$, where $n_{ijr}(t) \sim \text{Poisson}(\lambda_{ij}(t))$. Here, the larger the value of R , the smaller the noises are in the generated count data. We devise scenarios with various values of R to generate count data with different levels of noises, that is, $R = 1, 5, 10, \dots, 50$, and compare the mean values of MAEs using the proposed and benchmark methods under all scenarios. The results are displayed in Figure 9(a) and (b) with respect to the CIFs and counts, respectively. When R is small, indicating there are large noises in count data, all benchmark methods have large MAEs of CIFs and counts. As R increases, the noises become small, leading to better and more robust performance of all the methods. By comparison, the proposed method delivers fairly minimal MAEs of CIFs under all noise levels. The MAEs of counts are interfered by noises, especially when the noises are large, but the proposed method achieves the smallest MAEs under all noise levels.

Moreover, we conduct another simulation study on a large-scale network with 1000 nodes in Appendix F of the supplementary materials. The results demonstrate the proposed method has the potential to effectively model a large-scale network.

5. Real Case Study

We conduct a real case study based on the New York yellow taxi dataset, which was publicly available online and collected by technology providers authorized under the Taxicab and Livery Passenger Enhancement Programs (NYC Taxi 2017). The dataset includes all yellow taxi trips in the past few years. The pick-up and drop-off dates and times, pick-up and drop-off locations, trip distances, and payment information for all trips are recorded in this dataset. To illustrate the main idea of the proposed method, we focus on the network with 19 busiest zones in Manhattan and investigate the modeling of travel demand counts on these zones within each day. We illustrate the structure

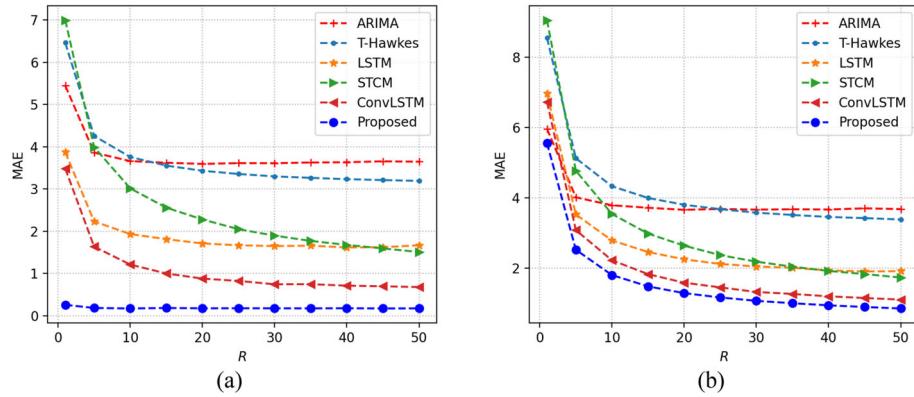


Figure 9. Mean values of MAEs using the proposed and benchmark methods under scenarios with various noise levels in the simulation study. (a) CIFs and (b) counts.

of the traffic network in Figure 10 for details, where the zones denote nodes in the network with $n = 19$. We collect the traffic network information, including the origin as the source node and the destination as the target node. For the time scope, we observe the demand counts of each OD pair per hour from corresponding taxi trips within the whole day. We use the data at the first three weeks and the last week in a month of one year as the training data and testing data, respectively. The training and testing data include a total of 743,292 and 259,771 taxi trips, respectively. Furthermore, the counts are aggregated to 1-hr interval, which translates to 181,944 counts (i.e., 24 hours \times 21 days \times 19^2 OD pairs) and 60,648 counts (i.e., 24 hours \times 7 days \times 19^2 OD pairs), respectively. For prior spatial structure knowledge, we consider the attractiveness of node pairs that provides useful information for traffic OD demand network by recalling Section 3.1.2, where the attractiveness B_{ij} from i to j is defined as $B_{ij} = D_{ij}/\text{dis}(i,j)$. We calculate the log-transformation of monthly average number of demands D_{ij} using the historical demand data in the last year and the distance $\text{dis}(i,j)$ using the average of trip distances recorded in the data directly. In a day, we predict the demand counts of each OD pair in the traffic network within some hour using the proposed method.

We consider the nearest three geographical neighbors based on distances between nodes, and determine the tuning parameters by cross validation as $\lambda_1 = \rho = 10$, $\lambda_2 = 20$, $\eta = 0.01$, and $\delta = 3$. In parameter estimation procedure, we propose the objective function of the proposed SIHP with regularization from spatial structural knowledge and solve it by ADMM. Figure 11 presents the estimated $\hat{\Theta}_0(t)$ at eight time points. Recall that $\hat{\Theta}_0(t)$ is time-varying background rate base term within a whole day. Given that few events occur after midnight, the estimated values $\hat{\Theta}_0(t_2)$ and $\hat{\Theta}_0(t_5)$ are small and even close to zero. $\hat{\Theta}_0(t_8)$, $\hat{\Theta}_0(t_{11})$, $\hat{\Theta}_0(t_{14})$, and $\hat{\Theta}_0(t_{17})$ are larger than $\hat{\Theta}_0(t_{20})$ and $\hat{\Theta}_0(t_{23})$ as more events occur in the daytime than the night. The other estimated parameters (i.e., $\hat{\Theta}_1$, $\hat{\alpha}_1$, $\hat{\alpha}_2$) are shown in Figure 12. $\hat{\Theta}_1$, $\hat{\alpha}_1$, $\hat{\alpha}_2$ can capture spatial and temporal interaction of the traffic network. We take the node pair ij as an example. Recall that we consider the influence from the geographical neighbors hj for the source node i and ik for the target node j , that is, $\beta_{ij,hj}^{(1)} = \alpha_{ij}^{(1)} \cdot \theta_{hj}^{(1)} \cdot w_{ih}$ and $\beta_{ij,ik}^{(1)} = \alpha_{ij}^{(1)} \cdot \theta_{ik}^{(1)} \cdot w_{kj}$ as formulated in (6) and (9). The larger values

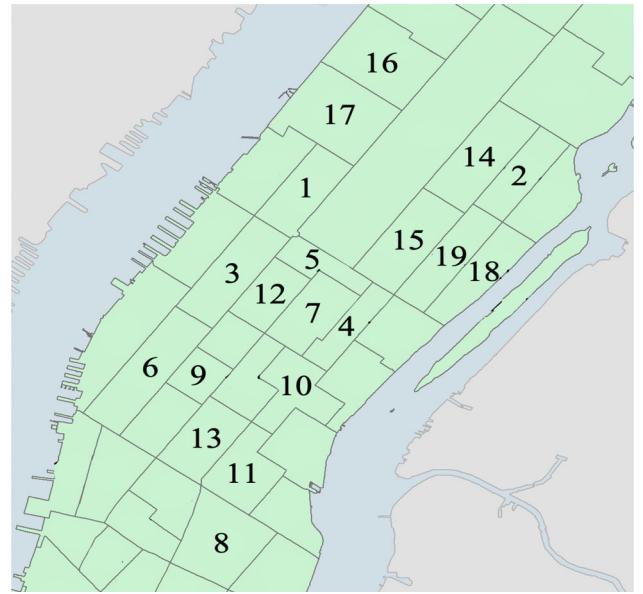


Figure 10. Yellow taxi zones in the case study.

in $\beta_{ij,hj}^{(1)}$ and $\beta_{ij,ik}^{(1)}$ indicate the interactions from hj and ik to the node pair ij are stronger. Recall that α_2 represents the influence base term in previous periods, and a large value of $\alpha_{ij}^{(2)}$ indicates the node pair ij is much induced by itself in previous periods.

Subsequently, we predict the event counts using the proposed method with estimated parameters and compare the performance of the proposed method with benchmark methods, which are provided in Table 4. We present the mean values and standard deviations of MAEs for count prediction. ARIMA, T-Hawkes and LSTM only capture the evolution of events over time at each node while ignoring the interactions among nodes, and thus perform poorly compared with the proposed method. ConvLSTM further captures spatial influence among nodes and thus has superiority over LSTM as a deep learning model. Although STCM attempts to model spatiotemporal dynamics of the network, it is restricted by the assumptions in the vector autoregressive model that cannot be satisfied in the case study of traffic demands. Therefore, STCM fails to predict traffic demands effectively with a large MAE. Overall, the proposed method yields the prediction results with the least errors. Compared with the benchmarks, the proposed

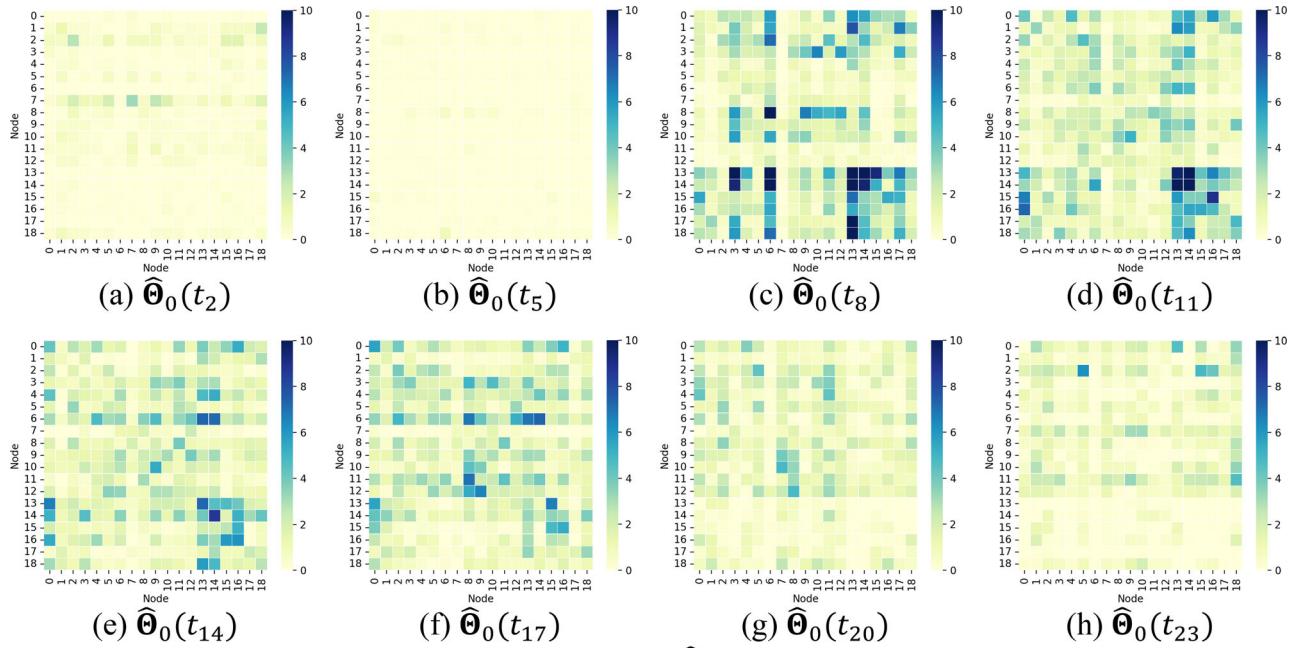


Figure 11. Estimated background rate base term $\hat{\Theta}_0(t)$ at eight time points in the case study.

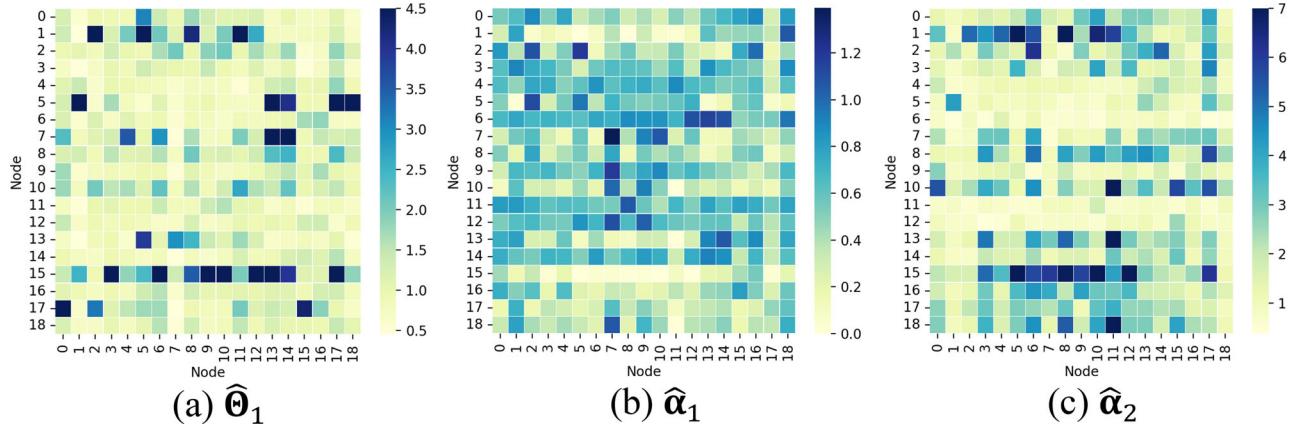


Figure 12. Other estimated model parameters in the case study.

Table 4. Model performances of the proposed and benchmark methods in the case study.

Models	MAEs of Counts	Model training time /s
ARIMA	1.9514 (2.4864)	97
T-Hawkes	2.1294 (2.7152)	83
LSTM	2.1284 (2.9020)	1218
STCM	2.2245 (2.7889)	<1
ConvLSTM	1.9306 (2.5122)	698
Proposed	1.7058 (2.0545)	409

NOTE: Values in parentheses denote the standard deviations of the MAEs.

method considers spatiotemporal dynamics and interactive influence patterns based on spatial structural knowledge, and thus performs the best with the smallest mean values and standard deviations of MAEs. Moreover, the training time of the proposed method is acceptable among all the methods.

Figure 13 further presents the specific taxi predicted demands of the OD pairs (15, 15) at test days 3 and 4, and (1, 17) at test days 6 and 7. In this figure, the blue lines with circle dots

represent the actual observations of dynamic traffic demand counts, where the actual taxi demand indeed exhibits high temporal evolution as expected. The red lines with star dots are the predicted demands using the proposed method. The prediction results of the proposed method adapt to the characteristics of the online observations dynamically and align with the actual observations, as the proposed prediction method exploits influence patterns with neighborhood information and temporal evolution learned from the training data and collectively uses the online observations of the OD pairs. The black lines with plus dots are the predicted demands by T-Hawkes, which perform unsatisfactorily with severely lagged predicted demands. The predicted demands by ARIMA (orange lines with right triangle dots) and LSTM (green lines with upper triangle dots) cannot capture the trends of demands accurately. Although STCM and ConvLSTM consider spatial correlation among nodes to construct an overall spatiotemporal model, demands are still poorly predicted at some node pairs (e.g., (15, 15)). Overall, the

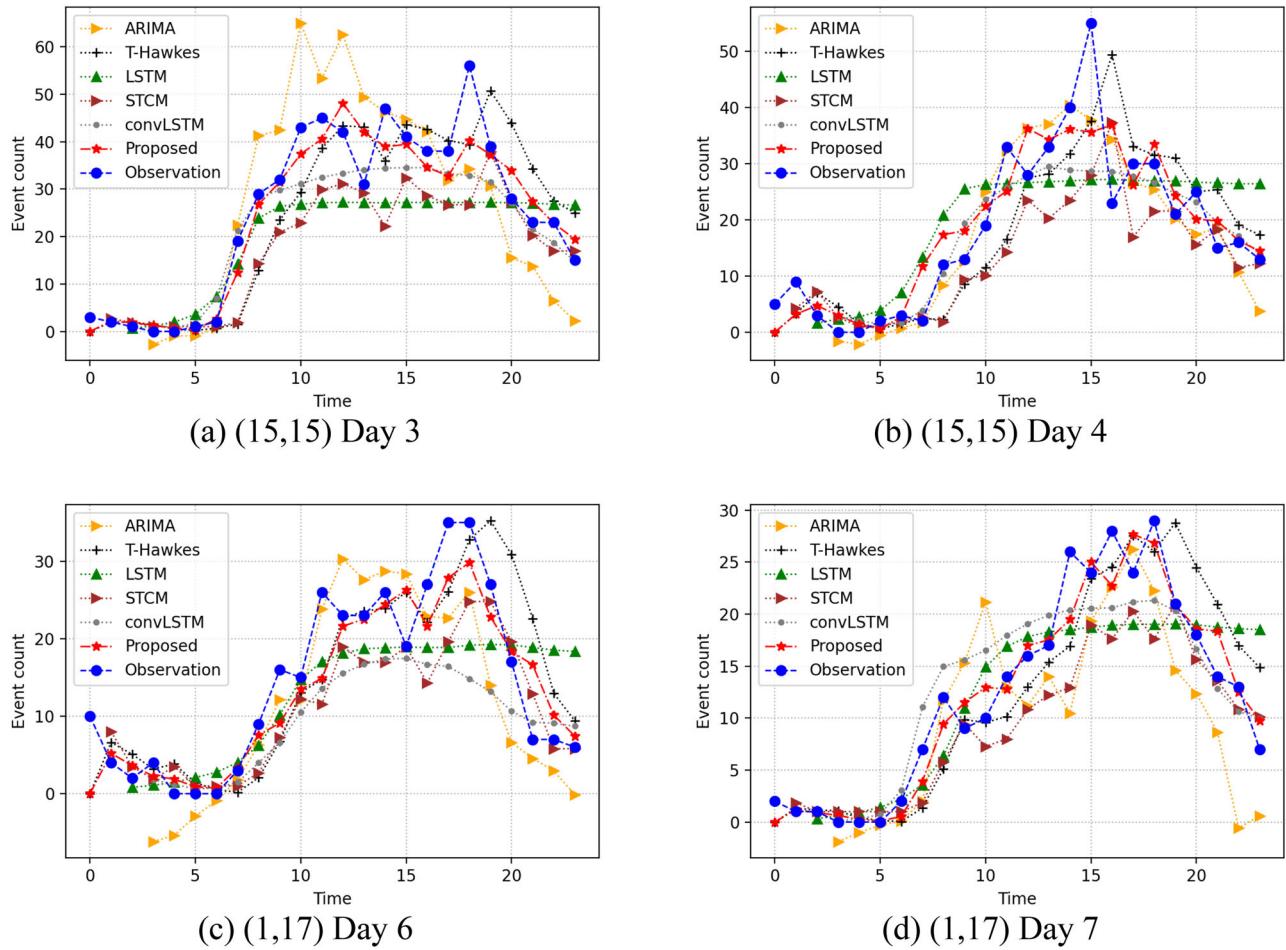


Figure 13. Examples of demands predicted by the proposed and benchmark methods.

proposed method has more satisfactory prediction results at the testing days than the benchmarks.

6. Conclusion

Event-based dynamic networks, in which interaction events that dynamically vary with space and time occur, widely exist in practice. Influence patterns and triggering motivations for the interaction events reflect the nature and semantics of human/object behaviors in network, and thus modeling event-based dynamic networks is critically important for the success of human/object-centric applications. In this article, we develop a SIHP to learn the influence patterns from historical interaction events on later ones and understand the dynamics on networks by incorporating spatial structure knowledge effectively. Given the historical event information in the network, the SIHP explicitly models the rate of interaction events between any pair of nodes in the network by considering neighboring nodes, including geographical neighbors and semantic neighbors. In addition, we incorporate the knowledge of spatial structures as regularization into SIHP and then learn the influence patterns between event nodes by maximizing the likelihood of the SIHP with regularization via the parameter estimation algorithm based on the ADMM framework.

Two related topics will be considered in the future work. First, scenarios with multiple types of variables for capturing

a network is common in practice, leading to multiple types of interactions between two nodes. Future work will extend to develop multilayer models to characterize the correlated multilayer interactions among networks. Second, sparse networks exist in some specific applications. Thus, an important future work is to incorporate sparsity and low rank to construct a more general framework for large-scale event-based network modeling.

Supplementary Materials

There are two supplementary materials. (i) Technical details on the log-likelihood function of the proposed method and updating procedures of model parameters; Additional figures of the simulation study in Section IV and a simulation study on a large-scale network. (ii) A Python code file that contains procedures to reproduce Figure 6, that is, true and estimated values of model parameters in the simulation study.

Acknowledgments

The authors thank the editor, associate editor and the referees for their useful comments and suggestions.

Disclosure Statement

The authors report no conflict of interest.

Funding

Wang's work was supported in part by the National Natural Science Foundation of China under Grant 72471145 and Grant 72101148, and in part by the National Natural Science Foundation of Shanghai under Grant 22ZR1433000. Xian's work was supported by the National Institutes of Health under Grant 1R21EB033455. Li's work was supported by the National Natural Science Foundation of China under Grant 72201006.

References

- Achcar, J. A., Martinez, E. Z., Souza, A. D. P. D., Tachibana, V. M., and Flores, E. F. (2011), "Use of Poisson Spatiotemporal Regression Models for the Brazilian Amazon Forest: Malaria Count Data," *Revista da Sociedade Brasileira de Medicina Tropical*, 44, 749–754. [295]
- Ashok, K., and Ben-Akiva, M. E. (2002), "Estimation and Prediction of Time-Dependent Origin-Destination Flows with a Stochastic Mapping to Path Flows and Link Flows," *Transportation Science*, 36, 184–198. [295]
- Bahdanau, D., Cho, K., and Bengio, Y. (2014), "Neural Machine Translation by Jointly Learning to Align and Translate," arXiv preprint arXiv:1409.0473. [298]
- Bera, S., and Rao, K. V. (2011), "Estimation of Origin-Destination Matrix from Traffic Counts: The State of the Art," *European Transport*, 49, 2–23. [295]
- Buddhavarapu, P., Bansal, P., and Prozzi, J. A. (2021), "A New Spatial Count Data Model with Time-Varying Parameters," *Transportation Research Part B: Methodological*, 150, 566–586. [295,301,304]
- Butts, C. T., Lomi, A., Snijders, T. A. B., and Stadtfeld, C. (2023), "Relational Event Models in Network Science," *Network Science*, 11, 175–183. [295]
- Cai, D., He, X., Han, J., and Huang, T. S. (2011), "Graph Regularized Nonnegative Matrix Factorization for Data Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33, 1548–1560. [298]
- Chu, K., Lam, A. Y. S., and Li, V. O. K. (2020), "Deep Multi-Scale Convolutional LSTM Network for Travel Demand and Origin-Destination Predictions," *IEEE Transactions on Intelligent Transportation Systems*, 21, 3219–3232. [293,304]
- Conn, P. B., Johnson, D. S., Hoef, J. M. V., Hooten, M. B., London, J. M., and Boveng, P. L. (2015), "Using Spatiotemporal Statistical Models to Estimate Animal Abundance and Infer Ecological Dynamics from Survey Counts," *Ecological Monographs*, 85, 235–252. [294]
- Daley, D. J., and Vere-Jones, D. (2003), *An Introduction to the Theory of Point Processes Volume 1: Elementary Theory and Methods*, New York: Springer. [295,296,304]
- Dong, H., and Wang, K. (2022), "Interaction Event Network Modeling Based on Temporal Point Process," *IIE Transactions*, 54, 630–642. [293,295]
- Dong, H., Chen, N., and Wang, K. (2020), "Modeling and Change Detection for Count-Weighted Multilayer Networks," *Technometrics*, 62, 184–195. [294,295]
- Du, H., Zhou, Y., Ma, Y., and Wang, S. (2021), "Astrologer: Exploiting Graph Neural Hawkes Process for Event Propagation Prediction with Spatio-Temporal Characteristics," *Knowledge-Based Systems*, 228, 107247. [296]
- DuBois, C., Butts, C. T., and Smyth, P. (2013), "Stochastic Blockmodeling of Relational Event Dynamics," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pp. 238–246. [296]
- Embrechts, P., Liniger, T., and Lin, L. (2011), "Multivariate Hawkes Processes: An Application to Financial Data," *Journal of Applied Probability*, 48, 367–378. [294]
- Farajtabar, M., Wang, Y., Gomez-Rodriguez, M., Li, S., Zha, H., and Song, L. (2015), "COEVOLVE: A Joint Point Process Model for Information Diffusion and Network co-Evolution," in *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pp. 1954–1962. [295]
- Flaxman, S., Wilson, A., Neill, D., Nickisch, H., and Smola, A. (2015), "Fast Kronecker Inference in Gaussian Processes with non-Gaussian Likelihoods," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning*, pp. 607–616, ACM. [294]
- Hall, E. C., and Willett, R. M. (2016), "Tracking Dynamic Point Processes on Networks," *IEEE Transactions on Information Theory*, 62, 4327–4346. [295]
- Hu, H., Lin, Z., Hu, Q., and Zhang, Y. (2022), "Attention Mechanism with Spatial-Temporal Joint Model for Traffic Flow Speed Prediction," *IEEE Transactions on Intelligent Transportation Systems*, 23, 16612–16621. [295]
- Ilhan, F., and Kozat, S. S. (2020), "Modeling of Spatio-Temporal Hawkes Processes with Randomized Kernels," *IEEE Transactions on Signal Processing*, 68, 4946–4958. [296]
- Junuthula, R., Haghdan, M., Xu, K. S., and Devabhaktuni, V. (2019), "The Block Point Process Model for Continuous-Time Event-Based Dynamic Networks," in *The World Wide Web Conference*, pp. 829–839. [294,295]
- Kamarianakis, Y., Shen, W., and Wynter, L. (2012), "Real-Time Road Traffic Forecasting Using Regime-Switching Space-Time Models and Adaptive Lasso," *Applied Stochastic Models in Business and Industry*, 28, 297–315. [295]
- Lerner, J., Bussmann, M., Snijders, T., and Brandes, U. (2013), "Modeling Frequency and Type of Interaction in Event Networks," *Corvinus Journal of Sociology and Social Policy*, 4, 3–32. [295]
- Li, G., Knoop, V. L., and Lint, H. (2021), "Multistep Traffic Forecasting by Dynamic Graph Convolution: Interpretations of Real-Time Spatial Correlations," *Transportation Research Part C: Emerging Technologies*, 128, 103185. [294,295]
- Liu, L., Qiu, Z., Li, G., Wang, Q., Ouyang, W., and Lin, L. (2019), "Contextualized Spatial-Temporal Network for Taxi Origin-Destination Demand Prediction," *IEEE Transactions on Intelligent Transportation Systems*, 20, 3875–3887. [295]
- Liu, Y., Liu, C., Liu, B., Qu, M., and Xiong, H. (2016), "Unified Point-of-Interest Recommendation with Temporal Interval Assessment," in *Proceedings of the 22nd International Conference on Knowledge Discovery & Data Mining*, pp. 1015–1024, ACM. [295]
- Liu, Y., Yan, T., and Chen, H. (2018), "Exploiting Graph Regularized Multi-Dimensional Hawkes Processes for Modeling Events with Spatio-Temporal Characteristics," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 2475–2482. [294,296]
- Marsan, D., and Lengline, O. (2008), "Extending Earthquakes' Reach through Cascading," *Science*, 319, 1076–1079. [295]
- Moreira-Matias, L., Gama, J., Ferreira, M., Mendes-Moreira, J., and Damas, L. (2013), "Predicting Taxi-Passenger Demand Using Streaming Data," *IEEE Transactions on Intelligent Transportation Systems*, 14, 1393–1402. [295]
- Newman, M. E. J., Watts, D. J., and Strogatz, S. H. (2002), "Random Graph Models of Social Networks," in *Proceedings of the National Academy of Sciences of the United States of America*, 99 Suppl 1, 2566–2572. [294]
- NYC Taxi. (2017), available at http://www.nyc.gov/html/tlc/html/about_trip_record_data.shtml. [305]
- Perrakis, K., Karlis, D., Cools, M., Janssens, D., Vanhoof, K., and Wets, G. (2012), "A Bayesian Approach for Modeling Origin-Destination Matrices," *Transportation Research Part A: Policy and Practice*, 46, 200–212. [295]
- Rasmussen, J. G. (2013), "Bayesian Inference for Hawkes Processes," *Methodology and Computing in Applied Probability*, 15, 623–642. [294,295,296]
- Schuijbroek, J., Hampshire, R. C., and van Hoeve, W.-J. (2017), "Inventory Rebalancing and Vehicle Routing in Bike Sharing Systems," *European Journal of Operational Research*, 257, 992–1004. [293]
- Soliman, H., Zhao, L., Huang, Z., Paul, S., and Xu, K. S. (2022), "The Multivariate Community Hawkes Model for Dependent Relational Events in Continuous-Time Networks," in *Proceedings of the 39th International Conference on Machine Learning*, pp. 20329–20346. [296]
- Tobler, W. R. (1970), "A Computer Movie Simulating Urban Growth in the Detroit Region," *Economic Geography*, 46, 234–240. [301]
- Wang, P., Fu, Y., Liu, G., Hu, W., and Aggarwal, C. (2017), "Human Mobility Synchronization and Trip Purpose Detection with Mixture of Hawkes Processes," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 495–503, ACM. [293]
- Wang, Y., Yin, H., Chen, H., Wo, T., Xu, J., and Zheng, K. (2019), "Origin-Destination Matrix Prediction via Graph Convolution: A New Perspective of Passenger Demand Modelling," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1227–1235, ACM. [293]

- Wei, E., and Ozdaglar, A. (2012), "Distributed Alternating Direction Method of Multipliers," in *Proceedings of IEEE 51st IEEE Conference on Decision and Control*, pp. 5445–5450. [299]
- Wheatley, S., Filimonov, V., and Sornette, D. (2016), "The Hawkes Process with Renewal Immigration and Its Estimation with an EM Algorithm," *Computational Statistics & Data Analysis*, 94, 120–135. [294]
- Xian, X., Ye, H., Wang, X., and Liu, K. (2021), "Spatiotemporal Modeling and Real-Time Prediction of Origin-Destination Traffic Demand," *Tech-nometrics*, 63, 77–89. [294,295,301]
- Xiao, S., Yan, J., Yang, X., Zha, H., and Chu, S. M. (2017), "Modeling the Intensity Function of Point Process via Recurrent Neural Networks," in *Proceedings of the 31rst AAAI Conference on Artificial Intelligence*, pp. 1597–1603.
- Xu, C., Tao, D., and Xu, C. (2013), "A Survey on Multi-View Learning." *arXiv preprint arXiv:1304.5634*. [299]
- Xu, H., Wu, W., Nemati, S., and Zha, H. (2017), "Patient Flow Prediction via Discriminative Learning of Mutually-Correcting Processes," *IEEE Transactions on Knowledge and Data Engineering*, 29, 157–171. [295]
- Yan, X., Wang, W., Gao, Z., and Lai, Y. (2017), "Universal Model of Individual and Population Mobility on Diverse Spatial Scales," *Nature Communications*, 8, 1639–1639. [293,295,299]
- Zhang, D., Feng, X., Shen, M., and Zhong, S. (2021a), "DNEAT: A Novel Dynamic Node-Edge Attention Network for Origin-Destination Demand Prediction," *Transportation Research Part C: Emerging Technologies*, 122, 102851. [295]
- Zhang, M., Xu, C., Sun, A., Qiu, F., and Xie, Y. (2021b), "Solar Radiation Anomaly Events Modeling Using Spatial-Temporal Mutually Interactive Processes." *arXiv:2101.11179*, 1–36. [293,295]
- Zhou, K., Zha, H., and Song, L. (2013), "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. [294,296,301]
- Zhu, L., Yu, F. R., Wang, Y., Ning, B., and Tang, T. (2019), "Big Data Analytics in Intelligent Transportation Systems: A Survey," *IEEE Transactions on Intelligent Transportation Systems*, 20, 383–398. [299]