Modeling of Spatio-Temporal Hawkes Processes With Randomized Kernels

Fatih Ilhan and Suleyman S. Kozat, *Senior Member, IEEE*

***Abstract*—We investigate spatio-temporal event analysis using point processes. Inferring the dynamics of event sequences spatio-temporally has many practical applications including crime prediction, social media analysis, and traffic forecasting. In particular, we focus on spatio-temporal Hawkes processes that are commonly used due to their capability to capture excitations between event occurrences. We introduce a novel inference framework based on randomized transformations and gradient descent to learn the process. We replace the spatial kernel calculations by randomized Fourier feature-based transformations. The introduced randomization by this representation provides flexibility while modeling the spatial excitation between events. Moreover, the system described by the process is expressed within closed-form in terms of scalable matrix operations. During the optimization, we use maximum likelihood estimation approach and gradient descent while properly handling positivity and orthonormality constraints. The experiment results show the improvements achieved by the introduced method in terms of fitting capability in synthetic and real-life datasets with respect to the conventional inference methods in the spatio-temporal Hawkes process literature. We also analyze the triggering interactions between event types and how their dynamics change in space and time through the interpretation of learned parameters.**

***Index Terms*—Parameter estimation, time series, system modeling, point processes, random Fourier features, event analysis.**

I. Introduction

A. Preliminaries

We study spatio-temporal event analysis using point processes, which has several applications in signal processing, computer networks, security and forecasting applications [1]–[5]. Most of the real-world events exhibit certain spatio-temporal patterns such as correlation, causation, and excitation, which can be modeled as a system whose latent structure is reflected into real-world with their realizations. Modeling and learning this structure is important due to its promising applications such as network analysis, event prediction and hotspot detection [6]–[13]. In this context, we analyze the triggering relations between events in a given sequence, and how these interactions evolve in space and time, which can be useful for forecasting and policy planning for security and business applications [7], [13]. To this end, we model the event sequence using point processes, which directly represent the underlying structure of spatio-temporal excitations by their internal parameters. Therefore, inferring these parameters provides an interpretable and forthright way to analyze the given spatio-temporal data.

Point processes are used to capture the dynamics of the event sequence by expressing their rate of occurrences with an intensity function conditioned on the history [14]. In our problem, events are described by their locations, times and types. Therefore, we consider a multi-dimensional form of point processes called as spatio-temporal point processes [11], [15], [16]. We particularly work on spatio-temporal Hawkes processes that have a self-exciting nature by their default form, in which the intensity value is triggered by past events. In this approach, the excitation between events is usually modeled as to be decaying exponentially in time with an exponential kernel, and in space with a Gaussian kernel [11], [16].

Although modeling of temporal excitation with exponential decay is shown to be effective in most scenarios, the assumption that spatial excitation can be completely represented with a Gaussian kernel may not hold in all cases. Hence, we introduce certain degree of randomness to the spatial kernel by using randomized kernel representation. We utilize random Fourier features [17], which approximate the output of a shift-invariant continuous kernel using the inner products of the embedded vectors. Flexing the structure of the spatial kernel enables our model to capture excitation without purely Gaussian decay in the spatial domain. The number of dimensions in the transformed feature space is a hyperparameter, which directly controls the randomization effect, thus we can readily tune it depending on the spatial characteristics of the given event sequence. In addition, replacing the pairwise kernel calculations with randomized vector products enables us to formulate the problem in a neat matrix form, which increases the scalability of the introduced framework.

We optimize the parameters of the process using maximum likelihood estimation (MLE) approach in terms of negative log-likelihood, which is shown to be quite efficient, consistent and asymptotically unbiased for point processes [16]. Therefore, we define our evaluation metric in terms of negative log-likelihood per event, which directly expresses the fitting performance. In order to learn the parameters of a spatio-temporal Hawkes process,

Manuscript received March 7, 2020; revised May 15, 2020 and July 19, 2020; accepted August 18, 2020. Date of publication August 25, 2020; date of current version September 14, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Y. Xie. This work was supported in part by Tubitak Contract No: 117E153. *(Corresponding author: Fatih Ilhan.)*

The authors are with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey, and also with DataBoss A. S., Bilkent Cyberpark, Ankara 06800, Turkey (e-mail: filhan@ee.bilkent.edu.tr; kozat@ee.bilkent.edu.tr).

Digital Object Identifier 10.1109/TSP.2020.3019329

there exist several inference methods in the point process literature, most notably, expectation-maximization (EM) [11], [18], [19] algorithm and stochastic declustering [20]–[23]. Recently, gradient descent-based optimization methods has also been preferred in the context of temporal point processes [24]–[26]. Nevertheless, employing gradient descent in the spatio-temporal case is not as straightforward as in the temporal case. The difficulty lies within the structure of the likelihood function, which includes multi-dimensional integrals of kernel outputs. Hence, maximum likelihood estimation with gradient-based methods is not directly viable [16]. The conditional intensity function of a temporal point process is only defined along the temporal dimension, hence expressing likelihood objective in a differentiable manner, and applying gradient descent-based optimization is rather straightforward compared to the spatio-temporal case. To address this issue, we analytically derive the intractable terms in the likelihood function and their derivatives. We also employ reparameterization techniques and projected gradient-descent to handle the numerical constraints over the process parameters properly.

Even though there exists a considerable amount of prior art about spatio-temporal Hawkes processes, we, for the first time in the literature, utilize random Fourier features based kernel representation for spatio-temporal Hawkes processes. Our approach provides flexibility thanks to the introduced controllable randomization over spatial modeling. We also introduce a novel inference framework with well-organized matrix formulations and gradient descent-based optimization, which provides scalability. Furthermore, we investigate the fitting performance of the introduced method through an extensive set of experiments involving synthetic and real-life datasets. The results show that our method provides significant improvements compared to the EM algorithm and stochastic declustering, which are commonly favored in the point process literature [11], [18]–[20], [27], [28]. Finally, we perform event analysis over real-life datasets by interpreting the inferred parameters.

B. Prior Art and Comparisons

A significant amount of research has been conducted in signal processing, applied mathematics, and machine learning literatures to learn and apply spatio-temporal point processes [1], [4], [7], [11]. The approach of spatio-temporal modeling with point processes has been applied to various real-world scenarios such as seismological modeling of earthquakes and aftershocks [13], [27]–[29], criminological modeling of the dynamics of illegal incidents [7], [26], forecasting of disease outbreaks [16], network analysis [11], [18], [30]–[32], and so on. When carefully analyzed, the behavior of underlying systems can vary among different contexts. To this end, several forms of point processes have been proposed with different characteristics, such as Poisson process, Cox process, self-correcting processes [33], and self-exciting processes [14]. In this study, we consider spatio-temporal Hawkes process, which was first applied for earthquake prediction [13] and then successfully adapted to other applications such as crime analysis [7].

While modeling spatio-temporal Hawkes processes, there has been several proposals for the form of spatial kernel such as isotropic kernels [13], diffusion kernels [29], and Gaussian kernels [7], [11]. Even though the proposed forms have the common characteristic of having an inverse relation between the excitation level and distance from the event center, their behaviors are considerably different. On the contrary, even though we employ Gaussian kernel as well, we introduce randomization to the spatial modeling of the problem through random Fourier features based kernel representation. The introduced tunable randomization while modeling the spatial excitation enhances the performance in real-life scenarios, particularly when the spatial dynamics of the underlying system deviates from pure Gaussian behavior.

Random Fourier features have been used successfully to increase the scalability of kernel-based methods such as support vector machines (SVMs) [34]. Introducing randomization to the learning process has also been studied in machine learning literature [35]. From the neural network perspective, our representation can be interpreted as a perceptron layer with randomly initialized weights and sinusoidal activation function, where the weights are not being updated and they are sampled from a distribution related to the spectral distribution of the kernel [36]. This architecture is called as extreme learning machines (ELM) that have universal approximation capability as the number of nodes (embedding dimensions) goes to infinity [35]. Since we only have to approximate the kernel outputs of two-dimensional spatial vectors, low embedding dimensions suffice with negligible approximation errors [17]. This enables us to replace complicated pairwise kernel calculations with scalable matrix operations.

To increase the generalization power in point process models, machine learning based approaches have also been proposed in recent studies [24]–[26], [37]. In particular, recurrent neural networks (RNNs) and variants such as long short-term memory networks (LSTMs) are employed in the context of temporal point processes [25], [37]. However in the spatio-temporal domain, increased number of dimensions and sparsity may lead to unstable and difficult training of machine learning models [38]. In [37], authors employ LSTMs to model temporal Hawkes processes and uses the Monte-Carlo estimation of the intractable terms in the likelihood function and parameter gradients. However, relying on Monte-Carlo estimation is also problematic due to the increased space size after introducing spatial dimensions, which would require a large number of samples and result in degraded approximation performance [39]. Since we formulate the problem in a tractable form, our approach does not need any sampling based approximation.

In addition to the application and modeling perspectives, there is also an extensive literature about estimating the parameters of a spatio-temporal Hawkes process. Several inference methods were proposed for this problem. The most commonly used techniques involve MLE approach, which can be solved using expectation maximization (EM) as applied in various studies [11], [18], [19]. EM algorithm exhibits certain nice properties such as consistently increasing likelihood at each iteration, and naturally producing valid estimations for desired parameters without any numerical constraints [40]. However, it can suffer from instability due to bad initialization and slow convergence in regions, where the likelihood function is flat [41].

Another method, stochastic declustering has shown successful results particularly for earthquake modeling [20]–[23], [27], [28]. This is a non-parametric approach and relies on the branching structure of events, which assumes that the events can be clustered into two separate groups: background events and triggered events branching from the background. Bayesian inference methods have also been studied for self-exciting temporal point processes [31], [32], [42], [43]. Finally, gradient descent-based numerical optimization methods are used in the most recent studies [25], [26], [37]. In this study, we optimize the parameters through likelihood maximization with gradient descent-based algorithms since these methods are shown to be simple yet effective, particularly for neural networks [24], [44].

C. Contributions

Our main contributions are as follows:

1) As the first time in the literature, we apply random Fourier features based transformations to represent kernel operations in spatio-temporal Hawkes processes. This transformation increases the flexibility of our spatial modeling due to the introduced randomization, and can easily be controlled by tuning the number of embedding dimensions depending on the application.
2) We introduce a novel framework to formulate the problem in terms of scalable matrix operations by utilizing the vector products of transformed features instead of explicit pairwise kernel calculations.
3) We employ gradient descent based optimization to learn the parameters of the proposed model. To this end, we analytically obtain the intractable terms of the likelihood and properly handle the constraints over parameters by using reparameterization techniques and projected gradient descent.
4) We propose a simulation algorithm that follows thinning procedure to generate synthetic spatio-temporal Hawkes process realizations with multiple event types.
5) Through an extensive set of experiments over synthetic and real-life datasets, we demonstrate that our method brings significant improvements in terms of fitting performance with respect to the EM algorithm and stochastic declustering, which have been extensively favored in the point process literature [11], [16], [19], [20], [27], [28].
6) We demonstrate the practical applications of the proposed method by performing event analysis over real-life spatio-temporal event sequences through the interpretation of inferred excitation coefficients.

D. Organization

The remainder of the paper is organized as follows. We provide the form of the spatio-temporal Hawkes process and introduce the optimization problem in Section II. Then we provide the matrix formulations to express the likelihood function in a closed-form using random Fourier features in Section III-B. Then, we analytically obtain the derivatives of the likelihood with respect to process parameters and provide in Appendix A. In Section III-C, we give the gradient-based optimization algorithm for maximum likelihood estimation under parameter constraints. We analyze the fitting performance of the proposed method over simulated and real-life datasets and perform network analysis in Section IV. We conclude the paper in Section V with several remarks.

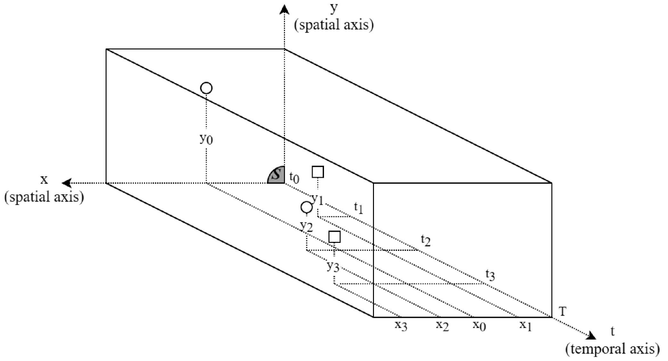


Fig. 1. An example of a spatio-temporal event sequence with four events. Each event is located inside the spatial domain $\mathcal{S}$, and distributed along the temporal axis. We use various shapes to represent different event types. Here, we have two types of events.

II. Problem Description

In this paper,[1] we study spatio-temporal event analysis with point processes. We observe an event sequence $\mathcal{E} = \{e_i\}_{i=1}^{N}$, and model it with spatio-temporal Hawkes processes. Here, N is the total number of events and $e_i = \{u_i, t_i, \boldsymbol{s_i}\}_{i=1}^{N}$ is the i^{th} event in the sequence with type $u_i \in \mathbb{N}$, time $t_i \in \mathcal{T}$ and location $\boldsymbol{s_i} = [x_i, y_i]^T \in \mathcal{S}$.[2] We visualize an example of a spatio-temporal event sequence in Fig. 1. Our goal is to infer the parameters of the process and perform analysis on the given event sequence through investigating the excitation between events in spatial and temporal domain. We model the spatial dynamics of $\mathcal{E}$ by expressing the kernels with random Fourier features-based kernel representations. The parameters of the process are denoted as $\boldsymbol{\theta}$, and we optimize them using MLE approach, in which the objective function is the log-likelihood of the given event sequence. Then, we solve the underlying optimization problem with gradient descent and properly handle numerical constraints using reparameterization techniques and projected gradient descent.

[1] All vectors are column vectors and denoted by boldface lower case letters. Matrices are denoted by boldface upper case letters. $\boldsymbol{x}^T$ and $\mathbf{X}^T$ are the corresponding transposes of $\boldsymbol{x}$ and $\mathbf{X}$. $\|\boldsymbol{x}\|$ is the ℓ^2-norm of $\boldsymbol{x}$. $\odot$ and $\oslash$ denotes the Hadamard product and division operations. $|\mathbf{X}|$ is the determinant of $\mathbf{X}$. For any vector $\boldsymbol{x}$, x_i is the i^{th} element of the vector. x_{ij} is the element that belongs to $\mathbf{X}$ at the i^{th} row and the j^{th} column. $\mathrm{sum}(\cdot)$ is the operation that sums the elements of a given vector or matrix. δ_{ij} is the Kronecker delta, which is equal to one if $i = j$ and zero otherwise.

[2] We define temporal space $\mathcal{T} \triangleq \{t \mid t \in [0, \infty)\}$, and consider spatial space $\mathcal{S}$ to be a rectangular subset of $\mathbb{R}^2$.

A. Temporal Point Processes

A temporal point process is a stochastic process that consists of realizations of subsequent events in discrete time $t_i \in \mathbb{R}$ with $i \in \mathbb{Z}$. We can interpret a temporal point process by specifying the distribution of the time distance between subsequent events (inter-event times). Let $f^*(t) = f(t|\mathcal{H}_t)$ be the conditional density function for the time of the next event given the time history of events $\mathcal{H}_t$. To express the past dependence in an evolutionary point process, conditional intensity function is defined as follows [14], [16],

$$\lambda^*(t) = \frac{f^*(t)}{1 - F^*(t)}, \tag{1}$$

where $F^*(t)$ is the cumulative density distribution of t such that $F^*(t) = 1 - \exp\left(-\int_{t_n}^{t} \lambda^*(t)\right)$ and t_n is the time of the last event before t. We can express the conditional density function $f^*(t)$ in terms of conditional intensity function $\lambda^*(t)$ using (1) as

$$f^*(t) = \lambda^*(t)e^{-\Lambda_{\lambda^*}(t)}, \quad \Lambda_{\lambda^*}(t) = \int_{t_n}^{t} \lambda^*(\tau)d\tau, \tag{2}$$

where t_n is the time of the last event before t. Here, the conditional intensity function can have many forms. As a simple example, in the case of a Poisson process, $\lambda^*(t) = \lambda(t) = \lambda$, i.e value of the conditional intensity function is constant through time.

B. Hawkes Processes

Unlike the Poisson Process, Hawkes process has an evolutionary nature, in which the events excite each other depending on their types and distance as expressed in the following form:

$$\lambda_u^*(t) = \mu_u + \sum_{j|t_j<t} k_{u_j u} g(t, t_j, u, u_j),$$

where μ_u denotes the background conditional intensity, $k_{u_j u}$ is the excitation of event type u_j over u for triggering conditional intensity and $g(t, t_j, u, u_j)$ is the output of the temporal triggering kernel evaluated at event times t and t_j for given event types u and u_j. This form enables us to model the point processes that show temporally clustered patterns.

C. Spatio-Temporal Hawkes Processes

In the spatio-temporal case, each event also has a spatial vector ($\boldsymbol{s}$) that describes its location. While expressing the conditional intensity function, we consider the following form in our problem:[3]

$$\lambda_u(t, \boldsymbol{s}) = \mu_u(\boldsymbol{s}) + \gamma_u(t, \boldsymbol{s}), \tag{3}$$

where $\mu_u(\boldsymbol{s})$ denotes the base conditional intensity for spatial vector $\boldsymbol{s}$ and event type u, and $\gamma_u(t, \boldsymbol{s})$ denotes the triggering conditional intensity for any time t, $\boldsymbol{s}$ and u. We can parametrize the base and triggering conditional intensities in (3) as follows,

$$\mu_u(\boldsymbol{s}) = \frac{1}{T}\sum_{j=1}^{N} k_{u_j u}^{(\mu)} g_2^{(\mu)}(\boldsymbol{s}, \boldsymbol{s_j}), \tag{4}$$

$$\gamma_u(t, \boldsymbol{s}) = \sum_{j|t_j<t} k_{u_j u}^{(\gamma)} g_1(t, t_j, u, u_j) g_2^{(\gamma)}(\boldsymbol{s}, \boldsymbol{s_j}), \tag{5}$$

where g_1 is the temporal kernel function and $g_2^{(\cdot)}$ is the spatial kernel function.[4] These functions can be expressed as

$$g_1(t, t_j, u, u_j) = w_{u_j u} e^{-w_{u_j u}(t - t_j)} \tag{6}$$

and

$$g_2^{(\cdot)}(\boldsymbol{s}, \boldsymbol{s_j}) = \frac{1}{2\pi}|\boldsymbol{\Sigma}^{(\cdot)}|^{-1/2} e^{-\frac{1}{2}(\boldsymbol{s}-\boldsymbol{s_j})^T \boldsymbol{\Sigma}^{(\cdot)^{-1}}(\boldsymbol{s}-\boldsymbol{s_j})}, \tag{7}$$

where T is the duration of the event sequence, N is the number of events, $\boldsymbol{\Sigma}^{(\cdot)}$ is the covariance matrix of the spatial Gaussian kernel, and $w_{u_j u} \geq 0$ is the decay rate of the intensity triggered by event type u_j over u.

The excitation values (k_{ij}) and weight decays (w_{ij}) are expressed in form of matrices $\mathbf{K}$ and $\mathbf{W}$, where $k_{ij}, w_{ij} > 0$. The multivariate normal distribution is said to be non-degenerate when the symmetric covariance matrix $\boldsymbol{\Sigma}^{(\cdot)}$ is positive definite. In this case, $g_2^{(\cdot)}(\boldsymbol{s}, \boldsymbol{s_j})$ will have an invertible covariance matrix and density.

It is still possible to use the form in (2) to express the conditional density function for the spatio-temporal case as

$$f_u(t, \boldsymbol{s}) = \lambda_u(t, \boldsymbol{s})e^{-\Lambda_\lambda(t)}, \tag{8}$$

where

$$\Lambda_\lambda(t) = \sum_{u'=1}^{U} \int_{t_n}^{t} \iint_{\boldsymbol{s'}\in\mathcal{S}} \lambda_{u'}(t,' \boldsymbol{s'}) d\boldsymbol{s'} dt'. \tag{9}$$

To estimate the optimum parameter set $\boldsymbol{\theta} = \{\mathbf{K}^{(\mu)}, \mathbf{K}^{(\gamma)}, \mathbf{W}, \boldsymbol{\Sigma}^{(\mu)}, \boldsymbol{\Sigma}^{(\gamma)}\}$, we follow maximum likelihood estimation approach. The negative log-likelihood over the real-life event sequence $\mathcal{E} = \{e_i\}_{i=1}^{N}$ is minimized, where N denotes the number of events. The objective is given below:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}, \tag{10}$$

where $\mathcal{L}$ is the negative log-likelihood and can be expressed as

$$\mathcal{L} = -\log\left(\prod_{i=1}^{N} f_{u_i}(t_i, \boldsymbol{s_i})\right) = -\sum_{i=1}^{N} \log \lambda_{u_i}(t_i, \boldsymbol{s_i}) + \sum_{i=1}^{N} \Lambda_\lambda(t_i), \tag{11}$$

where the second term involving $\Lambda_\lambda(t_i)$ can be interpreted as a regularizer, which prevents producing high intensity values over all space defined by $\mathcal{T}$ and $\mathcal{S}$.

We point out that certain parameters included in $\boldsymbol{\theta}$ are optimized indirectly through reparameterization to handle numerical

[3] Note that * sign, which denotes the conditionality on history will be dropped from now on for the sake of notational simplicity.

[4] For any scalar, vector, matrix or function, we denote the belonging to the intensity component $(\cdot)$ with power notation, e.g, $g_2^{(\mu)}$ is the spatial kernel parameterized for base intensity component.

constraints such as positivity of $\mathbf{K}^{(\cdot)}$ and $\mathbf{W}$, and unique properties of covariance matrices. Methods to handle these constraints during the optimization are explained in Sections III-C.

D. Random Fourier Features

Random Fourier features provide an efficient way to approximate the output of a shift-invariant continuous kernel $k(\boldsymbol{x}, \boldsymbol{y})$ with $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$ [17]. This technique embeds kernel inputs ($\boldsymbol{x}$ and $\boldsymbol{y}$) into a D-dimensional Euclidean inner product space using a transformation matrix $\mathbf{F} \in \mathbb{R}^{d \times D}$ and approximates $k(\boldsymbol{x}, \boldsymbol{y})$ through the inner product of embedded vectors. Although it is widely used to scale up kernel based methods such as SVM for large datasets, [45] we use it to increase the spatial flexibility and replace complex kernel calculations with straightforward matrix multiplications.

Random Fourier feature-based kernel representation relies on Bochner's Theorem, which states that any bounded, continuous and shift-invariant kernel is a Fourier transform of a bounded non-negative measure [46]. Assuming $p(\cdot)$ is the density function of the spectral measure, the corresponding shift-invariant kernel can be written as

$$k(\boldsymbol{x}, \boldsymbol{y}) = \int_{\mathcal{R}_d} p(\boldsymbol{w}) e^{j\boldsymbol{w}^T(\boldsymbol{x}-\boldsymbol{y})} d\boldsymbol{w} = E_{\boldsymbol{w}}[\zeta_{\boldsymbol{w}}(\boldsymbol{x})\zeta_{\boldsymbol{w}}(\boldsymbol{y})^*],$$

where $\zeta_{\boldsymbol{w}}(\boldsymbol{x}) = e^{j\boldsymbol{w}^T\boldsymbol{x}}$, and c^* denotes the complex conjugate of $c \in \mathcal{C}$. Finally, this expression is approximated by its Monte-Carlo estimate as follows,

$$\tilde{k}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{D}\sum_{i=1}^{D} z_i(\boldsymbol{x}) z_i(\boldsymbol{y}) = \boldsymbol{z}^T(\boldsymbol{x})\boldsymbol{z}(\boldsymbol{y}), \tag{12}$$

where $z_i(\boldsymbol{x}) = \sqrt{2}\cos(\boldsymbol{x}^T\boldsymbol{w_i} + b_i)$ with $\boldsymbol{w_i} \in \mathbb{R}^{d \times 1}$ sampled from $p(\boldsymbol{w})$ and $b_i \sim \mathrm{U}(0, 2\pi)$.

III. Spatio-Temporal Hawkes Process With Randomized Kernel Representation

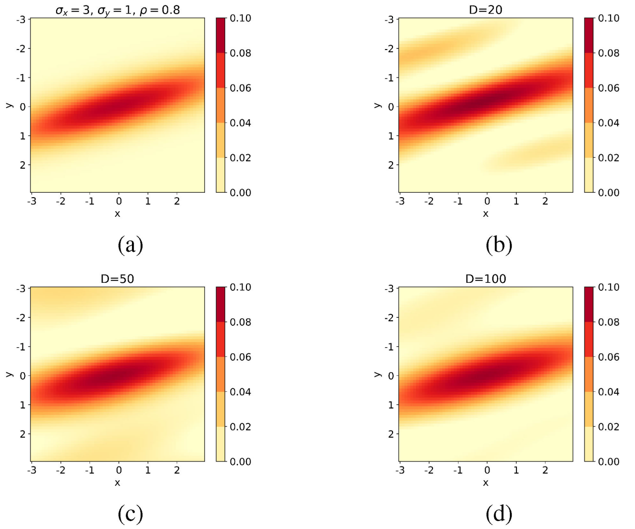
In this section, we describe our method to express the spatial kernels given in (4) and (5) with Random Fourier features using (12), and obtain a neat matrix formulation for the objective function given in (11). Then, we provide derivative calculations for gradient descent and describe the optimization procedure.

A. Random Fourier Features for Kernel Representation

We start with expressing the spatial Gaussian kernel functions of the base and triggering intensity components in (4) and (5) using random Fourier features. For given two locations $\boldsymbol{s_i} = [x_i, y_i]^T$ and $\boldsymbol{s_j} = [x_j, y_j]^T$, the result of the Gaussian kernel output in (7) can be approximated with the following D dimensional random Fourier approximation [17]s

$$g_2(\boldsymbol{s_i}, \boldsymbol{s_j}) \approx \frac{1}{2\pi}|\boldsymbol{\Sigma}|^{-1/2}\boldsymbol{z_i}^T\boldsymbol{z_j}, \tag{13}$$

where $\boldsymbol{z_i}^T = \sqrt{\frac{2}{D}}\cos(\boldsymbol{s_i}^T\mathbf{F} + \boldsymbol{b}^T)$ with $\mathbf{F} \in \mathbb{R}^{2 \times D}$, $\boldsymbol{f_d} \sim \mathcal{N}(0, \tilde{\boldsymbol{\Sigma}})$ and b_d is sampled uniformly from $[0, 2\pi]$, and $\tilde{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}^{-1}$.



(a) (b) (c) (d)

Fig. 2. (a) Gaussian kernel with $\sigma_x = 3$, $\sigma_y = 1$, and $\rho = 0.8$, (b)-(c)-(d) Approximated kernels with 20, 50 and 100-dimensional random Fourier features.

To analyze the behavior of the random Fourier features given in (13), we construct a Gaussian kernel with $\sigma_x = 3$, $\sigma_y = 1$ and $\rho = 0.8$, and perform three approximations with various embedding dimensions ($D = 20, 50, 100$). We visualize the results in Fig. 2. As the number of dimensions in random Fourier features increases, the approximation becomes more accurate. In the cases when D is small as in Fig. 2(b), some randomly repeating artifacts are visible around the kernel.

Since $\boldsymbol{\Sigma}$ is a positive definite and symmetric matrix as mentioned in the previous section, $\tilde{\boldsymbol{\Sigma}}$ is also positive-definite and symmetric. Therefore, we can decompose $\tilde{\boldsymbol{\Sigma}}$ using the Cholesky decomposition. and express as $\tilde{\boldsymbol{\Sigma}} = \tilde{\mathbf{C}}\tilde{\mathbf{C}}^T$, where $\tilde{\mathbf{C}}$ is a unique, invertible, lower triangular 2×2 matrix with real, and positive diagonal entries. Using this decomposition, we obtain the following form for vector embedding:

$$\boldsymbol{z_i}^T = \sqrt{\frac{2}{D}}\cos(\boldsymbol{s_i}^T\tilde{\mathbf{C}}\mathbf{U} + \boldsymbol{b}^T), \tag{14}$$

where $\mathbf{U} \in \mathbb{R}^{2 \times D}$, and $\boldsymbol{u_d} \sim \mathcal{N}(0, \mathbf{I})$.

We emphasize that $\tilde{\mathbf{C}}$ introduces certain numerical constraints because of being lower triangular and having positive, real diagonal entries that should be considered during optimization. To handle this issue, we use eigendecomposition of the covariance matrix $\boldsymbol{\Sigma}$ to express $\tilde{\mathbf{C}}$ in a simpler form still with constraints but more straightforward to handle:

$$\boldsymbol{\Sigma}^{-1} = (\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^T)^{-1} = (\mathbf{V}\boldsymbol{\Lambda}^{-1/2})(\boldsymbol{\Lambda}^{-1/2}\mathbf{V}^T), \tag{15}$$

where $\mathbf{V}\boldsymbol{\Lambda}^{-1/2} = \tilde{\mathbf{C}}$.

Now, we have two components: the eigenvector matrix $\mathbf{V} \in \mathbb{R}^{2 \times 2}$ with orthonormality, and the diagonal matrix of eigenvalues $\boldsymbol{\Lambda} \in \mathbb{R}^{2 \times 2}$ with positivity constraints. These components can be interpreted as the descriptors of the direction and magnitude of excitation caused by an event. As a result, we obtain the following final form for the vector embedding:

$$\boldsymbol{z_i}^T = \sqrt{\frac{2}{D}} \cos\left(\boldsymbol{s_i}^T \mathbf{V} \boldsymbol{\Lambda}^{\mathbf{-1/2}} \mathbf{U} + \boldsymbol{b}^T\right). \tag{16}$$

In addition, we can express $|\boldsymbol{\Sigma}|^{-1/2}$ in (13) in terms of the the diagonal elements of $\boldsymbol{\Lambda}$ such that $|\boldsymbol{\Sigma}|^{-1/2} = \frac{1}{\sqrt{\ell_1 \ell_2}}$, where $\boldsymbol{\ell} = [\ell_1, \ell_2]^T \triangleq [\Lambda_{11}, \Lambda_{22}]^T$ is the vector that consists of the diagonal elements of $\boldsymbol{\Lambda}$. Since any term including covariance matrices $\boldsymbol{\Sigma}^{(\mu)}$ and $\boldsymbol{\Sigma}^{(\gamma)}$ or their corresponding Cholevsky components can be expressed using $\mathbf{V}^{(\mu)}$, $\mathbf{V}^{(\gamma)}$, $\boldsymbol{\ell}^{(\mu)}$, and $\boldsymbol{\ell}^{(\gamma)}$, we update the notation given for the parameter set as $\boldsymbol{\theta} = \{\mathbf{K}^{(\mu)}, \mathbf{K}^{(\gamma)}, \mathbf{W}, \mathbf{V}^{(\mu)}, \mathbf{V}^{(\gamma)}, \boldsymbol{\ell}^{(\mu)}, \boldsymbol{\ell}^{(\gamma)}\}$.

B. Matrix Formulations

After representing the spatial kernel with the random Fourier feature-based approximation, we formulate the problem in a well-organized matrix form. First, we define the following matrices:[5]

$$\mathbf{Z}^{(\cdot)}_{J(t)} \triangleq \begin{bmatrix} & \vdots & \\ - & \boldsymbol{z}_j^{(\cdot)^T} & - \\ & \vdots & \end{bmatrix}_{N' \times D}, \tag{17}$$

$$\boldsymbol{d}_{J(t)} \triangleq \begin{bmatrix} \vdots \\ w_{u_j u_i} \exp\left(-w_{u_j u_i}(t - t_j)\right) \\ \vdots \end{bmatrix}_{N' \times 1}, \tag{18}$$

$$\boldsymbol{Y}_{J(t)} \triangleq \begin{bmatrix} & \vdots & \\ - & \boldsymbol{y}_j^T & - \\ & \vdots & \end{bmatrix}_{N' \times U}, \tag{19}$$

$$\mathbf{N}^{(\mu)}(t) \triangleq \frac{1}{2\pi} |\boldsymbol{\Sigma}|^{-1/2} \mathbf{Z}^{(\mu)^T}_{J(t)} \boldsymbol{Y}_{J(t)}, \tag{20}$$

$$\mathbf{N}^{(\gamma)}(t) \triangleq \frac{1}{2\pi} |\boldsymbol{\Sigma}|^{-1/2} \mathbf{Z}^{(\gamma)^T}_{J(t)} \operatorname{diag}(\boldsymbol{d}_{J(t)}) \boldsymbol{Y}_{J(t)}, \tag{21}$$

where $\boldsymbol{y}_j^T$ is the one-hot vector form of an event type for the j^{th} event.

Using (4), (5), (13) and (16), base and triggering conditional intensity function values for the i^{th} event can be factorized as

$$\mu_{u_i}(\boldsymbol{s_i}) = \frac{1}{T} \boldsymbol{z_i}^{(\mu)^T} \mathbf{N}^{(\mu)}(T) \boldsymbol{k}^{(\mu)}_{u_i}, \tag{22}$$

$$\gamma_{u_i}(t_i, \boldsymbol{s_i}) = \boldsymbol{z_i}^{(\gamma)^T} \mathbf{N}^{(\gamma)}(t_i) \boldsymbol{k}^{(\gamma)}_{u_i}, \tag{23}$$

where $\boldsymbol{k}^{(\cdot)}_{u_i}$ is the u_i^{th} column of $\mathbf{K}^{(\cdot)}$, which contains the effects of other event types over the event type of the i^{th} event. Finally, using (20) and (21), the conditional intensity values for $\mathcal{E}$ given in (11) can be expressed in the following matrix form:

$$\mathbf{A} \triangleq \begin{bmatrix} & \vdots & \\ - & \lambda(t_i, \boldsymbol{s_i}) & - \\ & \vdots & \end{bmatrix} = \mathbf{Q}^{(\mu)} \mathbf{K}^{(\mu)} + \mathbf{Q}^{(\gamma)} \mathbf{K}^{(\gamma)}, \tag{24}$$

where

$$\mathbf{Q}^{(\mu)} \triangleq \begin{bmatrix} & \vdots & \\ - & \frac{1}{T} \boldsymbol{z_i}^{(\mu)^T} \mathbf{N}^{(\mu)}(T) & - \\ & \vdots & \end{bmatrix}_{N \times U}$$

contains the relation between the i^{th} event and other events for base intensity, and

$$\mathbf{Q}^{(\gamma)} \triangleq \begin{bmatrix} & \vdots & \\ - & \boldsymbol{z_i}^{(\gamma)^T} \mathbf{N}^{(\gamma)}(t_i) & - \\ & \vdots & \end{bmatrix}_{N \times U}$$

contains the relation between the i^{th} event and past events for triggering intensity at each row.

Once obtaining the matrix-form expression for the conditional intensity in (24), we analytically derive the integral output to obtain the closed-form expression for the second term in (11) as

$$\begin{aligned}
\Lambda_\lambda(t_i) &= \sum_{u'=1}^{U} \int_{t_{i-1}}^{t_i} \iint_{\boldsymbol{s'} \in \mathcal{S}} \lambda_{u'}(t,' \boldsymbol{s'}) d\boldsymbol{s'} dt' \\
&\approx \sum_{u'=1}^{U} \int_{t_{i-1}}^{t_i} \iint_{\boldsymbol{s'} \in \mathbb{R}^2} \mu_{u'}(\boldsymbol{s'}) d\boldsymbol{s'} dt' \\
&\quad + \sum_{u'=1}^{U} \int_{t_{i-1}}^{t_i} \iint_{\boldsymbol{s'} \in \mathbb{R}^2} \gamma_{u'}(t,' \boldsymbol{s'}) d\boldsymbol{s'} dt' \\
&\approx \sum_{u'=1}^{U} \int_{t_{i-1}}^{t_i} \iint_{\boldsymbol{s'} \in \mathbb{R}^2} \frac{1}{T} \sum_{j=1}^{N} k^{(\mu)}_{u_j u'} g^{(\mu)}_2(\boldsymbol{s'}, \boldsymbol{s_j}) d\boldsymbol{s'} dt' \\
&\quad + \sum_{u'=1}^{U} \int_{t_{i-1}}^{t_i} \iint_{\boldsymbol{s'} \in \mathbb{R}^2} \sum_{j | t_j < t'} k^{(\gamma)}_{u_j u'} g_1(t', t_j, u', u_j) \\
&\quad \times g^{(\gamma)}_2(\boldsymbol{s'}, \boldsymbol{s_j}) d\boldsymbol{s'} dt' \\
&\approx \frac{t_i - t_{i-1}}{T} \sum_{j=1}^{N} \sum_{u'=1}^{U} k^{(\mu)}_{u_j u'} \\
&\quad + \sum_{j | t_j < t_i} \sum_{u'=1}^{U} k^{(\gamma)}_{u_j u'} \left(e^{-w_{u_j u'}(t_{i-1} - t_j)} - e^{-w_{u_j u'}(t_i - t_j)}\right),
\end{aligned} \tag{25}$$

where we approximate $\mathcal{S}$ with $\mathbb{R}^2$ since the boundary effects will have a negligible effect over the integral value. Then, the

[5] We use $J(t) = \{j | t_j < t\}$ to notate the rows that belong to the events occurred before t.

summation of $\Lambda_\lambda(t_i)$ for consecutive events is expressed as

$$\begin{aligned}\sum_{i=n-k}^{n}\Lambda(t_i) &= \frac{t_n - t_{n-k-1}}{T}\sum_{j=1}^{N}\sum_{u'=1}^{U} k^{(\mu)}_{u_j u'} \\ &\quad - \sum_{j|t_j<t_n}\sum_{u'=1}^{U} k^{(\gamma)}_{u_j u'}(e^{-w_{u_j u'}(t_n - t_j)}) \\ &\quad + \sum_{j|t_j<t_{n-k-1}}\sum_{u'=1}^{U} k^{(\gamma)}_{u_j u'}(e^{-w_{u_j u'}(t_{n-k-1} - t_j)}) \\ &\quad + \sum_{j|t_{n-k-1}\leq t_j<t_n}\sum_{u'=1}^{U} k^{(\gamma)}_{u_j u'} \qquad (26)\end{aligned}$$

for $0 \leq k < i$ and $t_0 = 0$. Here, we utilize the relation between consecutive terms, which cancels out most of the intermediate outputs. Inserting $n = N$ and $k = N - 1$ into (26) yields the following:

$$R \triangleq \sum_{i=1}^{N}\Lambda(t_i) = \sum_{j=1}^{N}\sum_{u'=1}^{U} k^{(\mu)}_{u_j u'} + k^{(\gamma)}_{u_j u'}(1 - e^{-w_{u_j u'}(T - t_j)}), \qquad (27)$$

where R is defined as the second term in (11), and has a suppressing effect over excitation matrices.

Finally, using (11), (24), and (27), we can express the negative log-likelihood as

$$\mathcal{L} = -\mathrm{sum}(\log(\mathbf{A}) \odot \mathbf{Y}) + R. \qquad (28)$$

In order to minimize the negative log-likelihood expressed in (28), we employ gradient descent through the back propagation of derivatives $\frac{\partial\mathcal{L}}{\partial\boldsymbol{\theta}} = \{\frac{\partial\mathcal{L}}{\partial\mathbf{K}^{(\mu)}}, \frac{\partial\mathcal{L}}{\partial\mathbf{K}^{(\gamma)}}, \frac{\partial\mathcal{L}}{\partial\mathbf{W}}, \frac{\partial\mathcal{L}}{\partial\mathbf{V}^{(\mu)}}, \frac{\partial\mathcal{L}}{\partial\mathbf{V}^{(\gamma)}}, \frac{\partial\mathcal{L}}{\partial\boldsymbol{l}^{(\mu)}}, \frac{\partial\mathcal{L}}{\partial\boldsymbol{l}^{(\gamma)}}\}$. We provide the equations for these gradients in Appendix A.

C. Optimization Algorithm

Here, we detail the optimization procedure to minimize the negative log-likelihood $\mathcal{L}$ expressed in (28). We adapt mini-batch gradient descent into our problem with a slightly modified batch generation procedure as explained in Algorithm 1. We also follow a training procedure with early stopping that stops the iterations if the model does not improve during k consecutive steps in terms of negative log-likelihood.

Algorithm 1: Mini-Batch Gradient Descent With Random Fourier Features (RFF-GD).

```
Require: $\boldsymbol{\theta}$ (Initial parameter set), $\mathcal{E}_{train}$ (Event sequence for training), $\mathcal{E}_{val}$ (Event sequence for validation), $b$ (batch size), $\eta$ (learning rate), max_epoch (number of maximum epochs) and $\pi = False$ (early stopping flag)
  while epoch $<$ max_epoch do
    step $\leftarrow 0$
    while step $< N_{train}/b$ do
      Sample $i_s$ uniformly from $\{1, 2, \ldots N_{train} - b\}$.
      $i_e \leftarrow i_s + b$
      $X \leftarrow \{e_{train_i}\}_{i=i_s}^{i_e}$
      for all $\theta_k \in \boldsymbol{\theta}$ do
        Update $\theta_k$ using (34)-(42).
      end for
      step $\leftarrow$ step $+ 1$
    end while
    Calculate $\mathcal{L}$ over $\mathcal{E}_{val}$ with (11), and update $\pi$ based on early stopping criteria.
    if $\pi$ then
      return $\boldsymbol{\theta}$
    end if
    epoch $\leftarrow$ epoch $+ 1$
  end while
  return $\boldsymbol{\theta}$
```

As mentioned before, certain parameters in $\boldsymbol{\theta}$ have constraints. The elements of the excitation matrices $\mathbf{K}^{(\mu)}$ and $\mathbf{K}^{(\gamma)}$, the decay matrix $\mathbf{W}$, and the eigenvalue vectors of covariance matrices, $\boldsymbol{\ell}^{(\mu)}$ and $\boldsymbol{\ell}^{(\gamma)}$ have to be positive. To satisfy these conditions, we simply introduce the following intermediate variables and perform gradient descent over the unconstrained parameters $\tilde{\mathbf{K}}^{(\cdot)}$, $\tilde{\mathbf{W}}$ and $\tilde{\boldsymbol{\ell}}^{(\cdot)}$:

$$\mathbf{K}^{(\cdot)} = \phi(\tilde{\mathbf{K}}^{(\cdot)}) = \frac{1}{s}\log(1 + e^{s\tilde{\mathbf{K}}^{(\cdot)}}),$$

$$\mathbf{W} = \phi(\tilde{\mathbf{W}}) = \frac{1}{s}\log(1 + e^{s\tilde{\mathbf{W}}}),$$

$$\boldsymbol{\ell}^{(\cdot)} = \phi(\tilde{\boldsymbol{\ell}}^{(\cdot)}) = \frac{1}{s}\log(1 + e^{s\tilde{\boldsymbol{\ell}}^{(\cdot)}}),$$

where ϕ is the soft-plus function parametrized by s. Soft-plus function provides a differentiable and smooth approximation of rectified linear unit function ($\sigma_{\mathrm{ReLU}}(x) = \max(0, x)$) such that as $s \to \infty$, $\phi \to \sigma_{\mathrm{ReLU}}$ [47].

Other constrained parameters are the eigenvector matrices $\mathbf{V}^{(\mu)}$ and $\mathbf{V}^{(\gamma)}$, which have to be orthonormal due to the eigendecomposition in (15). We employ projected gradient descent to meet this limitation by using the following update rule:

$$\mathbf{V}^{(\cdot)}_{t+1} = \Pi_\chi\left(\mathbf{V}^{(\cdot)}_t - \eta\frac{\partial\mathcal{L}}{\partial\mathbf{V}^{(\cdot)}_t}\right), \ \forall t \geq 1,$$

where $\chi = \{\mathbf{X} \mid \mathbf{X} \in \mathbb{R}^2, \mathbf{X}^T\mathbf{X} = \mathbf{I}\}$ is the convex set of orthonormal matrices. Here, Π_χ projects the updated parameter to χ through solving the following minimization problem known as orthonogal Procrustes problem [48]:

$$\begin{aligned}\Pi_\chi(\tilde{\mathbf{X}}) &= \arg\min_{\mathbf{X}}(||\mathbf{X} - \tilde{\mathbf{X}}||_F) \ \text{ subject to } \ \mathbf{X} \in \chi \\ &= \mathbf{U}\mathbf{V}^T, \qquad (29)\end{aligned}$$

where $||\cdot||_F$ denotes the Frobenius norm, and $\tilde{\mathbf{X}} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T$.

IV. Experiments

In this section, we report the results of our method in terms of fitting performance on synthetic and real-life datasets. We generate three synthetic datasets to analyze the behavior of the proposed approach in a controlled manner. Then, we demonstrate the performance of our method in two real-life datasets and compare it with the EM algorithm [11] and stochastic declustering [20]. We also analyze the effect of the randomized feature space size on our performance. Finally, we perform event analysis through the interpretation of the inferred process parameters.

A. Synthetic Dataset Experiments

We first introduce a thinning-based algorithm to simulate synthetic event sequences according to given process parameters. Then, we evaluate two simple baseline approaches in addition to our method over three different simulations.

1) Spatio-Temporal Thinning Algorithm for Simulations: In order to simulate a spatio-temporal Hawkes process, we use the thinning algorithm [49], which applies rejection sampling over pre-sampled points. Unlike the extension of the thinning algorithm for spatio-temporal case in [50], we have multiple event types. The details are given in Algorithm 2.

To apply rejection sampling, we need an upper bound for the conditional intensity function,

$$\bar{\lambda} \triangleq \max\left(\sum_{u=1}^{U} \lambda_u(t,' \boldsymbol{s'})\right) \text{ for } t' \in [t, +\infty) \text{ and } \boldsymbol{s'} \in \mathcal{S}$$

such that $\lambda(t,' \boldsymbol{s'}) < \bar{\lambda}$ for all $t \geq t'$ and $\boldsymbol{s'} \in \mathcal{S}$. Since the conditional intensity decreases in time exponentially, upper bound will take place at time t, so we can express $\bar{\lambda}$ as

$$\bar{\lambda} = \sum_{u=1}^{U} \max(\mu_u(\boldsymbol{s'}) + \sum_{j|t_j<t} k_{u_j u}^{(\gamma)} g_1(t, t_j, u, u_j) g_2^{(\gamma)}(\boldsymbol{s'}, \boldsymbol{s_j})) \tag{30}$$

for $\boldsymbol{s'} \in \mathcal{S}$. We perform calculations for densely sampled spatial points over $\mathcal{S}$ at time t, and take the maximum value due to the non-monotonous structure of the conditional intensity function over the spatial domain. Moreover, older events will have significantly less effect over the total sum in (30) due to the exponential decay kernel. Thus, to make the simulation process computationally efficient, we ignore the triggering effects of the events that occurred before a particular temporal offset ($\tau = 100$). We observe no difference in the simulation with this modification and the algorithm is robust to the selection of this parameter.

To generate the type of the thinned event in Algorithm 2, we apply the thinning procedure over the total conditional intensity and then draw the event type stochastically from the generated $p(u)$ for the generated spatio-temporal point. Instead of running rejection sampling for each event type separately, this procedure provides an efficient and convenient way to generate spatio-temporal Hawkes process with multiple event types.

We simulate realizations with $T = 100000$ and $\mathcal{S} = [[-1, 1], [-1, 1]]$. Each simulation is set to different parameters to analyze the behavior in different cases. Table I shows the parameter sets used for simulations.

Algorithm 2: Thinning Algorithm for Spatio-Temporal Hawkes Process Simulation.

```
Require: $\lambda$ (Conditional intensity function), $\mathcal{T}$ (Temporal
  space) and $\mathcal{S}$ (Spatial space), $t = 0$, $i = 1$, $\mathcal{E} = \{\}$
  while True do
    Estimate $\bar{\lambda} = \max(\sum_{u=1}^{U} \lambda_u(t,' \boldsymbol{s'}))$ for $t' \in [t, +\infty)$
      and $\boldsymbol{s'} \in \mathcal{S}$ by (30).
    Draw $q \sim \mathcal{U}(0, 1)$
    $\Delta t \Leftarrow -\log(q)/\bar{\lambda}$
    $t \Leftarrow t + \Delta t$
    if $t > T$ then
      return $\mathcal{E}$
    end if
    Draw $\boldsymbol{s} \sim \mathcal{U}(\mathcal{S})$, $v \sim \mathcal{U}(0, 1)$.
    Calculate $\lambda(t, \boldsymbol{s}) = \sum_{u=1}^{U} \lambda_u(t, \boldsymbol{s})$.
    if $\lambda(t, \boldsymbol{s}) > v\bar{\lambda}$ then
      Draw $u \sim p(u)$, where $p(u) = \frac{e^{-\lambda_u}}{\sum_{u'=1}^{U} e^{-\lambda_{u'}}}$
      $e_i \Leftarrow [t, \boldsymbol{s}, u]$
      $\mathcal{E} \Leftarrow \mathcal{E} \cup e_i$
      $i \Leftarrow i + 1$
    end if
  end while
```

TABLE I
SIMULATION CONFIGURATIONS. $\mathbf{K}^{(\mu)}$ AND $\mathbf{K}^{(\gamma)}$ ARE THE EXCITATION MATRICES FOR THE BASE AND TRIGGERING INTENSITIES, $\mathbf{\Sigma}^{(\mu)}$ AND $\mathbf{\Sigma}^{(\gamma)}$ ARE THE COVARIANCE MATRICES OF THE SPATIAL KERNELS FOR THE BASE AND TRIGGERING INTENSITIES, AND $\mathbf{W}$ IS THE WEIGHT DECAY MATRIX OF THE TEMPORAL KERNEL

ID	$\mathbf{K}^{(\mu)}$	$\mathbf{K}^{(\gamma)}$	$\mathbf{W}$	$\mathbf{\Sigma}^{(\mu)}$	$\mathbf{\Sigma}^{(\gamma)}$
1	$\begin{bmatrix} 0.01 \end{bmatrix}$	$\begin{bmatrix} 0 \end{bmatrix}$	-	$\begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$	-
2	$\begin{bmatrix} 0.01 & 0.005 \\ 0.01 & 0.02 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	-	$\begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$	-
3	$\begin{bmatrix} 0.01 & 0.005 \\ 0.01 & 0.02 \end{bmatrix}$	$\begin{bmatrix} 1 & 0.5 \\ 1 & 2 \end{bmatrix}$	$\begin{bmatrix} 2 & 1 \\ 1 & 4 \end{bmatrix}$	$\begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$	$\begin{bmatrix} 0.04 & 0 \\ 0 & 0.04 \end{bmatrix}$

2) Synthetic Dataset Performance: We also evaluate two baseline processes with more basic forms compared to the spatio-temporal Hawkes process to analyze the behavior of the proposed framework under different scenarios. First, we consider the Poisson process, where each event type has a constant intensity (λ_u) over the spatio-temporal space:

$$\lambda_u(t, \boldsymbol{s}) = \mu_u, \tag{31}$$

where μ_u is the base intensity for the u^{th} event type.

Second, we allow the conditional intensity to be locally variant, but temporally constant by setting $\mathbf{K}^{(\gamma)}$ to be a zero matrix. This can be interpreted as a spatially inhomogeneous Poisson process. For this baseline, the conditional intensity has the following form:

$$\lambda_u(t, \boldsymbol{s}) = \mu_u(\boldsymbol{s}) = \frac{1}{T} \sum_{j=1}^{N} k_{u_j u}^{(\mu)} g_2^{(\mu)}(\boldsymbol{s}, \boldsymbol{s_j}). \tag{32}$$

For all experiments, we divide each event sequence into training (80%) and test (20%) sets. We use 10% of the training set for the hyperparameter search and early stopping. We obtain maximum likelihood estimates of the process parameters using Algorithm

TABLE II
TRAINING PERFORMANCE OF OUR ALGORITHM WITH POISSON (31), SPATIAL POISSON (32) AND SPATIO-TEMPORAL HAWKES (3) PROCESS MODELING ON SYNTHETIC DATASETS. SYNTHETIC DATA ARE SIMULATED WITH THE PARAMETER CONFIGURATIONS GIVEN IN TABLE I (p: NUMBER OF PARAMETERS, $\overline{\mathcal{L}}$: NEGATIVE LOG-LIKELIHOOD PER EVENT)

Model	Simulation ID								
	1			2			3		
	p	$\overline{\mathcal{L}}$	AIC	p	$\overline{\mathcal{L}}$	AIC	p	$\overline{\mathcal{L}}$	AIC
Poisson	1	-1.51	-800.94	2	-0.04	-175.36	2	1.39	4927.38
Spatial Poisson	7	-1.44	-1705.36	10	-0.77	-3432.68	10	0.13	480.46
ST-Hawkes	15	-1.57	-1844.58	24	-0.78	-3449.52	24	-2.39	-8417.38

1 and report the negative log-likelihoods on the training and test sets. We also investigate the Akaike's Information Criterion (AIC) [51], which is also shown to be a consistent measure while evaluating point process models and preferred in numerous studies in the point process literature including [42], [52], [53]. AIC is defined as follows [51]:

$$\mathrm{AIC} = 2\mathcal{L} + 2\,k \tag{33}$$

where $\mathcal{L}$ is the negative log-likelihood and k is the number of parameters. Although these criteria are maximum likelihood driven and tend to choose the model which fits to the data best, they also penalize the number of parameters to address complexity. Since spatio-temporal Hawkes modeling have more parameters as provided in Table II, AIC penalizes it more heavily. Both measures indicate better performance at lower values.

The results for all model-simulation pairs are shown in Table II. All experiments are repeated 10 times. We highlight that synthetic event sequences are scaled temporally before training to prevent numerical instability issues related to very low/high temporal space size, and report comparable results among all simulations. We shrink the event times such that the average temporal distance between consecutive events becomes 1 unit. We also normalize the resulting negative log-likelihood by dividing it by the number of events, and consider the negative log-likelihood per event to provide comparability across datasets.

As seen in Table II, all models perform similarly in the first simulation since the conditional intensity function is spatio-temporally homogeneous. In the second simulation, the simple Poisson process performs worse because the spatial triggering effect is not included in its modeling. In the third simulation, due to the introduced temporal excitation, spatio-temporal (ST) Hawkes performs significantly better than others thanks to its capability to express spatial and temporal inhomogeneity in the conditional intensity function. Hence, we conclude that our algorithm performs consistent with different modeling choices. In Table III, we provide the negative log-likelihood per event values obtained on the test set. The results demonstrate the generalization capability of our method since there is no considerable gap between training and test performances. We also provide the recovered intensity function parameters for the synthetic data experiment, which simulates a spatio-temporal Hawkes process, in Table IV.

TABLE III
TEST PERFORMANCE OF OUR ALGORITHM WITH POISSON (31), SPATIAL POISSON (32) AND SPATIO-TEMPORAL HAWKES (3) PROCESS MODELING ON SYNTHETIC DATASETS IN TERMS OF NEGATIVE LOG-LIKELIHOOD PER EVENT

Model	Simulation ID		
	1	2	3
Poisson	-1.43	0.09	1.38
Spatial Poisson	-1.51	-0.59	0.32
ST-Hawkes	-1.59	-0.61	-2.18

TABLE IV
ESTIMATED PARAMETERS FOR THE 3 RD SIMULATION

$\mathbf{K}^{(\mu)}$	$\mathbf{K}^{(\gamma)}$	$\mathbf{W}$	$\mathbf{\Sigma}^{(\mu)}$	$\mathbf{\Sigma}^{(\gamma)}$
$\begin{bmatrix} 0.018 & 0.003 \\ 0.011 & 0.038 \end{bmatrix}$	$\begin{bmatrix} 1.111 & 0.44 \\ 0.983 & 1.976 \end{bmatrix}$	$\begin{bmatrix} 1.78 & 0.943 \\ 0.9 & 3.96 \end{bmatrix}$	$\begin{bmatrix} 0.008 & 0.003 \\ 0.003 & 0.011 \end{bmatrix}$	$\begin{bmatrix} 0.039 & 0.001 \\ 0.001 & 0.038 \end{bmatrix}$

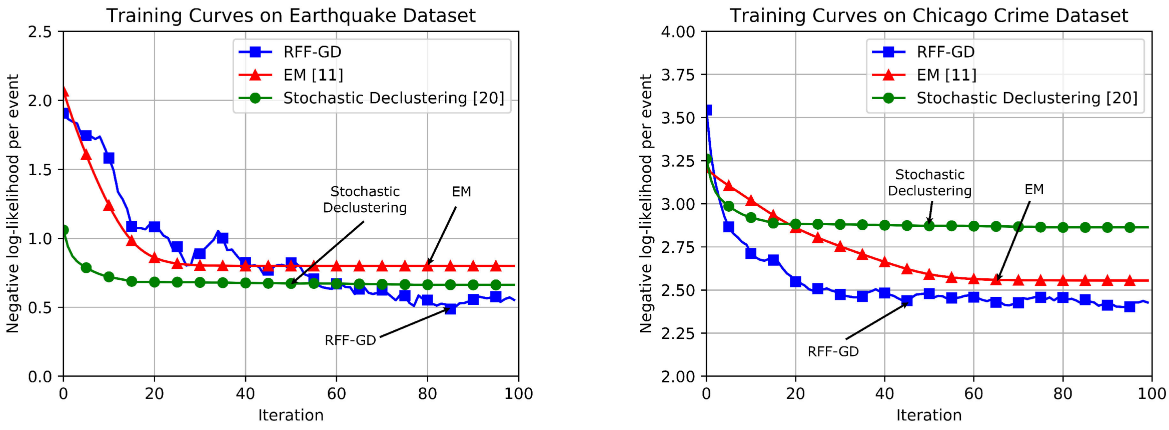
B. Real-life Dataset Experiments

To investigate the fitting performance of our method in real-life datasets, we investigate the negative log-likelihood per event values and AIC as we have done in synthetic data. After learning the process parameters, we perform event analysis by examining the interactions between different event types in terms of excitation relations and spatio-temporal effects. We also investigate the effect of the number of dimensions in the randomized feature space. To this end, we have chosen the following two datasets. These datasets have been studied in the context of point processes, with applications on spatio-temporal prediction, and hotspot analysis [7], [13]. They both exhibit certain characteristics such as having spatiotemporally clustered structures, which makes their modeling by spatio-temporal Hawkes processes plausible.

1) Datasets:

a) Chicago crime dataset: Chicago Crime Dataset includes the reported incidents in the City of Chicago from 2001, and is still being weekly updated by Chicago Police Department. The dataset includes the location and time of the incidents as well as their types such as theft, burglary, assault etc. Before collecting results, we grouped event types into four different classes considering their contextual meanings *(1: Assault/Battery/Offense; 2: Burglary/Robbery/Theft; 3: Criminal Damage/Violations; 4: Others)*. We particularly work in June 2019, and filter the locations spatially between the latitudes of [41.85, 41.92] and longitudes of [-87.65, -87.62] to remove outlier regions.

b) Earthquake dataset: The National Earthquake Information Center provides this dataset that includes earthquakes with a magnitude of 4.5 or higher since 1986. Every earthquake entry includes a record of the date, time, location and magnitude. We filter the dataset spatially and work on the events occurred in Turkey, which is between the latitudes of [36, 42] and longitudes of [26, 45]. In addition, we have defined the event types according to the common categorization in the seismology literature on the Richter scale [54] such that the first event type represents the earthquakes with a magnitude less than 5 (light), the second event type represents the earthquakes with a magnitude between 5 and 6 (moderate), and the third event type represents the earthquakes with a magnitude greater



(a) (b)

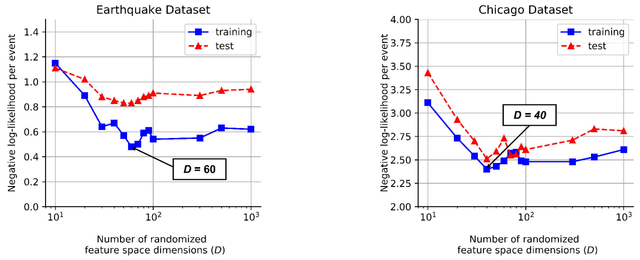
Fig. 3. Training curves of the introduced method (RFF-GD), EM [11] and stochastic declustering (SD) [20] for the earthquake dataset (a) and the Chicago crime dataset (b).

TABLE V
TRAINING PERFORMANCE OF OUR ALGORITHM, EM [11] AND STOCHASTIC DECLUSTERING (SD) [20] ON REAL-LIFE DATASETS (p: NUMBER OF PARAMETERS, $\overline{\mathcal{L}}$: NEGATIVE LOG-LIKELIHOOD PER EVENT)

Method	Dataset					
	Earthquake			Chicago		
	p	$\overline{\mathcal{L}}$	AIC	p	$\overline{\mathcal{L}}$	AIC
RFF-GD	**39**	**0.48**	**929.52**	**60**	**2.40**	**12480**
EM [11]	21	0.80	1461.2	35	2.55	13202.5
SD [20]	6	0.66	1182.84	6	2.86	14741

TABLE VI
TEST PERFORMANCE OF OUR ALGORITHM, EM [11] AND STOCHASTIC DECLUSTERING (SD) [20] ON REAL-LIFE DATASETS IN TERMS OF NEGATIVE LOG-LIKELIHOOD PER EVENT

Method	Dataset	
	Earthquake	Chicago
RFF-GD	**0.83**	**2.51**
EM [11]	1.20	2.68
SD [20]	0.97	2.90



(a) (b)

Fig. 4. D (number of randomized feature dimensions) vs. negative log-likelihood per event for the training and test sets of the earthquake dataset (a) and the Chicago crime dataset (b).

than 6 (strong). In the seismology literature, it has been shown that strong earthquakes cause aftershocks, i.e. earthquakes with small magnitudes [13], [29], [54]. Therefore, our representation enables us to infer the triggering relation between earthquakes from different magnitude ranges.

2) Real-Life Dataset Performance: We investigate the fitting performance of the proposed optimization algorithm, and compare it with the EM algorithm proposed in a recent work [11] and stochastic declustering [20]. As in the synthetic dataset experiments, we scale the given event sequence spatio-temporally. In addition, we repeat the experiments 10 times to reduce the random effects on performance.

In the first set of experiments, we consider the earthquake dataset. We perform hyperparameter search over $D \in [10, 1000]$, $\eta \in [0.0001, 0.1]$, $b \in [32, 512]$ and $s \in [0.001, 0.1]$. We stop the training if the performance does not improve for $k = 30$ consecutive steps and save the best iteration as our reference. For the proposed method, we obtain the best results for $D = 60$, $\eta = 0.002$, $b = 512$ and $s = 0.01$. For the second experiment set, we work on the Chicago crime dataset. We perform hyperparameter search over the same parameter ranges. In this experiment, we obtain the best performance with $D = 40$, $\eta = 0.01$, $b = 256$ and $s = 0.01$.

In Table V, we provide the number of parameters, negative log-likelihood per event and AIC values of the EM algorithm [11], stochastic declustering [20], and the introduced method for training set. Our method have more parameters due to the weight decay matrix and covariance matrices. We also provide the negative log-likelihood per event values for the test set in Table VI. On both datasets, our approach significantly outperforms other methods in terms of negative log-likelihood per event and AIC, which indicates that the inferred parameters by our method represent the given event sequence more successfully. We also illustrate the training curves for these experiments in Fig 3(a), 3(b) respectively.

To illustrate the effect of the number of random Fourier feature dimensions, we provide Fig. 4. In Fig. 4(a), for the earthquake dataset, we observe that the optimum choice for the randomized transformation dimensions is around 70. The performance significantly drops when D becomes very small. If D gets very high, we do not obtain a considerable amount of

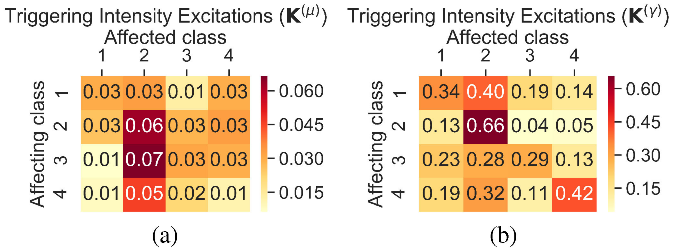


Fig. 5. For the Chicago crime dataset, (a) Excitation matrix of base intensity ($\mathbf{K}^{(\mu)}$), (b) Excitation matrix of triggering intensity ($\mathbf{K}^{(\gamma)}$).

performance gain, in fact, the performance drops slightly. For the Chicago crime dataset, we observe a similar behavior as can be seen in Fig. 4(b). In this case, the best performance is achieved when $D = 40$. Increasing this value causes negative log-likelihood per event to reach values between 2.5 and 2.6. Therefore, it is clear that the introduced tunable randomization while modeling the spatial excitation enhances the performance in real-life scenarios, particularly when the spatial dynamics of the underlying system deviates from pure Gaussian behavior.

After investigating the fitting performance, we focus on the inferred parameters, which inherently reflect the dynamics of the given event sequence. For this purpose, we provide the estimated excitation matrices for the base and triggering intensities in Fig. 5. This analysis directly reveals the triggering effect between different crime types. In this scenario, we observe that the base excitation values are more homogeneous compared to the triggering excitation values. In particular, crime events from class 2 (burglary/robbery/theft) have a strong self-excitation with respect to other event types. We also realize that events from class 2 are significantly triggered by other event types, whereas their effect on others is limited. On the contrary, events from class 3 (criminal damage/violations) exhibit strong excitation over all event types however, they are not considerably excited by other event types.

V. Conclusion

We studied spatio-temporal Hawkes processes to perform spatio-temporal event analysis. We introduce a novel framework for spatio-temporal Hawkes processes to extend the conventional methods in the literature such as EM [11] and stochastic declustering [20]. Our approach utilizes the randomization introduced by random Fourier features based spatial kernel representation, and increases the flexibility of the model in terms of spatial modeling capability. Moreover, we express the problem in a neat scalable matrix formulation. We analytically calculate the intractable terms in the likelihood function, and derive the gradient equations for maximum likelihood optimization. To satisfy the structural constraints of the process parameters, we use reparameterization techniques and projected gradient descent. We also propose a thinning-based simulation algorithm for spatio-temporal Hawkes processes with multiple event types. We analyze the improvements achieved by the proposed method on various simulations and two real-life datasets. The comparisons show that the proposed method significantly performs better in terms of negative log-likelihood and AIC compared to other methods. In addition, we interpret the learned process parameters and perform event analysis over these real-life datasets through analyzing the triggering relations between event types.

Appendix A

We can obtain the derivatives for the base and triggering intensity excitation matrices introduced in (4) and (5) as

$$\frac{\partial \mathcal{L}}{\partial \mathbf{K}^{(\cdot)}} = -\mathbf{Q}^{(\cdot)^T}(\mathbf{Y} \oslash \mathbf{A}) + \frac{\partial R}{\partial \mathbf{K}^{(\cdot)}}, \tag{34}$$

where $\frac{\partial R}{\partial \mathbf{K}^{(\cdot)}}$ consists of the elements $\left[\frac{\partial R}{\partial K_{mn}^{(\cdot)}}\right]$, which can be expressed as $\frac{\partial R}{\partial K_{mn}^{(\mu)}} = \sum_{j=1}^{N} \delta_{u_j m}$ and $\frac{\partial R}{\partial K_{mn}^{(\gamma)}} = \sum_{j=1}^{N} \delta_{u_j m}(1 - e^{-w_{u_j n}(T - t_j)})$. We, then, express the derivative of the decay rate matrix as $\frac{\partial \mathcal{L}}{\partial \mathbf{W}} = \left[\frac{\partial \mathcal{L}}{\partial w_{mn}}\right]$, where each element is derived as

$$\frac{\partial \mathcal{L}}{\partial w_{mn}} = -\text{sum}\left(\left(\frac{\partial \mathbf{A}}{\partial w_{mn}}\right)^T (\mathbf{Y} \oslash \mathbf{A})\right) + \frac{\partial R}{\partial w_{mn}}. \tag{35}$$

Here, $\frac{\partial R}{\partial w_{mn}} = \sum_{j=1}^{N} \delta_{u_j m}(T - t_j) e^{-w_{u_j n}(T - t_j)}$, and $\frac{\partial \mathbf{A}}{\partial w_{mn}}$ consists of the rows

$$\frac{\partial \boldsymbol{a}_i^T}{\partial w_{mn}} = \frac{1}{2\pi}|\boldsymbol{\Sigma}^{(\gamma)}|^{-1/2} \boldsymbol{z}_i^T \mathbf{Z}_{J(t_i)}^{(\gamma)^T} \text{diag}\left(\frac{\partial \boldsymbol{d}_{J(t_i)}}{\partial w_{mn}}\right) \mathbf{Y}_{J(t_i)} \mathbf{K}^{(\gamma)}, \tag{36}$$

where

$$\frac{\partial \boldsymbol{d}_{J(t_i)}}{\partial w_{mn}} = \begin{bmatrix} \vdots \\ \delta_{u_i m}\delta_{u_j n}(1 - w_{u_j u_i}(t_i - t_j))e^{-w_{u_j u_i}(t_i - t_j)} \\ \vdots \end{bmatrix}. \tag{37}$$

For the spatial kernel parameters, we first derive the gradients of $\mathbf{V}^{(\mu)}$ and $\mathbf{V}^{(\gamma)}$ as $\frac{\partial \mathcal{L}}{\partial \mathbf{V}^{(\cdot)}} = \left[\frac{\partial \mathcal{L}}{\partial V_{mn}^{(\cdot)}}\right]$ where each element is derived as

$$\frac{\partial \mathcal{L}}{\partial V_{mn}^{(\cdot)}} = -\text{sum}\left(\left(\frac{\partial \mathbf{A}}{\partial V_{mn}^{(\cdot)}}\right)^T (\mathbf{Y} \oslash \mathbf{A})\right). \tag{38}$$

Here, $\frac{\partial \mathbf{A}}{\partial V_{mn}^{(\cdot)}}$ consists of the rows $\frac{\partial \boldsymbol{a}_i^T}{\partial V_{mn}^{(\cdot)}}$ such that

$$\begin{aligned} \frac{\partial \boldsymbol{a}_i^T}{\partial V_{mn}^{(\mu)}} = \frac{1}{2\pi T}|\boldsymbol{\Sigma}^{(\mu)}|^{-1/2} \Bigg(& \frac{\partial \boldsymbol{z_i}^{(\mu)^T}}{\partial V_{mn}^{(\mu)}} \mathbf{Z}_{J(t_i)}^{(\mu)^T} \\ & + \boldsymbol{z_i}^{(\mu)^T} \frac{\partial \mathbf{Z}_{J(t_i)}^{(\mu)^T}}{\partial V_{mn}^{(\mu)}} \Bigg) \mathbf{Y}_{J_{(t_i)}} \mathbf{K}^{(\mu)}, \end{aligned} \tag{39}$$

$$\begin{aligned} \frac{\partial \boldsymbol{a}_i^T}{\partial V_{mn}^{(\gamma)}} = \frac{1}{2\pi}|\boldsymbol{\Sigma}^{(\gamma)}|^{-1/2} \Bigg(& \frac{\partial \boldsymbol{z_i}^{(\gamma)^T}}{\partial V_{mn}^{(\gamma)}} \mathbf{Z}_{J(t_i)}^{(\gamma)^T} \\ & + \boldsymbol{z_i}^{(\gamma)^T} \frac{\partial \mathbf{Z}_{J(t_i)}^{(\gamma)^T}}{\partial V_{mn}^{(\gamma)}} \Bigg) \text{diag}(\boldsymbol{d}_{J(t_i)}) \mathbf{Y}_{J(t_i)} \mathbf{K}^{(\gamma)}, \end{aligned} \tag{40}$$

where $\frac{\partial \mathbf{Z}_{J(t_i)}^{(\cdot)^T}}{\partial V_{mn}^{(\cdot)}}$ consists of the rows $\frac{\partial \boldsymbol{z_j}^{(\cdot)^T}}{\partial V_{mn}^{(\cdot)}} = -\sqrt{\frac{2}{D}} s_{im} \ell_n^{1/2} \boldsymbol{u_n}^T \odot \sin\left(\boldsymbol{s_i}^T \mathbf{V}^{(\cdot)} \boldsymbol{\Lambda}^{(\cdot)^{-\mathbf{1/2}}} \mathbf{U} + \boldsymbol{b}^T\right)$, and $\boldsymbol{u_n}^T$ is the n^{th} row of $\mathbf{U}$. Finally, we obtain the derivatives of $\boldsymbol{\ell}^{(\mu)}$ and $\boldsymbol{\ell}^{(\gamma)}$ as $\frac{\partial \mathcal{L}}{\partial \boldsymbol{\ell}} = \begin{bmatrix} \dots & \frac{\partial \mathcal{L}}{\partial \ell_n} & \dots \end{bmatrix}^T$, where each element is defined as $\frac{\partial \mathcal{L}}{\partial \ell_n} = -\mathrm{sum}((\frac{\partial \mathbf{A}}{\partial \ell_n})^T (\mathbf{Y} \oslash \mathbf{A}))$. Here, $\frac{\partial \mathbf{A}}{\partial \ell_n}$ consists of the rows $\frac{\partial \boldsymbol{a}_i^T}{\partial \ell_n}$ such that

$$\begin{aligned} \frac{\partial \boldsymbol{a}_i^T}{\partial \ell_n^{(\mu)}} = \frac{1}{2\pi T} \Bigg(& \frac{\partial |\boldsymbol{\Sigma}^{(\mu)}|^{-1/2}}{\partial \ell_n} \boldsymbol{z_i}^{(\mu)^T} \mathbf{Z}_{J(t_i)}^{(\mu)^T} \\ & + |\boldsymbol{\Sigma}^{(\mu)}|^{-1/2} \Bigg(\frac{\partial \boldsymbol{z_i}^{(\mu)^T}}{\partial \ell_n} \mathbf{Z}_{J(t_i)}^{(\mu)^T} \\ & + \boldsymbol{z_i}^{(\mu)^T} \frac{\partial \mathbf{Z}_{J(t_i)}^{(\mu)^T}}{\partial \ell_n} \Bigg) \Bigg) \mathbf{Y}_{J(t_i)} \mathbf{K}^{(\mu)}, \end{aligned} \tag{41}$$

$$\begin{aligned} \frac{\partial \boldsymbol{a}_i^T}{\partial \ell_n^{(\gamma)}} = \frac{1}{2\pi} \Bigg(& \frac{\partial |\boldsymbol{\Sigma}^{(\gamma)}|^{-1/2}}{\partial \ell_n} \boldsymbol{z_i}^{(\gamma)^T} \mathbf{Z}_{J(t_i)}^{(\gamma)^T} \\ & + |\boldsymbol{\Sigma}^{(\gamma)}|^{-1/2} \Bigg(\frac{\partial \boldsymbol{z_i}^{(\gamma)^T}}{\partial \ell_n} \mathbf{Z}_{J(t_i)}^{(\gamma)^T} \\ & + \boldsymbol{z_i}^{(\gamma)^T} \frac{\partial \mathbf{Z}_{J(t_i)}^{(\gamma)^T}}{\partial \ell_n} \Bigg) \Bigg) \mathrm{diag}(\boldsymbol{d}_{J(t_i)}) \mathbf{Y}_{J(t_i)} \mathbf{K}^{(\gamma)}, \end{aligned} \tag{42}$$

where $\frac{\partial \mathbf{Z}_{J(t_i)}^{(\cdot)^T}}{\partial \ell_n^{(\cdot)}}$ consists of the rows $\frac{\partial \boldsymbol{z_j}^{(\cdot)^T}}{\partial \ell_n^{(\cdot)}} = \sqrt{\frac{1}{2D}} \boldsymbol{s}_i^T \boldsymbol{v}_n^{(\cdot)} \ell_n^{(\cdot)^{-1/2}} \boldsymbol{u_n}^T \odot \sin\left(\boldsymbol{s_i}^T \mathbf{V}^{(\cdot)} \boldsymbol{\Lambda}^{(\cdot)^{-\mathbf{1/2}}} \mathbf{U} + \boldsymbol{b}^T\right)$, $\boldsymbol{v_n}^{(\cdot)}$ is the n^{th} column of $\mathbf{V}^{(\cdot)}$ and $\boldsymbol{u_n}^T$ is the n^{th} row of $\mathbf{U}$.

References

[1] C. Jiang, Y. Chen, and K. J. R. Liu, "Evolutionary dynamics of information diffusion over social networks," *IEEE Trans. Signal Process.*, vol. 62, no. 17, pp. 4573–4586, Sep. 2014.

[2] W. Ge and R. T. Collins, "Crowd density analysis with marked point processes [applications corner]," *IEEE Signal Process. Mag.*, vol. 27, no. 5, pp. 107–123, Sep. 2010.

[3] R. W. Heath, M. Kountouris, and T. Bai, "Modeling heterogeneous network interference using poisson point processes," *IEEE Trans. Signal Process.*, vol. 61, no. 16, pp. 4114–4126, Aug. 2013.

[4] P. Zhang, I. Nevat, G. Peters, G. Xiao, and H.-P. Tan, "Event detection in wireless sensor networks in random spatial sensors deployments," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 6122–6135, Nov. 2015.

[5] C. Luo, X. Zheng, and D. Zeng, "Inferring social influence and meme interaction with hawkes processes," in *Proc. IEEE Int. Conf. Intell. Security Informat.*, May 2015, pp. 135–137.

[6] C. Yu, W. Ding, M. Morabito, and P. Chen, "Hierarchical spatio-temporal pattern discovery and predictive modeling," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 4, pp. 979–993, Apr. 2016.

[7] G. O. Mohler, M. B. Short, P. J. Brantingham, F. P. Schoenberg, and G. E. Tita, "Self-exciting point process modeling of crime," *J. Amer. Statistical Assoc.*, vol. 106, no. 493, pp. 100–108, 2011.

[8] S. Sen, "OFDM radar space-time adaptive processing by exploiting spatio-temporal sparsity," *IEEE Trans. Signal Process.*, vol. 61, no. 1, pp. 118–130, Jan. 2013.

[9] D. Kuzin, O. Isupova, and L. Mihaylova, "Spatio-temporal structured sparse regression with hierarchical Gaussian process priors," *IEEE Trans. Signal Process.*, vol. 66, no. 17, pp. 4598–4611, Sep. 2018.

[10] V. Roberto and C. Chiaruttini, "Seismic signal understanding: A knowledge-based recognition system," *IEEE Trans. Signal Process.*, vol. 40, no. 7, pp. 1787–1806, Jul. 1992.

[11] B. Yuan, H. Li, A. L. Bertozzi, P. J. Brantingham, and M. A. Porter, "Multivariate spatiotemporal hawkes processes and network reconstruction," *SIAM J. Math. Data Sci.*, vol. 1, no. 2, pp. 356–382, 2019.

[12] Z. Gao *et al.*, "EEG-based spatio-temporal convolutional neural network for driver fatigue evaluation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2755–2763, Sep. 2019.

[13] Y. Ogata, "Statistical models for earthquake occurrences and residual analysis for point processes," *J. Amer. Statistical Assoc.*, vol. 83, no. 401, pp. 9–27, 1988.

[14] D. J. Daley and D. Vere-Jones, *An Introduction to the Theory of Point Processes*. Vol. I: Elementary Theory and Methods, 2nd ed., 2003, pp. 111–112.

[15] C. Li, Z. Song, and X. Wang, "Nonparametric method for modeling clustering phenomena in emergency calls under spatial-temporal self-exciting point processes," *IEEE Access*, vol. 7, pp. 24 865–24 876, 2019.

[16] A. Reinhart, "A review of self-exciting spatio-temporal point processes and their applications," *Statistical Sci.*, vol. 33, no. 3, pp. 299–318, Aug. 2018.

[17] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. Advances Neural Inf. Process. Syst.*, 2008, pp. 1177–1184.

[18] Y.-S. Cho, A. Galstyan, J. Brantingham, and G. Tita, "Latent point process models for spatial-temporal networks," *Discrete Continuous Dynamical Syst. - Series B*, vol. 19, no. 5, pp. 1335–1354, 2013.

[19] A. Veen and F. P. Schoenberg, "Estimation of spacetime branching process models in seismology using an emtype algorithm," *J. Amer. Statistical Assoc.*, vol. 103, no. 482, pp. 614–624, 2008.

[20] J. Zhuang, Y. Ogata, and D. Vere-Jones, "Stochastic declustering of space-time earthquake occurrences," *J. Amer. Statistical Assoc.*, vol. 97, no. 458, pp. 369–380, 2002.

[21] G. Adelfio and M. Chiodi, "FLP estimation of semi-parametric models for space-time point processes and diagnostic tools," *Spatial Statist.*, vol. 14, pp. 119–132, Jun. 2015.

[22] D. Marsan and O. Lenglin, "A new estimation of the decay of aftershock density with distance to the mainshock," *J. Geophysical Res.: Solid Earth*, vol. 115, no. B9, 2010, Art. no. 09302.

[23] E. W. Fox, F. P. Schoenberg, and J. S. Gordon, "Spatially inhomogeneous background rate estimators and uncertainty quantification for nonparametric hawkes point process models of earthquake occurrences," *Ann. Appl. Stat.*, vol. 10, no. 3, pp. 1725–1756, 2016.

[24] S. Li, S. Xiao, S. Zhu, N. Du, Y. Xie, and L. Song, "Learning temporal point processes via reinforcement learning," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, ser. NIPS18, 2018, pp. 10804–10814.

[25] N. Du, H. Dai, R. Trivedi, U. Upadhyay, M. Gomez-Rodriguez, and L. Song, "Recurrent marked temporal point processes: Embedding event history to vector," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2016, pp. 1555–1564.

[26] B. Wang, X. Luo, F. Zhang, B. Yuan, A. L. Bertozzi, and P. J. Brantingham, "Graph-based deep modeling and real time forecasting of sparse spatio-temporal data," 2018, *arXiv:1804.00684*.

[27] M. Chiodi and G. Adelfio, "Forward likelihood-based predictive approach for spacetime point processes," *Environmetrics*, vol. 22, no. 6, pp. 749–757, 2011.

[28] D. Marsan and O. Lengliné, "Extending earthquakes' reach through cascading," *Science*, vol. 319, no. 5866, pp. 1076–1079, 2008.

[29] F. Musmeci and D. Vere-Jones, "A space-time clustering model for historical earthquakes," *Ann. Inst. Statistical Math.*, vol. 44, pp. 1–11, 02 1992.

[30] B. Mark, G. Raskutti, and R. Willett, "Network estimation from point process data," *IEEE Trans. Inf. Theory*, vol. 65, no. 5, pp. 2953–2975, May 2019.

[31] S. W. Linderman and R. P. Adams, "Discovering latent network structure in point process data," ser. ICML14, 2014, pp. II-1413–II-1421.

[32] C. Blundell, J. Beck, and K. A. Heller, "Modelling reciprocating relationships with hawkes processes," in *Proc. Advances Neural Inf. Process. Syst. 25*, 2012, pp. 2600–2608.

[33] V. Isham and M. Westcott, "A self-correcting point process," *Stochastic Processes Their Appl.*, vol. 8, no. 3, pp. 335–347, 1979.

[34] K. Xiong and S. Wang, "The online random fourier features conjugate gradient algorithm," *IEEE Signal Process. Lett.*, vol. 26, no. 5, pp. 740–744, May 2019.

[35] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1, pp. 489–501, Dec. 2006.

[36] S. Zhou, X. Liu, Q. Liu, S. Wang, C. Zhu, and J. Yin, "Random fourier extreme learning machine with 2,1-norm regularization," *Neurocomputing*, vol. 174, pp. 143–153, 2014.

[37] H. Mei and J. Eisner, "The neural hawkes process: A neurally self-modulating multivariate point process," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, ser. NIPS17, 2017, pp. 6757–6767.

[38] T. Hastie, R. Tibshirani, and M. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*. Boston, MA, USA: Chapman & Hall/CRC, 2015.

[39] B. T. Polyak and P. Shcherbakov, "Why does monte carlo fail to work properly in high-dimensional optimization problems?" *J. Optim. Theory Appl.*, vol. 173, pp. 612–627, 2016.

[40] L. Xu and M. I. Jordan, "On convergence properties of the em algorithm for gaussian mixtures," *Neural Comput.*, vol. 8, no. 1, pp. 129–151, 1996.

[41] C. Couvreur, *The EM Algorithm: A Guided Tour*. Boston, MA, USA: Birkhäuser Boston, 1997, pp. 209–222.

[42] Y. Ogata and K. Katsura, "Likelihood analysis of spatial inhomogeneity for marked point patterns," *Ann. Inst. Statistical Math.*, vol. 40, no. 1, pp. 29–39, Mar. 1988.

[43] J. Rasmussen, "Bayesian inference for hawkes processes," *Methodology Comput. Appl. Probability*, vol. 15, pp. 623–642, 2013.

[44] M. Han, J. Xi, S. Xu, and F.-L. Yin, "Prediction of chaotic time series based on the recurrent predictor neural network," *IEEE Trans. Signal Process.*, vol. 52, no. 12, pp. 3409–3416, Dec. 2004.

[45] J. Lu, S. C. Hoi, J. Wang, P. Zhao, and Z.-Y. Liu, "Large scale online kernel learning," *J. Mach. Learn. Res.*, vol. 17, no. 47, pp. 1–43, 2016.

[46] S. S. Bochner, *Lectures on Fourier Integral*. Akademische Verlagsgesellschaft, 1932.

[47] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, vol. 15. Apr. 2011, pp. 315–323.

[48] P. H. Schnemann, "A generalized solution of the orthogonal procrustes problem," *Psychometrika*, vol. 31, no. 1, pp. 1–10, Mar. 1966.

[49] Y. Ogata, "On lewis' simulation method for point processes," *IEEE Trans. Inf. Theory*, vol. IT-27, no. 1, pp. 23–31, Jan. 1981.

[50] S. Zhu, S. Li, and Y. Xie, "Reinforcement learning of spatio-temporal point processes," *CoRR*, 2019. [Online]. Available: https://arxiv.org/abs/1906.05467

[51] H. Akaike, *Inf. Theory and an Extension of the Maximum Likelihood Principle*. New York, NY: Springer New York, 1998, pp. 199–213.

[52] J. Chen, A. Hawkes, E. Scalas, and M. Trinh, "Performance of information criteria used for model selection of hawkes process models of financial data," *SSRN Electron. J.*, pp. 1–11, Feb. 2017.

[53] Y. Ogata, "Space-time point-process models for earthquake occurrences," *Ann. Inst. Statistical Math.*, vol. 50, no. 2, pp. 379–402, Jun. 1998.

[54] H. Kanamori, "Quantification of earthquakes," *Nature*, vol. 271, no. 5644, pp. 411–414, Feb. 1978.