# Learning Triggering Kernels for Multi-dimensional Hawkes Processes

**Ke Zhou** KZHOU@GATECH.EDU
Georgia Institute of Technology

**Hongyuan Zha** ZHA@CC.GATECH.EDU
Georgia Institute of Technology

**Le Song** LSONG@CC.GATECH.EDU
Georgia Institute of Technology

## Abstract

How does the activity of one person affect that of another person? Does the strength of influence remain periodic or decay exponentially over time? In this paper, we study these critical questions in social network analysis quantitatively under the framework of multi-dimensional Hawkes processes. In particular, we focus on the nonparametric learning of the triggering kernels, and propose an algorithm MMEL that combines the idea of decoupling the parameters through constructing a tight upper-bound of the objective function and application of Euler-Lagrange equations for optimization in infinite dimensional functional space. We show that the proposed method performs significantly better than alternatives in experiments on both synthetic and real world datasets.

## 1. Introduction

Real world interactions between multiple entities, such as earthquake aftershocks (Vere-Jones, 1970), civilian death in conflicts (Lewis et al., 2011) and user behaviors in social network (Mitchell and Cates, 2010), often exhibit the self-exciting and mutually-exciting property. For example, the time of aftershocks are usually close to the main shock and may trigged further aftershocks in the future. Multi-dimensional Hakwes processes, an important class of mutually exciting process, can be used to capture these interactions.

*Proceedings of the 30$^{th}$ International Conference on Machine Learning*, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28. 

Formally, the multi-dimensional Hawkes process is defined by a $U$-dimensional point process $N_t^u, u = 1, \ldots, U$, with the conditional intensity for the $u$-th dimension expressed as follows:

$$\lambda_u(t) = \mu_u + \sum_{i:t_i<t} g_{uu_i}(t - t_i),$$

where $\mu_u \geq 0$ is the base intensity for the $u$-th Hawkes process. The kernel $g_{uu'}(t) \geq 0$ captures the mutually-exciting property between the $u$-th and $u'$-th dimension. Intuitively, it captures the dynamics of influence of events occurred in the $u'$-th dimension to the $u$-th dimension. Larger value of $g_{uu'}(t)$ indicates that events in $u'$-th dimension are more likely to trigger a event in the $u$-th dimension after a time interval $t$.

Despite the usefulness of the mutually exciting property in real world problems, the actual dynamics of *how* previous events trigger future events, which is modeled by the *triggering kernels* $g_{uu'}(t)$, can be quite complex and vary a lot across different applications. For example, the dynamics of user behaviors in social network can be very different from those in earthquake after shocks or disease contagion. Moreover, these dynamics can be inherently complex since the diverse nature of user behaviors. Unfortunately, most existing work based on Hawkes processes assumes that the triggering kernels are known or chosen manually in advance, which trends to be oversimplified or even infeasible for capturing the problem complexity in many applications. Therefore, it is highly desirable to estimate the temporal dynamics in a principled data-driven way rather than relying on ad-hoc manual selections.

In this paper, we propose a general framework to estimate the triggering kernels of Hawkes processes from the recurrent temporal events that can be viewed as samples from the Hawkes processes — without the knowledge of the actual triggering structure in the

events. The challenge of the problem arises not only from the fact that the parameters is in an infinite dimensional space but also they are coupled with each other in the likelihood function. To address the problem, we propose MMEL which applies idea of majorization minimization (Hunter and Lange, 2003) to construct an upper-bound of the objective function at each iteration that decouples the parameters so that they can be optimized independently. Another novelty of our method is that we used the Euler-Lagrange equation to derive an ordinary differential equation (ODE) that the optimal trigger kernel should satisfy. This connection allows us to exploit the fruitful and well-developed techniques of ODE solvers. In our experiments on both synthetic and real world datasets, the proposed method performs significantly better than alternatives.

The rest of the paper is organized as follows: In Section 2, we briefly summarize related work in Hawkes process, smoothing splines and kernel learning. In Section 3, we describe the multi-dimensional Hawkes model in details, together with the proposed algorithm for estimating the triggering kernel. In Section 4, the results on both numerical simulations and real-world applications are reported and analyzed. Finally, we conclude our work and discuss future directions in Section 5.

## 2. Related Work

Mutually-exciting point processes are frequently used to model continuous-time events where the occurrences of previous events increase the possibility of future events. Hawkes processes (Hawkes, 1971), an important type of mutually-exciting process, have been investigated for a wide range of applications such as market modeling (Toke, 2010), earth quake prediction (Marsan and Lengliné, 2008), crime modeling (Stomakhin et al., 2011). Additionally, (Simma and Jordan, 2012) models cascades of events using marked Poisson processes with marks representing the types of events while (Blundell et al., 2012) propose a model based on Hawkes processes that models events between pairs of nodes. The problem of nonparametric estimation of the triggering kernels has been addressed for special cases such as the one-dimensional Hawkes processes (Lewis and Mohler, 2011) or symmetric Hawkes processes (Bacry et al., 2012). In this work, we study the general case of multi-dimensional Hawkes process under the framework for optimizing the triggering kernels in the infinite dimensional functional space. The recent work of (Du et al., 2012) considers the problem of learning the triggering kernel for diffusion processes based on linear combination of basis functions. Our proposed method, on the other hand, does not assume any known parametric forms of basis functions and estimate them from observed data through optimization in an infinite dimensional functional space.

Another related direction of studies is smoothing splines (Reinsch, 1967; Wahba, 1990) in the sense that the problem of estimating the triggering kernels can be viewed as a smoothing problem with nonnegative constraints. The goal of smoothing splines is to estimate a smooth function based on its value on finite points. The nonnegative constraints are usually studied as the more general case of the shape restrictions (Reinsch, 1967; Mammen and Thomas-agnan, 1998; Turlach, 1997). The main difference in our work is that the loss function we consider is more complex and depends on the values of the triggering kernels on infinite points, which makes it difficult to directly apply the smoothing spline methods. Moreover, as we will see later, the nonnegative constraints can be naturally enforced in our algorithm.

Positive definite kernels have also been used extensively in machine learning. This type of kernel can be viewed as similarity function between data points. Its learning has been addressed extensively in recent literature where one tries to learn a better positive definite kernel by combining several positive definite kernels (Cortes et al., 2012; Argyriou et al., 2006; Bach, 2008; Dinuzzo et al., 2011; Sonnenburg et al., 2006). Nonparametric positive kernel learning, instead of learning combination of existing positive kernels, directly learns the full Gram matrix with respect to certain constraints and prior knowledges about the data, such as pairwise constraints (Hoi et al., 2007) or distribution of the data (Zhu et al., 2004).

## 3. Nonparametric Triggering Kernel Estimation using Euler-Lagrange Equations

We collect the parameters of the multi-dimensional Hawkes process into matrix-vector forms, $\boldsymbol{\mu} = (\mu_u)$ for the base intensity and $\mathbf{G} = (g_{uu'}(t))$ into a matrix. These parameters can be estimated by optimizing the log-likelihood over the observed events that are sampled from the process.

### 3.1. Optimization Problem and Space

Suppose we have $m$ samples, $\{c_1, \ldots, c_m\}$, from the multi-dimensional Hawkes process. Each sample $c$ is a sequence of events observed during a time period

of $[0, T_c]$, which is in the form of $\{(t_i^c, u_i^c)\}_{i=1}^{n_c}$. Each pair $(t_i^c, u_i^c)$ represents an event occurring at the $u_i^c$-th dimension at time $t_i^c$. Thus, the log-likelihood of model parameters $\Theta = \{\mathbf{G}, \boldsymbol{\mu}\}$ can be expressed as follows (Liniger, 2009):

$$\begin{aligned}\mathcal{L}(\Theta) &= \sum_c \left( \sum_{i=1}^{n_c} \log \lambda_{u_i^c}(t_i^c) - \sum_{u=1}^{U} \int_0^{T_c} \lambda_u(t) dt \right) \\ &= \sum_c \left( \sum_{i=1}^{n_c} \log \left( \mu_{u_i^c} + \sum_{t_j^c < t} g_{u_i^c u_j^c}(t_i^c - t_j^c) \right) \right. \\ &\quad \left. - T_c \sum_{u=1}^{U} \mu_u - \sum_{u=1}^{U} \sum_{j=1}^{n_c} \int_0^{T_c - t_j} g_{u u_j^c}(s) ds \right). \quad (1)\end{aligned}$$

In general, the triggering kernels $g_{uu'}(t)$ as well as the base intensity $\boldsymbol{\mu}$ can be estimated by maximizing the log-likelihood, *i.e.,* $\min_{g_{uu'}(t) \geq 0, \boldsymbol{\mu} \geq 0} -\mathcal{L}(\Theta)$.

We assume that the triggering kernels $g_{uu'}(t)$ can be expressed by linear combination of a set of $D$ base kernels. Formally, we have

$$g_{uu'}(t) = \sum_{d=1}^{D} a_{uu'}^d g_d(t),$$

where $\{g_d(t) | d = 1, 2, \ldots, D\}$ are the base kernels and $a_{uu'}^d$ are the coefficients for the linear combination. In our work, both $a_{uu'}^d$ and $g_d(t)$ are estimated from the data. In particular, we propose to estimate the base kernels $g_d(t)$ from an infinite dimensional functional space. To this end, we consider the following penalized log-likelihood function function, i.e., $\min_\Theta \mathcal{L}_\alpha(\Theta)$, where the penalized log-likelihood function $\mathcal{L}_\alpha(\Theta)$ is defined as follows:

$$\mathcal{L}_\alpha(\Theta) = -\mathcal{L}(\Theta) + \alpha \left( \sum_d \mathcal{R}(g_d) + \sum_{u,u',d} (a_{uu'}^d)^2 \right).$$

Here the first term is the negative log-likelihood of the parameters and the second term represents the regularization of both the base function $g_d(t)$ and the coefficient $a_{uu'}^d$. The parameter $\alpha$ determines the trade-off between these two terms. Moreover, the functional $\mathcal{R}(g_d)$ is a penalty term preferring smooth base kernels. In general, the choice of the penalty should take into account of the prior knowledge of the triggering kernels and thus is application-dependent. For the sake of concreteness and tractability, we use $\mathcal{R}(g) = \int_0^\infty g'(t)^2 dt$ in the rest of our paper, where $g'(t)$ is the derivative of $g(t)$ with respect to $t$.

It appears that the above problem is similar to the smoothing splines and can be solved through methods that transform the above problem to a finite dimensional least squares optimization problem (Wahba, 1990). As we discussed in Section 2, however, the main difference is that the log-likelihood function defined in Equation (1) contains the integral over the triggering kernels that depends on the values of the triggering kernels over the whole time interval rather than only a finite number of points as required by the smoothing splines. As a result, it is difficult to directly apply the smoothing spline methods in our case.

Even more challenging is that the above objective function is difficult to optimize in general due to the fact that the parameters are not only infinite dimensional but also are coupled. In this paper, inspired by Lewis and Mohler (2011), we propose MMEL which combines the idea of constructing a tight upper-bound as a surrogate to decouple parameters and the application of Euler-Lagrange equations to deal with the infinite dimensionality of the parameters.

### 3.2. Iterative Algorithm

Our algorithm updates the parameters $\Theta$ in an iterative manner which, as we will show later, ensures that the objective function $\mathcal{L}_\alpha$ decrease monotonically. In particular, we construct a tight upper-bound $Q(\Theta|\Theta^{(k)})$ for current parameter estimation $\Theta^{(k)}$ and optimize the upper-bound $Q(\Theta|\Theta^{(k)})$ to obtain the updates for the parameters. Specifically, the upper-bound $Q(\Theta|\Theta^{(k)})$ is defined as follows:

$$\begin{aligned}Q(\Theta|\Theta^{(k)}) = &- \sum_c \left[ \sum_{i=1}^{n_c} \left( p_{ii}^c \log \frac{\mu_{u_i^c}}{p_{ii}^c} + \right. \right. \\ &\left. \sum_{j=1}^{i-1} \sum_{d=1}^{D} p_{ijd}^c \log \frac{a_{u_i^c u_j^c}^d g_d(t_i^c - t_j^c)}{p_{ijd}^c} \right) + \left( T_c \sum_u \mu_u + \right. \\ &\left. \left. \sum_{u=1}^{U} \sum_{j=1}^{n_c} \sum_{d=1}^{D} \int_0^{\tau_j^c} \left( (a_{u u_j^c}^d)^2 \frac{g_d^{(k)}(t)}{2 a_{u u_j^c}^{d,(k)}} + g_d^2(t) \frac{a_{u u_j^c}^{d,(k)}}{2 g_d^{(k)}(t)} \right) dt \right) \right] \\ &+ \alpha \left( \sum_d \mathcal{R}(g_d) + \sum_{u,u',d} (a_{uu'}^d)^2 \right), \quad (2)\end{aligned}$$

where $\tau_j^c = T_c - t_j^c$ and $p_{ij}^c$ and $p_{ii}^c$ are defined as follows:

$$\begin{aligned}p_{ii}^c &= \frac{\mu_{u_i^c}^{(k)}}{\mu_{u_i^c}^{(k)} + \sum_{j=1}^{i-1} \sum_d a_{u_i u_j}^{d,(k)} g_d^{(k)}(t_i^c - t_j^c)}, \\ p_{ijd}^c &= \frac{a_{u_i u_j}^{d,(k)} g_d^{(k)}(t_i^c - t_j^c)}{\mu_{u_i^c}^{(k)} + \sum_{j=1}^{i-1} \sum_d a_{u_i u_j}^{d,(k)} g_d^{(k)}(t_i^c - t_j^c)}.\end{aligned}$$

Intuitively, $p_{ijd}^c$ can be interpreted as the probability that the $i$-th event is influenced by a previous event $j$

through the $d$-th base kernel and $p_{ii}^c$ is the probability that $i$-th event is sampled from the base intensity. Thus, the first two terms of $Q(\Theta|\Theta^{(k)})$ can be viewed as the joint probability of the unknown influence structures and the observed events.

As is further shown in the Appendix, the following properties hold for $Q(\Theta;\Theta^{(k)})$ defined in Equation (2):

**1.** For all $\Theta$ and $\Theta^{(k)}$, $Q(\Theta;\Theta^{(k)}) \geq \mathcal{L}_\alpha(\Theta)$.

**2.** $Q(\Theta^{(k)};\Theta^{(k)}) = \mathcal{L}_\alpha(\Theta^{(k)})$.

The above two properties imply that if $\Theta^{(k+1)} = \operatorname{argmin}_\Theta Q(\Theta;\Theta^{(k)})$, we have $\mathcal{L}_\alpha(\Theta^{(k)}) \geq \mathcal{L}_\alpha(\Theta^{(k+1)})$. Thus, optimizing $Q$ with respect to $\Theta$ at each iteration ensures that the value of $\mathcal{L}_\alpha(\Theta)$ decrease monotonically.

**Update for $\mu_u$ and $a_{uu'}^d$.** Moreover, the advantage of optimizing $Q(\Theta|\Theta^{(k)})$ is that all parameters $g_d$ and $a_{uu'}^d$ can be solved independently from each other in closed form, and the non-negativity constraints are automatically taken care of. Specifically, we have the following update rules for $\mu_u$ and $a_{uu'}^d$:

$$\mu_u^{(k+1)} = \frac{1}{\sum_c T_c}\left(\sum_c \sum_{i=1,u_i^c=u}^{n_c} p_{ii}^c\right) \tag{3}$$

$$a_{uu'}^{d,(k+1)} = \left(\frac{a_{uu'}^{d,(k)} \sum_c \sum_{i:u_i^c=u} \sum_{j<i,u_j^c=u'} p_{ijd}^c}{\sum_c \sum_{j:u_j^c=u'} \int_0^{T_c-t_j^c} g_d^{(k)}(t)dt + \alpha}\right)^{\frac{1}{2}} \tag{4}$$

**Update for $g_d$.** The corresponding update for $g_d$ can be derived by optimizing in an infinite dimensional space. Specifically, for every $d = 1,\ldots,D$, we consider the terms in $Q(\Theta|\Theta^{(k)})$ that are related to $g_d$ as follows:

$$\min_{g_d \in L_1(\mathbb{R})} -\sum_c \left(\sum_{i=1}^{n_c}\sum_{j=1}^{i-1} p_{ijd}^c \log g_d(t_i^c - t_j^c) - \sum_{u=1}^{U}\sum_{j=1}^{n_c}\int_0^{T_c-t_i^c} g_d^2(t)\frac{a_{uu_j^c}^{d,(k)}}{2g_d^{(k)}(t)}dt\right) + \alpha\mathcal{R}(g_d). \tag{5}$$

The solution of the above minimization problem satisfies the Euler-Lagrange equation (see also Appendix):

$$-\frac{D(t)}{g_d(t)} + C(t)g_d(t) - 2\alpha g_d''(t) = 0, \tag{6}$$

where

$$C(t) = \sum_c \sum_{u=1}^{U}\sum_{j=1}^{n_c} \frac{a_{uu_j^c}^{d,(k)}}{g_d^{(k)}(t)}\mathbb{I}[t \leq T_c - t_i^c]$$

$$D(t) = \sum_c \sum_{i=1}^{n_c}\sum_{j=1}^{i-1} p_{ijd}^c \mathbb{I}[t = t_i^c - t_j^c],$$

**Algorithm 1** (MMEL) for estimating parameters

**Input:** Observed samples $\{c_1,\ldots,c_m\}$.
**Output:** Estimation of $\mu_u$, $a_{uu'}^d$ and $g_d$.
Initial $\mu_u$, $a_{uu'}$ and $g_d$ randomly.
**while** not converge **do**
  Update $\mu_u$, $a_{uu'}$ by Equation (3) and (4) for $u,u' = 1,\ldots,U$.
  **for** d=1,..., D **do**
    **while** not converge **do**
      Solve Equation (7) for $m = 1,2,\ldots,M$.
    **end while**
  **end for**
**end while**

where $\mathbb{I}[\cdot]$ is the indicator function which returns 1 if the predicate in parameter is true and 0 otherwise. We solve the above ODE numerically using the following Seidel type iterations which is quite efficient. Specifically, we discretized the differential equation over small intervals $m\Delta t$, for $m = 1,\ldots,M$, as follows:

$$-2\alpha\frac{g_{d,m+1} - 2g_{dm} + g_{d,m-1}}{\Delta t^2} + C_m g_{dm} = \frac{D_m}{g_{dm}}, \tag{7}$$

where $g_{dm} = g_d(m\Delta t)$ and

$$C_m = \frac{1}{g_d^{(k)}(m\Delta t)}\sum_c \sum_{u=1}^{U}\sum_{j=1}^{n_c} a_{uu_j^c}^{d,(k)}\mathbb{I}[m\Delta t \leq T_c - t_i^c]$$

$$D_m = \frac{1}{\Delta t}\sum_c \sum_{i,j:m\Delta t \leq t_i^c - t_j^c < (m+1)\Delta t} p_{ijd}^c$$

Therefore, we can solve for $g_{d,m}$ by fixing all other $g_{d,m'}$, $m' \neq m$ but solving the above quadratic equation. We summary the proposed algorithm in Algorithm 1.

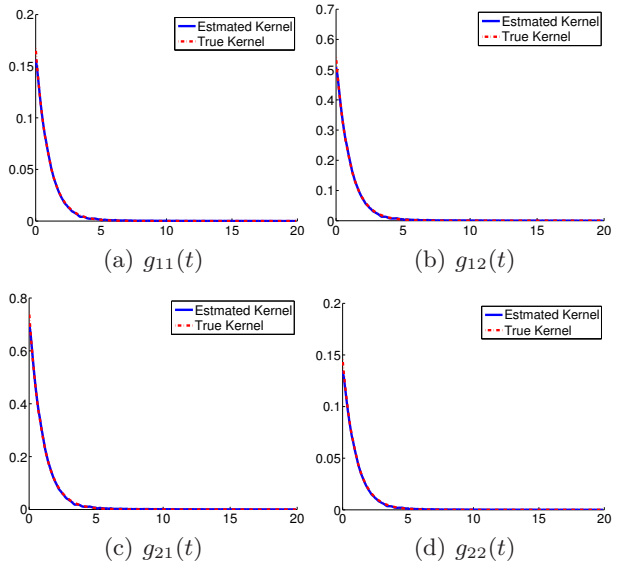
## 4. Experiments

In this section, we conduct experiments on both synthetic and real-world datasets to evaluate the performance of the proposed method MMEL.

### 4.1. Toy Data

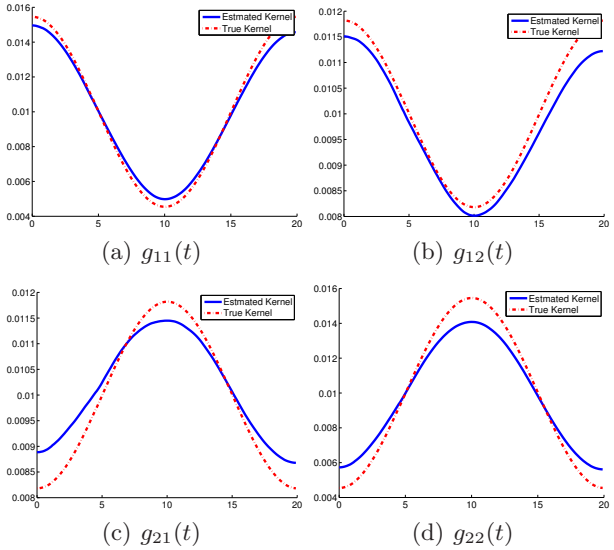
In order to illustrate that the proposed method can estimate the triggering kernels from data very accurately, we first conduct a set of experiments in toy data sets of 2-dimensional Hawkes processes. The goal is to visualize and compare the triggering kernels estimated from data to the ground-truth.

**Data Generation.** The true parameters of the 2-dimensional Hawkes process are generated as follows:




(a) $g_{11}(t)$ (b) $g_{12}(t)$ (c) $g_{21}(t)$ (d) $g_{22}(t)$

*Figure 1.* Estimated vs. True Triggering Kernels on toy data DataExp with exponential kernels. The four sub-figures show both the true triggering kernels and the estimated ones from the data




(a) $g_{11}(t)$ (b) $g_{12}(t)$ (c) $g_{21}(t)$ (d) $g_{22}(t)$

*Figure 2.* Estimated vs. True Triggering Kernels on toy data DataCos. The four sub-figures show both the true triggering kernels and the estimated ones from the data

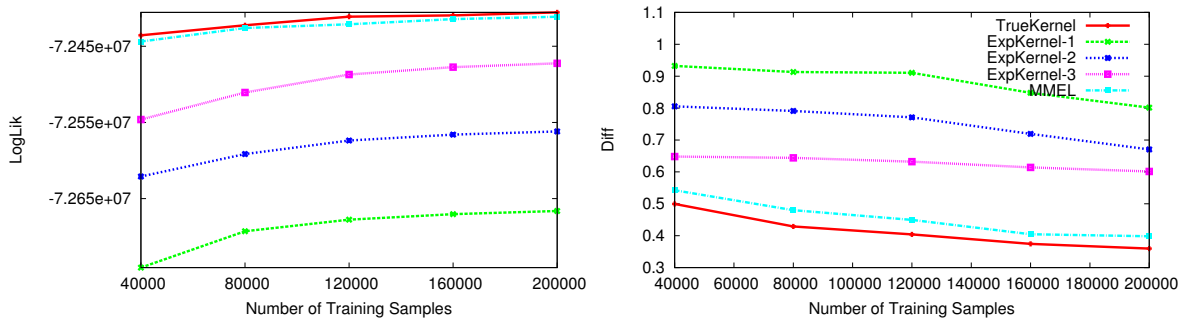
the base intensity $\mu_1 = \mu_2 = 0.1$. We generate two data sets with different triggering kernels: 1) DataExp. In the first data set, we use the exponential kernel $g(t) = \exp(-t)$, one of the most widely used trigger kernel for Hawkes processes, to demonstrate that the proposed algorithm can obtain good estimations in this case. 2) DataCos. In this case, we consider relatively complex kernels rather than simple exponential kernels. Specifically, in this case, the triggering kernels are generated by the linear combination of two base functions: $g_1(t) = \cos(\pi t/10) + 1.1$ and $g_2(t) = \cos(\pi(t/10 + 1)) + 1.1$. In both data sets, the coefficients of the linear combinations are generated from a uniform distribution on $[0.1, 0.2]$.

**Results.** We generate 100,000 samples from the two 2-dimensional Hawkes processes described above on time interval $[0, 20]$ as the training sets and run MMEL on the sampled data to obtain the estimations for the triggering kernels. In order to visualize the estimated kernels and compare them to the ground truth, we plot both the estimated and the true kernels for each pairs of dimensions. In Figure 1, we plot the triggering kernel obtained from the data by MMEL together with the true exponential kernel. We can observe that the estimated and true kernels almost overlap each other, which indicates that the estimated triggering kernels are very accurate in this case.

In Figure 2, we visualize the triggering kernels learnt from the DataCos data set. It can be observed that the proposed MMEL can reconstruct the kernels quite accurately from the data: The estimated kernel is very close to the true kernel in most places, although the estimated kernel may diverge from the ground-true in a few points. The proposed algorithm generally works quite well in this case since our method do not assume any parametric forms of the triggering kernels. Another observation is that MMEL tends to underestimate the triggering kernels at their peak points while overestimate them at valleys. In fact, similar bias exists in a lot of nonparametric estimators which is related to the curvature of the function. Several methods has been proposed to correct this problem (Sain and Scott, 2002). Thus, we leave this problem for future investigation.

### 4.2. Synthetic Data

**Data Generation.** A relatively large synthetic data set is generated as follows: We consider Hawkes processes of $U = 300$ dimensions with base intensity $\mu_u$ sampled from a uniform distribution over $[0, 0.001]$ for each $u$. The triggering kernels are the linear combinations of three base functions: $g_d(t) = \frac{\cos(2\pi t/w_d)+2}{t+2}$ where $w_d = 1, 3, 5$ for $d = 1, 2, 3$, respectively. The coefficients of the linear combinations are generated from a uniform distribution on $[0, 0.1]$. We generate 200,000 samples from the multi-dimensional Hawkes process as training set and another disjoint 200,000 samples as test set. We run the above process for five times and the performance are reported by the average




(a) LogLik (b) Diff

*Figure 3.* Performance measured by LogLik and Diff with respect to the number of training samples on the synthetic data.

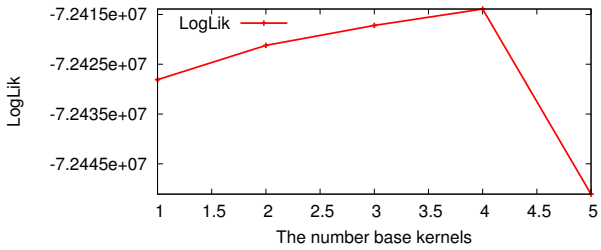



*Figure 4.* Performance measured by LogLik with respect to number of base kernels $D$.

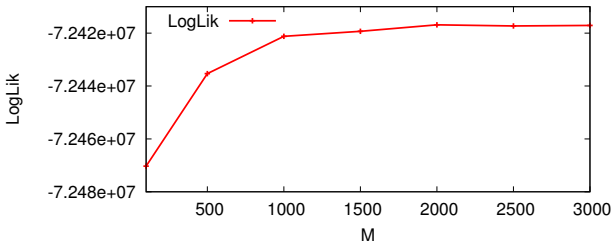



*Figure 5.* Performance measured by LogLik with respect to point used to discretize the ODE.

over five data sets.

**Evaluation Metrics.** We use two evaluation metrics LogLik and Diff to evaluate the proposed method. Specifically, LogLik is defined as the log-likelihood on the test set of 200,000 samples that is disjoint with the training set. Diff measures the relative distances between the estimated and true kernels as follows:

$$\text{diff} = \frac{1}{U^2}\sum_{u=1}^{U}\sum_{u'=1}^{U}\frac{\int(\hat{g}_{uu'}(t) - g_{uu'}(t))^2 dt}{\int g_{uu'}(t)^2 dt},$$

where $\hat{g}_{uu'}(t)$ and $g_{uu'}(t)$ are the estimated and true kernels between dimension $u$ and $u'$, respectively.

**Baselines.** We compare the proposed MMEL with the following baselines to demonstrate its effectiveness:

- ExpKernel. In this method, the triggering kernel is assumed to be fixed to be the exponential kernel $g(t) = \frac{1}{w}\exp(-t/w)$ which is one of the most widely used kernels for multi-dimensional Hawkes process. In this work, we use $w = 1, 3, 5$ as baselines and label them as ExpKernel-1, ExpKernel-3 and ExpKernel-5, respectively.

- TrueKernel. In this method, we assume that the bases $g_d(t)$ used to generate the data are known and fixed. Only the coefficients are estimated from the data. This method is used as an upper-bound to show that how the proposed MMEL algorithm can perform in the ideal situation that the true bases for the triggering kernels are known.

**Results.** We train models with MMEL as well as the baselines on the training set. For MMEL, we use regularization parameter $\alpha = 1000$ and discretize the ODE with $M = 3000$ intervals. In Figure 3, we present the performance on the synthetic data set with respect to the number of training data. From Figure 3, we can observe that the performance of all methods improves as the number of training data grows, which indicates that more training data can improve the performance. Comparing the performance of methods using exponential kernels, i.e., ExpKernel-1, ExpKernel-3 and ExpKernel-5, we can observe that the selection of the parameters for the exponential kernel can greatly impact the performance, which confirms that the triggering kernels plays a central role in multi-dimensional Hawkes processes.

The proposed method MMEL performs significantly better than the method using exponential kernels with respect to both metrics. Moreover, its performance is very close to TrueKernel, the method that fixes the base kernels to be the ground-truth and does not estimate them from data. Therefore, we conclude that the MMEL can estimate the triggering kernels very accurately.

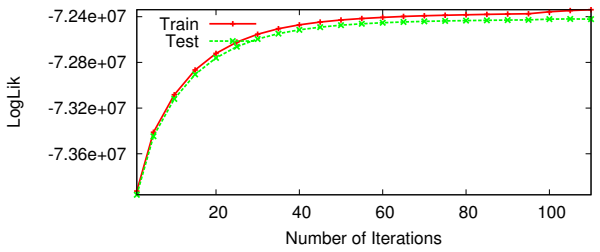



*Figure 6.* Performance measured by LogLik with respect to number of iterations.

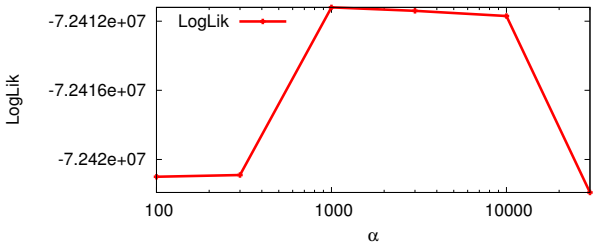



*Figure 7.* Performance measured by LogLik with respect to regularization parameter $\alpha$.

In Figure 4, we present the performance of MMEL measured by LogLik with respect to the number of base kernels. In our previous experiments on the synthetic data set, we set the number of base kernel to be 3, which is the true value used to generate the data. It is interesting to observe that slightly larger number of base kernels such as 4 can archive better performance.

In Figure 5, we investigate the number of intervals $M$ used to discretize the ordinary differential equation in Equation (6). In particular, we vary $M$ in range of 100 to 3000 and plot the performance measured by LogLik. From Figure 5, we can see that when the number of intervals is relatively large ($\geq 1000$), the performance is quite good and stable.

In Figure 6, we show that the performance measured by LogLik on both training and test sets with respect to the number of iterations of MMEL. It can be observed that the performance on both training and test sets increases as the number of iterations grows and converges after 100 iterations. In Figure 7, we present the performance measured by LogLik on the test set with respect to the value of regularization parameter $\alpha$. We can observe that small values of $\alpha$ usually reduce the performance, while the performance is quite stable for $\alpha$ between $[1000, 10000]$.

### 4.3. Real World Data

We also evaluate the proposed method on a real world data set. To this end, we use the MemeTracker data set[1]. The data set contains the information flows captured by hyper-links between different sites with timestamps. In particular, we first extract the top 100 popular sites and the links between them. The events are in the form that a site created a hyper-link to another site at a particular time. We use 50% data as training data and 50% as test data. In this data set, we use $D = 1$ and $\alpha = 10$.

In Figure 8, we present the performance measured by the negative log-likelihood on test set for MMEL, ExpKernel-1, ExpKernel-3 and ExpKernel-5 on the MemeTracker data set. We can observe that MMEL outperforms all the baselines, which indicates that MMEL can capture the dynamics of the temporal events more accurately. We also tried to fix the kernel to be other forms such as cosine functions. The performance is much worse, which suggests the importance of the triggering kernel. In order to further investigate the performance, we transform the events by the fitted model based on the time rescaling theorem (Papangelou, 1972), and generate the quantile-quantile (Q-Q) plot with respect to the exponential distribution, since it is the theoretical distribution for the perfect model of intensity functions as shown by the theorem. Generally speaking, Q-Q plot visualizes the goodness-of-fit for different models. The perfect model follows a straight line of $y = x$. In Figure 9, we present the Q-Q plot for MMEL, ExpKernel-1 and Poisson, which is the Poisson process model with constant intensity. We can observe that MMEL are generally closer to the straight line, which suggests that MMEL can fit the data better than other models.

In Figure 10, we plot the base kernel estimated from the data by MMEL. The base kernel has quite intuitive in the sense that in the first several days, the estimated base kernel has relatively large values, which can be explained by the fact that new blogs or webpages are more likely to be related to hot topics and thus are more likely to trigger further discussions. The base has relatively small values at almost all other points, except for two small peaks as we can observe in the figure. We think it reflects the long-term discussions of some topics.

## 5. Conclusions

In this paper, we address the problem of learning the triggering kernels, which capture the underlying temporal dynamics of the observed events, for multi-dimensional Hawkes processes. In particular, we estimate the triggering kernels from the data through optimizing the penalized log-likelihood function in infi-

[1] http://memetracker.org

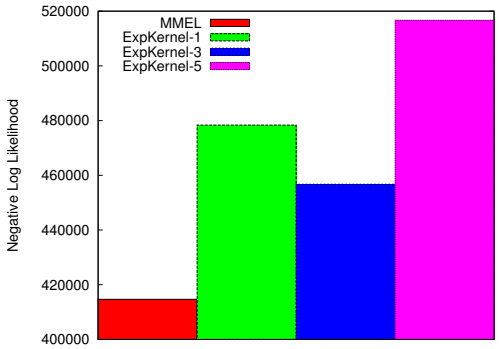



*Figure 8.* Performance measured by negative log-likelihood on MemeTracker data set.

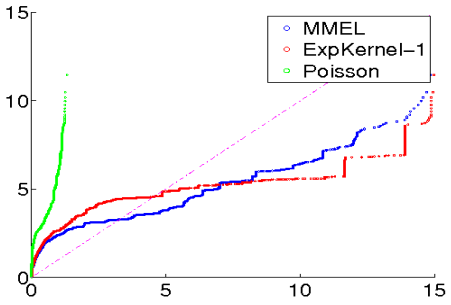



*Figure 9.* Q-Q plot for comparing the transformed events with respect to the exponential distribution.

nite dimensional spaces. An iterative algorithm MMEL is proposed to optimized the penalized log-likelihood function effectively. The optimization problem in infinite dimensional functional space is transformed to solving an ordinary differential equation which is derived from the Euler-Lagrange equation. Experimental results on both synthesis and real-world data set show that the proposed method can estimate the triggering kernels more accurately and thus provide better models for recurrent events.

There are several directions that are interesting for further investigation: First, we can extend this work to more general case of the spatial-temporal process, where the triggering kernel defined on multi-dimensional spaces rather than 1-dimensional real lines. Moreover, we plan to study more applications for the proposed method by considering different constraints on the triggering kernel, e.g., monotonic constraints.

## Acknowledgements

Part of the work is supported by NSF IIS-1116886, NSF IIS-1218749 and a DARPA Xdata grant.

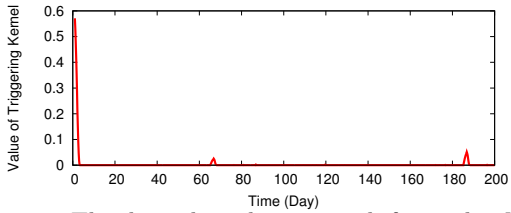



*Figure 10.* The base kernel estimated from the MemeTracker data set.

## Appendix

**Majorization Minmization.** First, we claim that the following properties hold for $Q(\Theta;\Theta^{(k)})$ defined in Equation (2) :

**1.** For all $\Theta$ and $\Theta^{(k)}$, $Q(\Theta;\Theta^{(k)}) \geq \mathcal{L}_\alpha(\Theta)$.

**2.** $Q(\Theta^{(k)};\Theta^{(k)}) = \mathcal{L}_\alpha(\Theta^{(k)})$.

*Proof.* The first claim can be shown by utilizing the Jensen's inequality: For all $c$ and $i$, we have

$$\log(\mu_{u_i^c} + \sum_{j=1}^{i-1}\sum_{d=1}^{D} a^d_{u_i^c u_j^c} g_d(t_i^c - t_j^c))$$

$$\geq p^c_{ii} \log \frac{u_i^c}{p^c_{ii}} + \sum_{j=1}^{i-1}\sum_{d=1}^{D} p^c_{ijd} \frac{a^d_{u_i^c u_j^c} g_d(t_i^c - t_j^c)}{p^c_{ijd}}$$

Moreover, by the inequality of arithmetic and geometric means:

$$(a^d_{uu_j^c})^2 \frac{g_d^{(k)}(t)}{2a^{d,(k)}_{uu_j^c}} + g_d^2(t) \frac{a^{d,(k)}_{uu_j^c}}{2g_d^{(k)}(t)} \geq a^d_{uu_j^c} g_d(t)$$

By noting that summation and integration preserve the above two inequalities, we prove the first claim.

The second claim can be checked by setting $g_d(t) = g_d^{(k)}(t)$, $a^d_{uu'} = a^{d,(k)}_{uu'}$ and $\mu_u = \mu_u^{(k)}$. □

**Euler-Lagrange Equation.** The optimization problem in Equation (5) is equivalent to minimize $L[g, g'] = \int_0^\infty F(g, g')dt$, where

$$F(g,g') = -\sum_c \sum_{i=1}^{n_c}\sum_{j=1}^{i-1} p^c_{ijd} \log g_d(t)\mathbb{I}[t = t_i^c - t_j^c]$$

$$+ \sum_c \sum_{u=1}^{U}\sum_{j=1}^{n_c} g_d^2(t) \frac{a^{d,(k)}_{uu_j^c}}{2g_d^{(k)}(t)} \mathbb{I}[t \leq T_c - t_i^c] + \alpha_2 (g_d'(t))^2$$

By Euler-Lagrange equation, the solution satifies

$$\frac{\partial F}{\partial g_d} - \frac{d}{dt}[\frac{\partial F}{\partial g_d'}] = 0$$

Substitute $F$ into the above equation, we get the ordinary differential equation for solving $g_d(t)$ in Equation (6).

## References


Argyriou, A., Hauser, R., Micchelli, C. A., and Pontil, M. (2006). A DC-programming algorithm for kernel selection. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 41–48, New York, New York, USA. ACM Press.

Bach, F. (2008). Exploring Large Feature Spaces with Hierarchical Multiple Kernel Learning. *Advances in Neural Information Processing Systems (NIPS)*, 21(2):105–112.

Bacry, E., Dayri, K., and Muzy, J. F. (2012). Non-parametric kernel estimation for symmetric Hawkes processes. Application to high frequency financial data. *The European Physical Journal B*, 85(5).

Blundell, C., Heller, K., and Beck, J. (2012). Modelling Reciprocating Relationships with Hawkes Processes. *Advances in Neural Information Processing Systems (NIPS)*.

Cortes, C., Mohri, M., and Rostamizadeh, A. (2012). Ensembles of Kernel Predictors.

Dinuzzo, F., Ong, C., Gehler, P., and Pillonetto., G. (2011). Learning output kernels with block coordinate descent. In *Proceedings of the 28th International Conference on Machine Learning*.

Du, N., Song, L., Smola, A., and Yuan, M. (2012). Learning Networks of Heterogeneous Influence. *Advances in Neural Information Processing Systems*.

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

Hoi, S. C. H., Jin, R., and Lyu, M. R. (2007). Learning nonparametric kernel matrices from pairwise constraints. In *Proceedings of the 24th international conference on Machine learning - ICML '07*, pages 361–368, New York, New York, USA. ACM Press.

Hunter, D. R. and Lange, K. (2003). A Tutorial on MM Algorithms. pages 1–28.

Lewis, E. and Mohler, G. (2011). A Nonparametric EM algorithm for Multiscale Hawkes Processes. *Journal of Nonparametric Statistics*, (1).

Lewis, E., Mohler, G., Brantingham, P. J., and Bertozzi, A. L. (2011). Self-exciting point process models of civilian deaths in Iraq. *Security Journal*, 25(3):244–264.

Liniger, T. J. (2009). *Multivariate Hawkes Processes*. PhD thesis, Swiss Federal Institute Of Technology Zurich.

Mammen, E. and Thomas-agnan, C. (1998). Smoothing Splines And Shape Restrictions. *Scandinavian Journal of Statistics*, 26:239–251.

Marsan, D. and Lengliné, O. (2008). Extending earthquakes' reach through cascading. *Science*, 319(5866):1076–9.

Mitchell, L. and Cates, M. E. (2010). Hawkes process as a model of social interactions: a view on video dynamics. *Journal of Physics A: Mathematical and Theoretical*, 43(4):045101.

Papangelou, F. (1972). Integrability of Expected Increments of Point Processes and a Related Random Change of Scale. *Transactions of the American Mathematical Society*, 165:483.

Reinsch, C. h. (1967). Smoothing by Spline Functions. *Numerische Mathematik*, pages 177–183.

Sain, S. R. and Scott, D. W. (2002). Zero-bias locally adaptive density estimators. *Scandinavian Journal of Statistics*, 29(3):441–460.

Simma, A. and Jordan, M. (2012). Modeling events with cascades of Poisson processes. *Uncertainty in Artificial Intelligence (UAI)*.

Sonnenburg, S., Raetsch, G., Schaefer, C., and Schölkopf, B. (2006). Large scale multiple kernel learning. *Journal of Machine Learning Research*, 7(1):1531–1565.

Stomakhin, A., Short, M. B., and Bertozzi, A. L. (2011). Reconstruction of missing data in social networks based on temporal patterns of interactions. *Inverse Problems*, 27(11):115013.

Toke, I. M. (2010). "Market making" behaviour in an order book model and its impact on the bid-ask spread. *Arxiv*, page 17.

Turlach, B. A. (1997). *Constrained Smoothing Splines Revisited*. PhD thesis, Australian National University.

Vere-Jones, D. (1970). Stochastic Models for Earthquake Occurrence. *Journal of the Royal Statistical Society*, 32(1):1–62.

Wahba, G. (1990). *Spline models for observational data*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

Zhu, X., Kandola, J. S., Ghahramani, Z., and Lafferty, J. D. (2004). Nonparametric Transforms of Graph Kernels for Semi-Supervised Learning. *Advances in Neural Information Processing Systems (NIPS)*.