Analysis of wireless networks using Hawkes processes

Michael G. Moore and Mark A. Davenport
Georgia Institute of Technology
School of Electrical and Computer Engineering
{mmoore90,mdav}@gatech.edu

***Abstract*—In this paper we consider the problem of characterizing and analyzing a wireless network from limited passive observations of network activity. In particular, we will assume that the only information that we can acquire is knowledge of when each particular transmitter in the network initiates any given transmission. From this data, we wish to be able to solve problems such as learning the network topology, detecting changes to the existing topology, and extracting higher-level summaries of information flow in the network. We show how one can use a multidimensional autoregressive point process known as a *Hawkes process* to model the observed data and approach these problems.**

I. Introduction

A fundamental problem in network monitoring consists of characterizing and analyzing the structure of wireless networks. There are a variety of techniques for approaching these problems as an observer *within* the network, but in many important applications we are limited to the role of a passive observer *outside* the network, meaning that we cannot directly discover the details of network traffic (message content, routing information, etc.). This may arise in cases where traffic is encrypted or otherwise unavailable, or where there is simply too much traffic to process. These scenarios are particularly common in signals intelligence and electronic warfare applications, but arise in other applications as well.

Our focus in this paper is on how we can solve problems such as learning the network topology, detecting changes to the existing topology, and extracting high-level summaries of information flow in the network under the assumption that the only information we can observe is when each transmitter in the network initiates a transmission. While learning from this limited source of data might seem difficult, we will see that a significant amount of information can be extracted by exploiting the fact that communication is typically *reciprocal*. That is, a transmission from a particular transmitter is likely to cause other transmissions in response (such as return messages, acknowledgments, packet forwarding, etc.).

With this assumption, we can use the co-occurrence of transmissions from different transmitters to infer their relationships by modeling the data as a multidimensional autoregressive point process known as a *Hawkes process*. This approach relies on the assumption that we are able to accurately determine who is transmitting at any given time. In practice, this can be achieved through geolocation, specific emitter identification, and other content-agnostic features.

II. Hawkes processes

Hawkes processes are autoregressive point processes that provide a powerful method for inferring connections between nodes based on when events are observed. Initially developed in the context of modeling earthquake occurrence [1], they have since been used for studying relations in social [2], neural [3], financial [4], epidemic [5], and other varieties of networks.

Here we provide a brief introduction to Hawkes processes and how they can be used to model activity in a network with a given structure.

A. One-dimensional Hawkes processes

We begin by considering a one-dimensional Hawkes process, which is simply a point process where the conditional intensity function (CIF) – a rate parameter denoted by $\lambda(t)$ – depends on the history (previous event times) of the process:

$$\lambda(t) = \mu(t) + A \sum_{k=1}^{K} \gamma(t - t_k). \tag{1}$$

The set of times at which events occur is $\{t_k\}_{k=1}^{K}$. The parameter $\mu(t) \geq 0$ is the base rate for the process, which can potentially vary as a function of time. In this paper we will restrict our attention to the case where $\mu(t)$ is constant, so from this point onward we will omit its dependence on t. The parameter $A \geq 0$ represents the tendency of the process to *self-excite*, with larger A resulting in a process that produces clusters of events. The kernel $\gamma(t)$ represents the temporal relationship between events and responses. It should be causal ($\gamma(t) = 0$ for $t \leq 0$), nonnegative, and integrable.[1]

Without loss of generality, we will require that $\int_0^\infty \gamma(t)dt = 1$. With this definition, A can be interpreted as the expected number of events occurring in response to a previous event. Note that when $A = 0$ this reduces to a Poisson process. For $A \geq 1$, the process is *unstable* in the sense that the event rate will increase without bound because each event will produce (in expectation) one or more additional events.

B. Multidimensional Hawkes processes

This model is easily extended to the case of a collection of many related subprocesses by letting the rate of each

[1]While it is not strictly necessary to constrain ourselves to the case of nonnegative μ, A, or $\gamma(t)$, we choose to because doing otherwise requires careful treatment to avoid or otherwise handle cases where $\lambda(t) < 0$.

978-1-5090-1749-2/16/$31.00 ©2016 IEEE

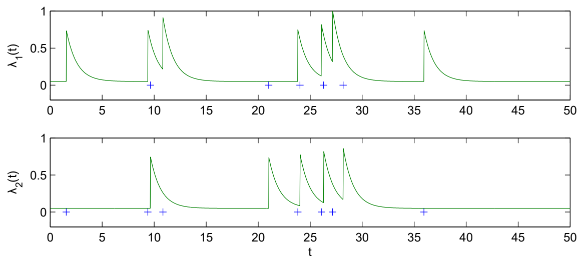

Fig. 1. Realization of a two-dimensional Hawkes process where each subprocess is only excited by the other (the influence matrix is anti-diagonal). Each $+$ indicates an event. Notice how an event in one subprocess leads to an increase in the CIF for the other, which often leads to clusters of activity.

subprocess be a function not only of its own history, but of the other subprocesses' histories as well. We can use such *multi-dimensional Hawkes processes* to model a wireless networks by viewing transmissions as the events of the process, where each event is assigned to a subprocess based on the transmitter from which the transmission originated. For N subprocesses, the CIF of the i^{th} subprocess generalizes (1) to

$$\lambda_i(t) = \mu_i + \sum_{j=1}^{N} A_{ij} \sum_{k \in K_j} \gamma(t - t_k) \tag{2}$$

where K_j is the set of events on subprocess j. In this model, each subprocess may have its own base rate μ_i and each nonnegative parameter A_{ij} quantifies how much subprocess i reacts to subprocess j. We can think of A, sometimes called the *influence matrix* or *infectivity matrix*, as a representation of the topology of our network in the form of a weighted adjacency matrix, where larger values indicate a stronger connection. In our specific context, $A_{ij} > 0$ means that a transmission from transmitter j creates a temporary increase in the probability of a transmission from transmitter i. This results in small clusters of activity or "conversations" between these transmitters. An example that illustrates this phenomenon is given in Figure 1.

The stability condition generalizes naturally to the multidimensional case. The event rate can increase without bound if the spectral radius $\rho(A) \geq 1$. Constraining $\rho(A) < 1$ ensures stability, in which case the average event rate converges to $\bar{\lambda} = (I - A)^{-1}\mu$.[2]

III. Parameter Inference

In many scenarios of interest, the parameters μ and A will not be known *a priori*. Thus, if we are given a set of observations (i.e., a list of transmission times for each transmitter), we would like to estimate the parameters that best agree with this data.

[2]The asymptotic average event rate must satisfy $\bar{\lambda} = A\bar{\lambda} + \mu$ for some $\bar{\lambda} \geq 0$. A solution always exists when $\rho(A) < 1$. Even when a solution exists for $\rho(A) \geq 1$, process variability will usually excite unstable modes.

A. Maximum-likelihood inference

A natural approach is to use maximum-likelihood estimation to infer the parameter values. The negative log-likelihood of a set of event observations in interval $\mathcal{T}$ on process i is

$$\mathcal{L}_i(\mu, A | \{t\}) = \int_{\mathcal{T}} \lambda_i(t) dt - \sum_{\substack{k \in K_i \\ t_k \in \mathcal{T}}} \log \lambda_i(t_k). \tag{3}$$

For the ensemble of subprocesses in a multidimensional Hawkes process, the negative log-likelihood is simply the summation of (3) across all N subprocesses:

$$\mathcal{L}(\mu, A | \{t\}) = \sum_{i=1}^{N} \mathcal{L}_i(\mu, A | \{t\}). \tag{4}$$

Because $\lambda_i(t)$ depends only on μ_i and the i^{th} row of A, we can optimize each subprocess likelihood independently. This divides the optimization program minimizing (4) (which optimizes over $N(N+1)$ parameters) into N independent subproblems of the form in (3) (each optimizing over only $N+1$ parameters). Each subproblem is a convex optimization and is thus relatively straightforward to solve. One can readily apply existing methods such as SPIRAL [6] or composite self-concordant minimization [7]. The authors use an alternative algorithm, consisting of projected quasi-Newton descent with a backtracking line search (for step size) and where the Hessian is approximated by its diagonal.

B. Inference accuracy

An important question when performing inference concerns the number of events we must observe in order to learn the parameters to a given accuracy. While precise theoretical bounds addressing this question are not yet known, we can share some empirical findings that will at least provide rough guidelines as to how many observations may be necessary for accurate inference in practice.

Clearly, in order to discover a connection between two elements in the network (i.e., a nonzero entry in A), we must see it used at least once. Thus, if there are r nonzeros in A, then the *coupon collector* problem suggests we will require at least $O(r \log r)$ events to observe all connections "in action" at least once. More realistically, we will want to see connections used several times so that we can deal with ambiguous cases and avoid false-positives. In other words, we expect the number of events necessary to accurately recover the locations of the nonzeros in A to be slightly super-linear in the number of nonzeros.

Figure 2 provides evidence of this behavior [8]. We consider an 80 node network where bidirectional interconnections are randomly added such that the expected total number of connections in the network is r. Each connection has the same strength, which is chosen so that the Hawkes model has a stable event rate ($\rho(A) < 1$). Events are generated using a multidimensional Hawkes process with these parameters and we then compute the maximum likelihood estimates $\widehat{\mu}$ and $\widehat{A}$ by minimizing (4). We threshold the resulting entries of $\widehat{A}$

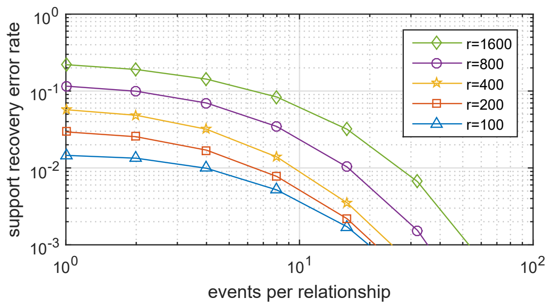

Fig. 2. Support recovery rate as a function of the number of events observed per relationship. Each line denotes a different number of relationships (nonzeros) in the 80×80 influence matrix.

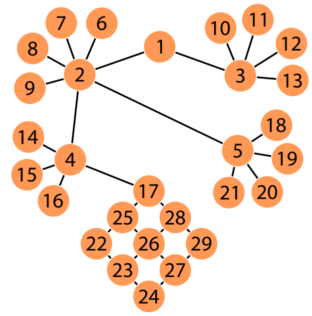

Fig. 3. Connections within the simulated EMANE network. Some links are not utilized in the simulation.

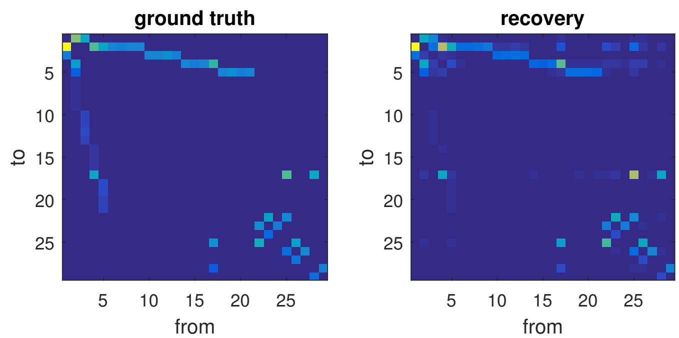

Fig. 4. Ground truth and recovered influence of EMANE-simulated data.

and then compute the error rate in estimating the locations of the nonzeros (the *support recovery error rate*). This is repeated for many trials for each r, and number of observations and the error rates are averaged and reported in the figure. The horizontal axis is displayed in terms of the number of events observed *divided by* r. The results of the simulation are consistent with our intuition that the number of events is weakly super-linear in r (something like $O(r \operatorname{polylog} r)$).

There are a number of additional factors that likely play into the scaling, as well. As the overall event rate increases, greater ambiguity arises as it becomes more difficult to determine which subprocess is responsible for exciting another. This can eventually lead to a dramatic increase in the number of events required to perform inference. Other factors that can affect this requirement include the dynamic range of the entries of A and model mismatch between the Hawkes process and the data.

IV. Application to Wireless Networks

In practice, we do not expect the data in a real-world network to truly follow a Hawkes process. For example, there will typically be additional structure in the data not captured by the Hawkes model. However, a Hawkes process does capture the "reciprocal" nature of the interactions we would expect to observe and so, despite some degree of model mismatch, it can be useful as an inference tool. Here we provide a brief demonstration of the use of Hawkes processes applied to realistic network communication data.

The data we will use is a trace created by the EMANE network emulator.[3] The network used consists of 29 interconnected nodes, arranged as in Figure 3, transmitting a total of over 2.2 million packets. Packets are generated and then propagated along the network from node to node to reach their destination. We then strip the trace of all routing information, except for the transmitter of each packet, to provide a simulation of the kind of data we would be able to observe in a typical wireless network surveillance scenario.

Maximum likelihood Hawkes parameters are estimated by minimizing (4). We use an exponential decay kernel for $\gamma(t)$, though typically the exact shape of the kernel has a limited impact relative to more general features such as its duration. We find it useful to impose the constraint $A_{ii} = 0$, since we have no reason to believe that a radio will transmit information in response to its own previous transmissions.

The notion is that nodes which appear to influence each other strongly are likely connected. The ground truth and the recovered influence are shown in Figure 4. While a relatively small number of false-positives have persisted and some of the weaker links have been missed, all of the strong connections within the network have been recovered. We emphasize that, even though the data was not generated according to a Hawkes process, this model was able to exploit the reciprocity in the network to discover most of the connections. Furthermore, it has estimated the strength of those connections.

V. Beyond Network Recovery

We have shown how to use Hawkes processes to discover connections in a network. There are a number of ways to extend the techniques described above to solve other specific problems that arise in the context of monitoring wireless networks. We present a selection of such extensions below.

A. Detecting changes in the network

A benefit of using Hawkes processes to model data is that they provide a means to asses the plausibility of an observation. Using this, we can determine whether sets of data are drawn from the same distribution. In the specific context of wireless networks, we can determine if two sets of observations are plausible under the same network topology. This is useful if we wish to recognize when the topology of the network changes.

[3] http://www.nrl.navy.mil/itd/ncs/products/emane

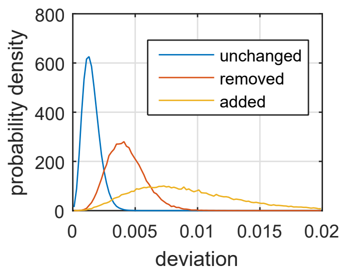

Fig. 5. Probability density functions showing how likelihood deviation can change when single connections are added to or removed from nodes in a network. Unchanged connections result in little deviation while changed, especially added, connections results in significant deviation.

Toward this end, one possible measure of deviation between two sets of data is the log-likelihood distance. We define $\{\widehat{\mu}_1, \widehat{A}_1\}$ and $\{\widehat{\mu}_2, \widehat{A}_2\}$ to be the parameters that minimize (3) for the two sets of observations $\{t\}_1$ and $\{t\}_2$. We then define the deviation of observations $\{t\}_2$ from the model suggested by $\{t\}_1$ (with regard to subprocess i) to be

$$d_i(\{t\}_1, \{t\}_2) = \frac{\mathcal{L}_i(\widehat{\mu}_1, \widehat{A}_1 | \{t\}_2) - \mathcal{L}_i(\widehat{\mu}_2, \widehat{A}_2 | \{t\}_2)}{|\{t\}_2|}. \quad (5)$$

This expression is always nonnegative because $\{\widehat{\mu}_2, \widehat{A}_2\}$ minimizes (3) given $\{t\}_2$. The denominator serves to appropriately scale the deviation to account for the negative log-likelihood scaling linearly with the number of events.

This metric can aid in detecting changes in network topology. As an example, we consider a 50-node network modeled by a multidimensional Hawkes process where the matrix A is binary-valued (scaled to maintain stability) with 200 nonzeros that are uniformly selected at random. We assume that we know the base parameters exactly (i.e., we are given many observations to estimate μ and A). We then modify 10% of the nodes by adding an additional connection (adding a nonzero to the corresponding row of A) and a different 10% by subtracting a connection. We simulate 5000 events (25 per connection) using these new parameters and we call the resulting observations $\{t\}_2$. With an abuse of the notation of (5), we calculate $d_i(\{\mu, A\}, \{t_2\})$, the change in the likelihood of the observed events when using the MLE parameters versus our prior model. We categorize nodes as either unchanged (having the same links in both the base and modified processes) or as nodes with a removed or added connection, and examine the distribution of the d_i's for each group.

The distributions of the d_i's, estimated from thousands of trials, are presented in Figure 5. It is relatively easy to recognize when a new connection is added (the "added" distribution has little overlap with the "unchanged" distribution) but it is somewhat more challenging to recognize when a connection is removed. Intuitively, this phenomenon can be explained by the fact that new connections can be identified quickly when unexpected transmissions occur. A missing connection requires detection of the more subtle absence of transmissions that we would expect to see. In both cases, the modified distributions deviate further from the unchanged distribution as more events are observed, making the discrimination easier.

B. *Incorporating additional structure via marks*

The Hawkes process model we have considered up to this point captures the "conversational" aspect of typical network interactions. However, it may still be a poor representation of actual behavior in some specific networks. For example, cell phones do not exhibit a back-and-forth transmission pattern (like a push-to-talk or packet radio network) but instead establish one enduring transmission (or continuous string of transmissions) for the entire interaction. We may further expect that transmission lengths will be comparable between two interacting nodes. The simple multidimensional Hawkes process described earlier cannot model this additional structure.

We can incorporate such details into the model via the addition of *marks* – additional information corresponding to each event – to a Hawkes process. Marks are drawn from some distribution and can depend on other event times and marks. In fact, the multidimensional Hawkes process can be viewed as a one-dimensional marked Hawkes process. In that interpretation, the CIF of the process is $\lambda(t) = \sum_i \lambda_i(t)$ and the marks are the set memberships $k \in K_i$.

As a more elaborate example, we will describe a model that better resembles cellular traffic. Assume that we can observe the start and end times of calls on every phone, but not with whom they are speaking (we will ignore base stations because they convey the same call start/stop information). For this example, let us assume that calls have a duration with probability density function $\beta(t)$, that a phone that receives a call will answer within 20 seconds, and that both phones will terminate the call within 5 seconds of each other. We will use t_k to indicate the start time of a call and v_k to denote the end time. A CIF that incorporates this assumption might be

$$\lambda_i(t, v) = \mu_i \beta(v-t) + \sum_j \frac{A_{ij}}{20} \sum_{k \in K_j} \mathcal{I}_{(0,20)}(t-t_k) \mathcal{I}_{(-5,5)}(v-v_k)$$

where $\mathcal{I}_S(t)$ is 1 when $t \in S$ and 0 otherwise.

This model, while a closer match than a standard Hawkes process, still has some limitations. It theoretically allows for many phones to engage in the same call and allows individual phones to be engaged in multiple calls at once (or even the same call multiple times). It also doesn't account for missed calls or a number of other scenarios. But while this model may be lacking in a generative sense, when we use it as an inferential model we do not necessarily need to be concerned by such cases since they do not typically arise in observed data. What this model *does* impose (in the context of inference) is the assumption that if two phones do not have start and end times closely matched then they are very unlikely to be involved in the same call. Incorporating this additional assumption can dramatically improve the quality of inference.

While the base model for a Hawkes process may not perfectly match some applications, marks provide a powerful technique to extend it. With creativity, one can produce models that impose a variety of additional structural assumptions.

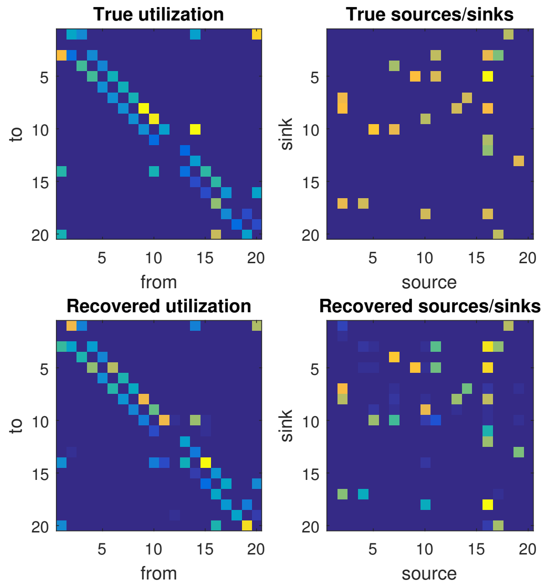

Fig. 6. The utilization of individual links in a network may not always be the information we are after. For example, we might be more interested in tracing the paths of packets through the network to find source/sink pairs.

C. Identifying chains of events and higher-level structure

Up to this point, we have focused on learning about the basic connectivity structure of networks. However, this information may not be exactly what we are interested in learning. For example, in an ad-hoc wireless network we may be more interested in differentiating between the source/destination of a particular transmission and transmitters that are simply acting as relays. In order to identify this kind of higher-level detail about the flow of information through the network, it would be extremely helpful to be able to determine the relationship between events, in order to associate related events. This information can then be used to discover large-scale structure in the network (i.e., who is ultimately communicating with whom), or to discover key nodes that act as relays.

We can do this using Hawkes processes via some simple post-processing. As described more fully in [9], this can be accomplished by first estimating the probability that pairs of events are related. This is enabled by the insight that the probability that event k from subprocess j resulted in event ℓ on subprocess i occurring is

$$\mathbb{P}(k \to \ell | k \in K_j, \ell \in K_i) = \frac{A_{ij}\gamma(t_\ell - t_k)}{\lambda_i(t_\ell)}. \qquad (6)$$

Once these inter-event influences are computed, a dynamic program can assemble related pairs into longer chains of events. In this way, it is possible to discover when two nodes communicate using one or more intermediate nodes as relays.

As an example, Figure 6 shows a network where we can easily recover the connectivity pattern using a Hawkes process model. However, we do not discover the underlying structure of sources, sinks, and relays in the network until we determine the influence individual events have on each other.

VI. Conclusions

We have presented a number of techniques and observations that may be helpful when modeling activity patterns in wireless networks using Hawkes processes. This convenient statistical model captures the behaviors common in communication networks and allows for useful inferences to be made from limited information.

There are a variety of additional innovations not discussed here. For example, there is no need to restrict ourselves to a single kernel or a single influence matrix, additional terms (involving different kernels and/or matrices) can be added without affecting the convexity of the inference program. These additional kernels and coefficients can dramatically enhance the representative power of Hawkes models. It is even possible to learn kernels from the data by inferring different weights at different delays [10]. We leave the exploration of such refinements in this context to future work.

MATLAB code that implements much of the functionality described in this paper is available for download at http://users.ece.gatech.edu/∼mdavenport/software/.

Acknowledgments

This work was supported by grants NRL N00173-14-2-C001, AFOSR FA9550-14-1-0342, NSF CCF-1350616, CCF-1409406, and CMMI-1537261. The authors would like to thank Crystal Acosta and Silvija Kokalj-Filipovic for many helpful conversations.

References

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

[2] C. Blundell, J. Beck, and K. Heller, "Modelling Reciprocating Relationships with Hawkes Processes," in *Proc. Adv. in Neural Processing Systems (NIPS)*. Lake Tahoe, NV, Dec. 2012.

[3] P. Reynaud-Bouret, V. Rivoirard, and C. Tuleau-Malot, "Inference of functional connectivity in Neurosciences via Hawkes processes," in *Proc. IEEE Global Conf. Signal and Information Processing (GlobalSIP)*, Austin, TX, Dec. 2013.

[4] D. Lando and M. Nielsen, "Correlation in corporate defaults: Contagion or conditional independence?," *J. Financial Intermediation*, vol. 19, no. 3, pp. 355–372, 2010.

[5] R. Crane and D. Sornette, "Robust dynamic classes revealed by measuring the response function of a social system," *Proc. Natl. Acad. Sci.*, vol. 105, no. 41, pp. 15649–15653, 2008.

[6] Z. Harmany, R. Marcia, and R. Willett, "This is SPIRAL-TAP: Sparse Poisson Intensity Reconstruction ALgorithms – Theory and Practice," *IEEE Trans. Image Processing*, vol. 21, no. 3, pp. 1084–1096, 2012.

[7] Q. Tran-Dinh, A. Kyrillidis, and V. Cevher, "Composite self-concordant minimization," *J. Machine Learning Research*, vol. 16, pp. 371–416, 2015.

[8] M. Moore and M. Davenport, "Learning network structure via Hawkes processes," in *Proc. Work. Signal Processing with Adaptive Sparse Structured Representations (SPARS)*, Cambridge, UK, July 2015.

[9] M. Moore and M. Davenport, "A Hawkes' eye view of network information flow," in *Proc. IEEE Work. Stat. Signal Processing (SSP)*, Palma de Mallorca, Spain, June 2016.

[10] K. Zhou, H. Zha, and L. Song, "Learning triggering kernels for multi-dimensional Hawkes processes," in *Proc. Int. Conf. Machine Learning (ICML)*, Atlanta, GA, June 2013.