Sequential data assimilation for 1D self-exciting processes with application to urban crime data

N. Santitissadeekorn[a,*], M. B. Short[b], D. J. B. Lloyd[a]

[a] *Department of Mathematics, University of Surrey, Guildford, GU2 7XH, UK*

[b] *Georgia Institute of Technology, School of Mathematics, Atlanta, GA, USA*

Abstract

A number of models – such as the Hawkes process and log Gaussian Cox process – have been used to understand how crime rates evolve in time and/or space. Within the context of these models and actual crime data, parameters are often estimated using maximum likelihood estimation (MLE) on batch data, but this approach has several limitations such as limited tracking in real-time and uncertainty quantification. For practical purposes, it would be desirable to move beyond batch data estimation to sequential data assimilation. A novel and general Bayesian sequential data assimilation algorithm is developed for joint state-parameter estimation for an inhomogeneous Poisson process by deriving an approximating Poisson-Gamma 'Kalman' filter that allows for uncertainty quantification. The ensemble-based implementation of the filter is developed in a similar approach to the ensemble Kalman filter, making the filter applicable to large-scale real world applications unlike nonlinear filters such as the particle filter. The filter has the advantage that it is independent of the underlying model for the process intensity, and can therefore be used for many different crime models, as well as other application domains. The performance of the filter is demonstrated on synthetic data and real Los Angeles gang crime data and compared against a very large sample-size particle filter, showing its effectiveness in practice. In

*Corresponding author

Email addresses: n.santitissadeekorn@surrey.ac.uk (N. Santitissadeekorn), mshort9@math.gatech.edu (M. B. Short), d.j.lloyd@surrey.ac.uk (D. J. B. Lloyd)

addition the forecast skill of the Hawkes model is investigated for a forecast system using the Receiver Operating Characteristic (ROC) to provide a useful indicator for when predictive policing software for a crime type is likely to be useful. The ROC and Brier scores are used to compare and analyse the forecast skill of sequential data assimilation and MLE. It is found that sequential data assimilation produces improved probabilistic forecasts over the MLE.

Keywords: Nonlinear filtering, Hawkes Process, joint state-parameter estimation, count data, particle filtering, ensemble Kalman filter

---

1. Introduction

Constructing computational algorithms for predictive policing is one of the emerging areas of mathematical research. Given that police departments worldwide are frequently asked to deliver better service with the same level of resources, algorithms that can better allow authorities to focus their resources could be of great value. To this end, there have been several methods developed over the years to help make crime predictions and ultimately guide policing resources to areas where they are likely to have the biggest impact.

One recently developed algorithm for predictive policing is the Epidemic-Type-Aftershock-Sequence (ETAS) model described in some detail in [1, 2]. The idea behind the ETAS model is that crimes are generated stochastically, but the rate of crime generation is history-dependent, such that crimes occurring within an area will increase the rate of future crime generation in that same or nearby areas for at least some period of time. In practice, the ETAS model functions by taking in daily, up-to-date historical crime data in the form of event times and geolocations, processing them within the model's mathematical framework described below, and then highlighting on a fixed grid (typically 150m squares) the top N locations likely to have crime on that day. The processing is done by carrying out a time series analysis in each grid cell by fitting a self-exciting rate model known as a Hawkes process to the historical data. The stochastic event rate $\lambda(t)$ in this Hawkes

process is given by

$$\lambda(t) = \mu + \sum_{\tau_j < t} q\beta e^{-\beta(t-\tau_j)}, \tag{1.1}$$

where μ is the baseline crime rate, q is a sort of reproduction number that is equal to the expected number of future events spawned by any single crime occurrence, β is the decay rate of the increased crime rate back to the baseline, and τ_j are the times of prior crime events, and each of these vary by grid cell. Hence, at any given moment the crime rate is a linear superposition of Poisson processes, including the homogeneous baseline rate and several exponentially decaying rates equal in number to the number of prior events. The ETAS algorithm then carries out a maximum likelihood parameter estimation (MLE) on batch data to find μ, q, and β; typically q and β are assumed to be the same across all grid cells, while μ is allowed to vary from cell to cell. Finally, those N cells with the highest estimated λ are highlighted for that day.

Two recent randomised field-trials conducted with police departments in Los Angeles, CA and Kent, UK [2] showed that the ETAS algorithm was able to predict 1.4-2.2 times as much crime as a dedicated crime analyst using existing criminal intelligence and hotspot mapping practices. The trials were also able to show that dynamic police patrolling based on the ETAS algorithm led to an average of 7.4% reduction in crime volume at mean weekly directed patrol levels, whereas patrols based upon analyst predictions showed no statistically significant effect.

Despite the success of these field-trials, one fundamental question for any predictive policing algorithm is whether or not a given crime type is 'predictable' at any practical level, and by how much. Analysing the operational predictability and forecast skill of any predictive policing software is crucial in determining its worth. That is, even if one had a perfect model for the crime rate and complete knowledge of all model parameters, would the resulting predictions be actionable in any useful way? In spatio-temporal crime forecasting, a classic measure is the prediction efficiency index (PEI) [3]. This index requires that for each prediction period of interest (day, week, etc.) a subset of the total spatial region in question

be marked as the region of interest for that period. The PEI is the ratio of the number of events occurring within the chosen region of interest to the greatest possible number of events that could have occurred over all potential regions of interest having the same size over that period; the PEI is therefore ≤ 1. Hence, the PEI essentially captures how well the prediction algorithm performs versus an oracle that had true knowledge of where events would occur over the period in question for a fixed predicted area size. While this measure is of practical importance, it does not take into account the probabilistic nature of the underlying crime process. That is, if one assumes crime is in fact a stochastic process, then even if one knew with certainty all details of the process, there would be no reason to necessarily expect a PEI of 1, and in fact the expected PEI in such a scenario could still be quite small. Currently, there has been no assessment from a probabilistic view of when crime location and rate is fundamentally predictable (or not), despite the fact that this knowledge would be useful for both police forces and predictive policing researchers and companies.

Another problem with the ETAS method is that there is no ability to track in real time uncertainty in either the fitted parameters or the model predictions, which could arise due to noisy and limited data or model selection errors. While forecasts such as the ETAS cell highlighting would not necessarily take into account such uncertainty, it is important to know from a police patrolling strategy perspective. For instance, measures of uncertainty can help to determine if the police are more likely to cover the most crime locations by increasing/decreasing the number of locations to patrol.

More sophisticated Bayesian methods for estimation using batch data have been looked at by Shinichiro & Gelfand [4] for a Log Gaussian Cox Process (LGCP) and Mohler [5] for a combination of an LGCP and Hawkes process to model the crime rate. These methods, though, are somewhat specific to the model in question. It would be desirable to have a sequential Bayesian method for estimation that is independent of the underlying model for the crime rate so that model comparison can be carried out for instance between the Hawkes process and LGCP using the same estimator in each case. One of the major computational advantages of such a method

would be that the entire data history of observations would not be needed. Taddy [6] developed a sequential Monte-Carlo filtering method for a Poisson dynamic linear model. This filtering method has the drawback that it can not be applied to self-exciting models such as the Hawkes process. Particle filtering [7–11] would be able to achieve this and is considered the "gold-standard" in sequential Bayesian filtering as it has been proved to converge to the posterior distribution as the number of particles tends to infinity. However, it comes with a major draw-back that is suffers from the "curse of dimensionality" and hence one needs to develop a sequential Bayesian filter that is computationally feasible in practice.

In order to overcome some of these problems and fill in gaps within the literature, we propose here a sequential data assimilation approach to estimation of predictive policing models that will systematically incorporate uncertainty and real-time tracking, enabling us to investigate the effect of uncertainty in an operational context. To do this in a computationally efficient manner, we develop an Ensemble Poisson-Gamma filter motivated by the Ensemble Kalman Filter (EnKF) used in geophysical applications [12, 13]. Even though EnKF allows for non-normal prior distributions and relaxes the assumption of a normal likelihood, a highly skewed and non-negative (posterior) distribution can be better approximated, for instance, by a gamma distribution. The uncertainty of crime intensity rate and observations for some type of crimes (e.g. burglary) could be small, leading to a highly skewed uncertainty for the burglary rate. By taking the EnKF philosophy we build a computationally efficient and robust filter for point processes, with emphasis on the Hawkes process in our examples, that compares well with the "gold-standard" particle filter implemented with a large sample size limit. We note that while we are able to use the gold-standard particle filter for a single 1D process, in practice spatio-temporal predictive policing software has to carry out filtering for many grid cells, making the particle filter computationally infeasible. We note that the filter can easily be applied to other crime rate models, such as the LGCP, that allows for model comparison to be carried out, and we demonstrate this on synthetic data.

We also assess the operational predictability and forecasting skills of Poisson

rate models at a fundamental level using the Receiver Operating Characteristic (ROC). This characteristic measures the positive hit rate verses the false alarm rate and allows us to assess predictive policing models precisely. The ROC has been suggested before by a few authors (e.g. [14]) as a good way to measure the success or failure of predictive policing software; however their studies have focused on particular data sets rather than a theoretical assessment of parameter regions where the underlying process is predictable or not. By carrying out a theoretical synthetic experiment using the particle filter, we find parameter regions where the Hawkes process is "ROC-predictable" and where the data assimilation approach shows improved skill over the MLE based ETAS algorithm. We further demonstrate our method on real LA gang violence data to show its effectiveness in practice. The MLE and filter methods are also compared using the Brier score for probabilistic forecasts.

The paper is outlined as follows. In section 2 we introduce ensemble filtering for sequential data assimilation, provide an overview of the "gold-standard" particle filter approach, then develop our own approach that we call the Ensemble Poisson-Gamma filter for a univariate variable. In section 3 we demonstrate both the accuracy and efficiency of our approach versus the particle filter on simulated data generated via a Hawkes process that also allows a joint state-parameter estimation. In section 4, we demonstrate the Ensemble Poisson-Gamma filter using a Log Gaussian Cox Process for the crime rate using synthetic data. In section 5 we employ our method on real data from Los Angeles and assess the results. In section 6 we undertake a general study on the inherent predictability of Poisson rate models, then compare our method to the ETAS algorithm. Finally, we conclude and discuss future directions and open questions in section 7.

2. Ensemble-based filtering

Consider a counting process $N(t)$ associated with the conditional intensity function

$$\lambda(t|H_t) := \lim_{\delta t \to 0} \frac{\mathbf{Pr}(N(t+\delta t) - N(t) = 1|H_t)}{\delta t}, \tag{2.1}$$

where H_t is the event history of the process up to time t, containing the list $\{0 < \tau_1 < \cdots < \tau_{N(t)} < t\}$, where τ_j is the time of the j-th event and $\tau_{N(t)}$ is the time of the last event prior to t. We will use a shorthand notation $\lambda(t) := \lambda(t|H_t)$ in the rest of this work. For this work, we consider a discrete-time intensity process $\lambda_j := \lambda(t_j)$ being a constant in the j-th time step $[(j-1)\delta t, j\delta t)$ for $j = 1, \ldots, n$, where a time step δt is small enough such that the discrete process is a good approximation of the continuous time process. Generally, this means that the number of events occurring within any step is small. Let y_j be the number of events in the j-th time step and $y_{1:n} = \{y_1, \ldots, y_n\}$ denote the collection of observations up to the n-th time step. Then the probability of observing y_j is Poisson distributed

$$\mathbf{Pr}(y_j|\lambda_j) = (\lambda_j \delta t)^{y_j} \exp(-\lambda_j \delta t). \tag{2.2}$$

The state of the system and model parameters during any time interval δt are assumed to be constant within the interval and a random variable v_j collectively denotes both state and parameters in the $j-$th time interval. One goal of this paper is to develop a discrete-time filtering method for the intensity process described by (2.1) and (2.2) that involves a recursive approximation of the probability density $p(v_j|y_{1:j})$ given the probability density $p(v_{j-1}|y_{1:j-1})$. In other words, we wish to recursively make an inference of the unknown state of a dynamical system (as well as model parameters) using only the data from the past up to the present.

In most filtering algorithms, the computation of $p(v_j|y_{1:j})$ consists of two main steps: (1) the Prediction step, which computes $p(v_j|y_{1:j-1})$ based on $p(v_{j-1}|y_{1:j-1})$ using the transition kernel $p(v_j|v_{j-1})$; and (2) the Analysis step, which uses Bayes's formula to compute $p(v_j|y_{1:j})$ given a prior density $p(v_j|y_{1:j-1})$ for v_j. When the prior density and likelihood are both normal, the normal posterior density $p(v_j|y_{1:j})$ is recursively given in a closed-form expression by the Kalman filter. However, a numerical approximation is typically needed in general cases. One such method, discussed more extensively below, is the particle filtering (PF) method, which provides an ensemble approximation of $p(v_j|y_{1:j})$. This method has become increasingly popular in practical applications since it is relatively simple to implement and able to

reproduce the true posterior $p(v_j|y_{1:j})$ in the large sample limit. Nevertheless, it suffers from the curse of dimensionality and the design of efficient algorithms can be challenging. Though the time-series examples considered in this work may all be tractable with the standard PF method, our ultimate goal is to consider higher dimensional spatio-temporal data, which may require an algorithm that is more scalable than PF. In this work, we develop a novel ensemble-based filtering algorithm geared to assimilating data where the likelihood function is described by (2.2) with the application to crime data analysis in mind. The new algorithm is built upon the Poisson-Gamma conjugate pair in the univariate case, which can be extended to a multivariate case via the serial update scheme as commonly used in the serial-update version of the ensemble Kalman filter (EnKF), see [15]. Unlike the PF, where particle weight is updated according to Bayes's rule, the new algorithm provides a formula that attempts to directly move the ensemble into the region with a high posterior probability. In the rest of this section, we will briefly review the concept of PF and then describe how to construct a new ensemble-based algorithm.

2.1. Particle filter (PF)

Since we will be using a particle filter in a large sample size limit to assess the quality of our new algorithm, we present here how a basic particle filter works in a nutshell for our specific application and encourage the reader to consult [7–11] for theoretical details and discussions in general cases. We begin with a discrete-time (hidden) Markov process $\{V_j\}$ corresponding to an $\mathbb{R}^d$-value that is not directly observed. Instead we observe a process $\{Y_j\}$, which is the count data in the current application. Owing to the Markovian assumption of the hidden process, the joint probability density of $\{V_j\}$ for $j = 1, \ldots, k$ is given by,

$$p(v_{1:k}) = f_1(v_1) \prod_{j=2}^{k} f_j(v_j|v_{j-1}), \tag{2.3}$$

where $f_j(v_j|v_{j-1})$ is a transition density function at the j-time step and $f_1(v_1)$ is an initial density. For a hidden Markov model (HMM), the con-

ditional joint density $Y_{1:k}$ given $V_{1:k} := \{v_1, \ldots, v_k\}$ is typically assumed to have the following conditional independence form:

$$p(y_{1:k}|v_{1:k}) = \prod_{j=1}^{k} p(y_j|v_j). \tag{2.4}$$

For the current application $p(y_j|v_j)$ is the Poisson likelihood probability given by (2.2) and v_j usually combines the conditional intensity λ_k, which also depends on unknown model parameters. The inference problem then follows a recursive decomposition:

$$p(v_k|y_{1:k}) = \frac{p(y_k|v_k)}{p(y_k|y_{1:k-1})} p(v_k|y_{1:k-1}). \tag{2.5}$$

Under the Markovian assumption, we can write

$$p(v_k|y_{1:k-1}) = \int f_k(v_k|v_{k-1}) p(v_{k-1}|y_{1:k-1}) dv_{k-1}. \tag{2.6}$$

In other words, given $p(v_{k-1}|y_{1:k-1})$, the filtering here is concerned with the sequential computation of $p(v_k|y_{1:k})$ as the index k is incremented.

A PF algorithm is designed to approximate the density $p(v_k|y_{1:k})$ by M weighted particles (i.e. empirical random measure)

$$p(v_k|y_{1:k}) \approx \sum_{i=1}^{M} w_k^{(i)} \delta(v_k - v_k^{(i)}), \qquad \sum_{i=1}^{M} w_k^{(i)} = 1. \tag{2.7}$$

The weighted particles are sequentially updated via two main recursive steps: prediction and analysis. The prediction step draws $v_k^{(i)} \sim p(v_k|v_{k-1}^{(i)}, y_{1:k})$ to generate a new particle $v_k^{(i)}$. This sampling scheme is often the most convenient choice. The particle weight is unchanged in this step. Thus, the prediction step yields an ensemble approximation

$$p(v_k|y_{1:k-1}) \approx \sum_{i=1}^{M} w_{k-1}^{(i)} \delta(v_k - v_k^{(i)}). \tag{2.8}$$

In the analysis step, the current data y_k is assimilated to update the particle weight via Bayes's formula. For the above implementation of the prediction

step, the new particle weight is updated as

$$w_k^{(i)} = \frac{\tilde{w}_k^{(i)}}{\sum_{i=1}^{M} \tilde{w}_k^{(i)}}, \qquad \tilde{w}_k^{(i)} = p(y_k|v_k^{(i)}) w_{k-1}^{(i)}. \tag{2.9}$$

After the analysis step, we obtain a new empirical approximation of $P(v_k|y_{1:k})$ represented by weighted particles $\{w_k^{(i)}, v_k^{(i)}\}$.

The above algorithm may lead to the issue of weight degeneracy when only few particles have significant particle weights and all other weights are negligibly small. If unabated, there could eventually be only one particle left with weight 1. An additional step called resampling is conventionally employed to mitigate this issue. The implementation of the resampling step in this work is based on the residual resampling method, see Appendix A. Note that the above weight update is usually called the "bootstrap filter", which is the simplest version of PF, but it is usually considered to be inefficient since it may require a large number of particle to well approximate the desired density, depending on many factors such as the dynamic of the model, the likelihood function, and the dimension of the problem. A more general weight update equation can be designed based on importance sampling and in some cases an optimal proposal density can be achieved to minimise the variation of the sample representation. The detail of the optimal particle filtering is out of the scope of the current work and in-depth discussion may be found in [8, 10]. We will use the bootstrap filter in this work since we are able to increase the sample size to the level where the ensemble distribution is unchanged as the sample size increases. Due to its convergence property, the PF-generated ensemble, in a large sample limit, will be used as the gold standard to test the performance of our novel ensemble method in Section 2.2.

2.2. Poisson-Gamma filter

Suppose for now that the hidden state v_k includes only the intensity λ_k, i.e., $v_k \equiv \lambda_k$. We will extend our algorithm for the case of the combined state and parameter later on. In contrast to PF, which constructs the ensemble approximation by updating the particle weights, our new algorithm will take

a different approach where a set of uniformly-weighted particles is used for the approximation; hence we will attempt to place most of these particles in a high probability region. The new algorithm also consists of two main steps. In the prediction step, the particles are propagated in the same fashion as PF, i.e., $\lambda_k^{(i)} \sim f_k(\lambda_k|\lambda_{k-1})$. However, we will denote these particles by $\lambda_{k|k-1}^{(i)}$ instead since they will be transformed in the analysis step; recall that for PF algorithm these particles are not changed in the analysis but their weights are changed. In the analysis step, the proposed algorithm will then transform the predicted particles $\lambda_{k|k-1}^{(i)}$ to a new set of particles $\lambda_k^{(i)}$, which approximates $p(\lambda_k|y_{1:k})$, according to a stochastic transformation that will be derived below.

To this end, suppose that the predicted particle $\lambda_{k|k-1}^{(i)}$ has been obtained, usually by propagating $\lambda_{k-1}^{(i)}$ to time step k via some mathematical model. We now demonstrate how we develop the ensemble-based algorithm for the analysis step based on the Poisson-Gamma conjugate pair, specified through the mean and "relative variance" of the (univariate) conditional intensity. To ease notational cluttering, we will suppress the time subscript in this section and it should be understood that the algorithm below is applied to the analysis step at each time step k. It will be seen later that the update formula will be more compact when employing the relative variance $P_r = P/\langle\lambda\rangle^2$, where $\langle\lambda\rangle$ is the mean of λ, instead of the variance of λ, denoted by P. Following standard Bayesian analysis, it is simple to show that if λ has a gamma prior distribution with a mean $\langle\lambda\rangle$ and relative variance P_r, then given the Poisson distribution on y in (2.2), the posterior on λ is also gamma distributed with mean and relative variance $\langle\lambda^a\rangle$ and P_r^a given by

$$\begin{aligned} \langle\lambda^a\rangle &= \langle\lambda\rangle + \frac{\langle\lambda\rangle}{P_r^{-1} + \langle\lambda\rangle\delta t}(y - \langle\lambda\rangle\delta t) \\ (P_r^a)^{-1} &= P_r^{-1} + y. \end{aligned} \tag{2.10}$$

Note that the conventional Bayesian scheme updates the posterior gamma distribution via the so-called scale and shape parameters instead of mean and relative variance. The update formula (2.10) will, however, suite well our ensemble-based filtering algorithm that is intended to approximately

sample the posterior density for a given prior ensemble; this is analogous to the well-known Ensemble Kalman filter (EnKF) which is widely used to sample the posterior distribution when the assumption of normality is not strictly valid but can still be approximately satisfied. In our application, although the prior density may not be exactly a gamma distribution, which tends to be the case in practice, we may still insist to update our ensemble of λ so that its mean and relative variance satisfy (2.10). This is drastically different from fitting the gamma distribution to the ensemble of λ and then updating the scale and shape parameters of the gamma distribution through Bayesian analysis and finally drawing the posterior sample from the posterior gamma distribution described by the updated scale and shape parameters. The latter will always have the sample distributed exactly as a gamma distribution while the former can have a non-gamma sample. We will refer to (2.10) as the Poisson-Gamma filter (PGF) and its ensemble-based version as the ensemble Poisson-Gamma filter (EnPGF), which will be derived in the subsequent section.

We now explain how we will generate a posterior sample that satisfies (2.10). Let us suppose that we have a prior sample $\lambda^{(i)}$ for $i = 1, \ldots, M$. Let $A = [\lambda^{(1)}, \ldots, \lambda^{(M)}] - \bar{\lambda}$ be the "anomaly" matrix of size $1 \times M$, where $\bar{\lambda}$ is the sample mean. Thus we can write the sample variance by $P = (AA^T)/(M-1)$ and the relative sample variance P_r can be found accordingly using the sample mean. Given (2.10), we can easily update the posterior ensemble mean, denoted by $\bar{\lambda^a}$, as follows:

$$\bar{\lambda^a} = \bar{\lambda} + \frac{\bar{\lambda}}{P_r^{-1} + \bar{\lambda}\delta t}(y - \bar{\lambda}\delta t) \tag{2.11}$$

The update of the posterior ensemble anomaly, denoted by A^a, is also required so that the posterior sample can be generated by $\lambda^a = \bar{\lambda^a} + A^a$. It is important that the anomaly A^a must be able to produce an ensemble that is consistent with the second line of (2.10). There are several ways to achieve this, which are analogous to several ensemble-based schemes of EnKF, see [13, 16] for "stochastic" formulations and [17, 18] for "deterministic formulations". We focus only on the development of the so-called stochastic formulation in the next section.

2.3. EnPGF: Stochastic update

We first note that, if $y = 0$, (2.10) indicates that the ensemble mean should update, but the ensemble relative variance should remain unchanged. In order to achieve this along with (2.11), one can simply scale each ensemble member such that $\lambda^{(i),a} = \lambda^{(i)}\bar{\lambda}^a/\bar{\lambda}$, and the update is complete. But, for $y \neq 0$, we use a stochastic update scheme in which each individual ensemble member is stochastically perturbed to achieve the sample variance that satisfies (2.10). This can be achieved based on the following stochastic equation:

$$\frac{\lambda^{(i),a} - \bar{\lambda}^a}{\bar{\lambda}^a} = \frac{\lambda^{(i)} - \bar{\lambda}}{\bar{\lambda}} + P_r(P_r + (y)^{-1})^{-1}\left[\frac{\tilde{y}^{(i)} - \bar{\tilde{y}}}{\bar{\tilde{y}}} - \frac{\lambda^{(i)} - \bar{\lambda}}{\bar{\lambda}}\right], \tag{2.12}$$

where $\tilde{y}^{(i)} \overset{iid}{\sim} \mathrm{Ga}(y, 1)$ for $i = 1 \ldots, M$.

The derivation of (2.12) follows a similar idea of the gamma prior and inverse gamma likelihood filter introduced by [19]. Denote each term in (2.12) as the following:

$$\underbrace{\frac{\lambda^{(i),a} - \bar{\lambda}^a}{\bar{\lambda}^a}}_{:=w} = \underbrace{\frac{\lambda^{(i)} - \bar{\lambda}}{\bar{\lambda}}}_{:=s} + \underbrace{P_r(P_r + (y)^{-1})^{-1}}_{:=c}\left[\underbrace{\frac{\tilde{y}^{(i)} - \bar{\tilde{y}}}{\bar{\tilde{y}}}}_{:=t} - \frac{\lambda^{(i)} - \bar{\lambda}}{\bar{\lambda}}\right]. \tag{2.13}$$

Note that $\mathrm{E}[w^2]$ is the posterior relative variance P_r^a, $\mathrm{E}[s^2]$ is the prior relative variance P_r, and $\mathrm{E}[t^2] = \mathrm{Var}(\tilde{y})/(\mathrm{E}[\tilde{y}])^2 = (y)^{-1}$ since $\tilde{y} \sim \mathrm{Ga}(y, 1)$. It is also simple to check that $\mathrm{E}[st] = 0$. Then, by taking the expectation $\mathrm{E}[w^2]$ given the equation above, it follows that

$$\begin{aligned}
\mathrm{E}[w^2] &= \mathrm{E}[s^2] - 2c\mathrm{E}[s^2] + 2c^2\mathrm{E}[s^2 + t^2] \\
P_r^a &= P_r - 2cP_r + c^2(P_r + (y)^{-1}) \\
&= P_r - 2P_r(P_r + (y)^{-1})^{-1}P_r + P_r(P_r + (y)^{-1})^{-1}P_r \\
&= P_r - P_r(P_r + (y)^{-1})^{-1}P_r,
\end{aligned}$$

which, after some simple algebra, matches the update in (2.10). Based on the relative anomaly (2.12), the anomaly A^a for the posterior ensemble can be readily obtained

$$A^a = \left(\frac{\lambda^{(i),a} - \bar{\lambda}^a}{\bar{\lambda}^a}\right)\bar{\lambda}^a. \tag{2.14}$$

Therefore, (2.11) and (2.12) together complete our ensemble update algorithm, which we call the Ensemble Poisson-Gamma filter (EnPGF).

2.4. Tests: Gamma prior and mixture of gamma prior

In this section, we compare the performance of EnPGF and the ensemble Kalman filter (EnKF) in the scenarios where the analytical form of the posterior distribution is available. The sequential aspect of the algorithm is not tested in these cases (i.e. there is just a single observation and $\delta t = 1$ in the above formula). It will be shown that EnPGF outperforms EnKF in most cases, even in the case of large observation y. To this end, we first present a stochastic update method for the ensemble Kalman filter (EnKF), which is a very popular method, especially in geophysical applications, for approximation of filtered distributions in high-dimensional applications. The EnKF exploits the mean and covariance update of the Kalman filter to sample a high probability region of the filtered distribution in the applications where prior sample and observation likelihood are close to being normal. For large λ, it is appealing to apply the EnKF to approximate the uncertainty of λ because if $y \sim Poi(\lambda)$, y can be approximated by a normal distribution $N(\lambda, \lambda)$. Nonetheless, we would have to deal with the homoskedasticity issue. To get around this, we apply the variance stabilizing transformation, i.e., $z = \sqrt{y + 1/4} \sim N(\sqrt{\lambda}, 1/4)$. Therefore, we may use the transformed observation equation for EnKF:

$$z = \sqrt{\lambda} + \eta,$$

where $\eta \sim N(0, 1/4)$. The standard EnKF with stochastically perturbed observation provides a formulation to update the sample as the following:

$$\lambda^{(i),a} = \lambda^{(i)} + K_e(\sqrt{y + 1/4} + \eta^{(i)} - z^{(i)}),$$

where $z^{(i)} = \sqrt{\lambda^{(i)}}$, $\eta^{(i)} \sim N(0, 1/4)$ and K_e is the (ensemble-based) Kalman gain. The discussion of above implementation of EnKF can be found in [13]. We will show in the subsequent section that, albeit appealing, EnKF fails to provide a correct sample representation of the true posterior even in the case of a large λ.

In the tests below, the ensemble size is 100 for both EnPGF and EnKF.

TEST 1: We want to ensure that when the prior sample comes from a gamma distribution, the EnPGF in (2.12) can accurately sample the correct posterior density. We test the EnPGF and EnKF algorithms for various gamma prior densities and observations, which are chosen so that the overlap between prior and posterior densities are gradually reduced. The experimental results in Figure 1 show that EnPGF provides accurate samples in all cases. However, EnKF performs reasonably well only in the case that the prior and posterior densities nearly overlap. Otherwise, it consistently underestimates the mean and variance, even in the case of a large count data. This result may suggest that if the prior uncertainty of λ is similar to a gamma density, EnKF could be useful but only if the data is observed near the mode of the prior density. Thus, if a mathematical model is used to generate a prior distribution, it would have to be able to predict the data very well in order to allow accurate uncertainty quantification, which may be difficult to achieve in practice.

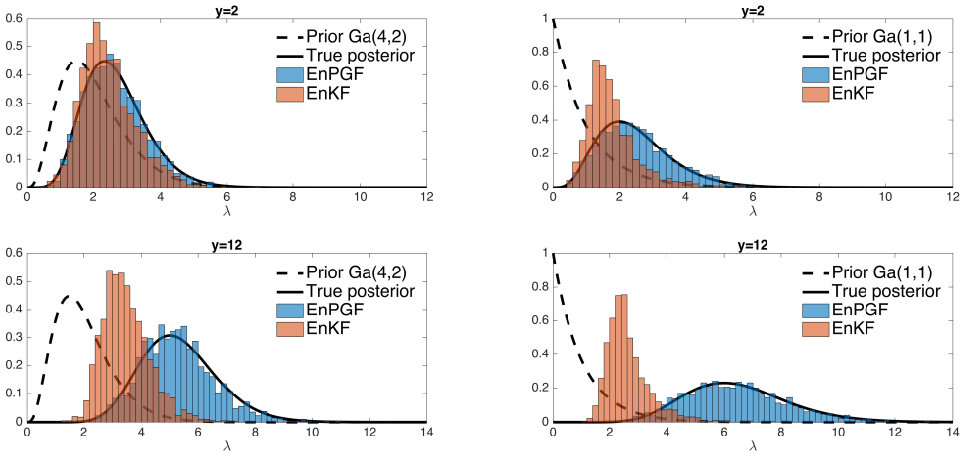

Figure 1: *Comparing histograms generated by EnPGF and EnKF with the true posterior density*

TEST 2: We violate the assumption of gamma prior density by using a

mixture of two gamma densities:

$$p(\lambda) = 0.5\text{Ga}(c_1, d_1) + 0.5\text{Ga}(c_2, d_2),$$

where $\text{Ga}(a, b)$ is a gamma distribution with parameters a and b. The posterior density can be analytically calculated. The results for $y = 4$ and $y = 12$ and various values of c_1, d_1, c_2, d_2 are shown in Figure 2. When the prior and posterior densities significantly overlap, both EnPGF and EnKF work reasonably well and they are only slightly different. However, as the prior and posterior densities becomes more different, EnKF again shows a clear underestimation of the mean while EnPGF can still reliably approximate the significant probability region of the true posterior density, except in the extreme case where the overlap is very small.

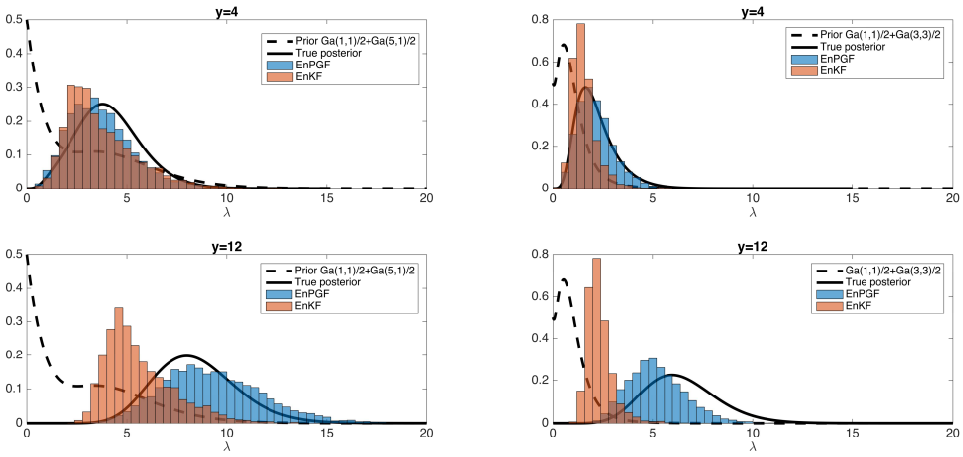

Figure 2: *Comparing histograms generated by EnPGF and EnKF with the true posterior density for the mixture of gamma prior densities. (Top left)* $y = 4$ *and Prior distribution of* λ *is* $0.5\text{Ga}(1, 1) + 0.5\text{Ga}(5, 1)$. *(Top right)* $y = 4$ *and Prior distribution is* $0.5\text{Ga}(1, 1) + 0.5\text{Ga}(3, 3)$. *(Bottom left)* $y = 12$ *and Prior distribution is* $0.5\text{Ga}(1, 1) + 0.5\text{Ga}(5, 1)$. *(Bottom right)* $y = 12$ *and Prior distribution is* $0.5\text{Ga}(1, 1) + 0.5\text{Ga}(5, 1)$.

3. State-space model for 1D Hawkes process

In order to implement the EnPGF for our chosen application, we require a state-space model for the crime rate $\lambda(t)$. For the log Gaussian Cox process, it is defined as a dynamical state-space model and so can easily be implemented for the EnPGF. However, for the Hawkes process this is not the case and one needs to define a state-space model that approximates the process. Hence, we consider the stochastic state-space model

$$\lambda(t+\delta t) = \mu + (1-\beta\delta t)(\lambda(t)-\mu) + kN_t, \qquad N_t \sim \text{Poi}(\lambda(t)\delta t), \tag{3.1}$$

where $\lambda(t)$ is assumed to be a constant in the interval $[t, t+\delta t)$. Under the assumption of the Poisson likelihood (2.2), the EnPGF is available for the state-space model (3.1), even though the distribution of λ may not strictly follow a gamma distribution.

Note that the model (3.1) approximates the first two moments of the Hawkes process (1.1), with $k = q\beta$. In fact, the evolution of the mean $M(t)$ and variance $V(t)$ of (3.1) satisfy the ordinary differential equations

$$\begin{aligned} M' =& \mu\beta + (k-\beta)M \\ V' =& 2(k-\beta)V + k^2 M; \end{aligned} \tag{3.2}$$

see also [20]. Figure 3 demonstrates a good agreement between the sample mean and variance of the Hawkes process (1.1) and the solution of $M(t)$ and $V(t)$ in (3.2), both in the transient and equilibrium stages. In fact, it is well known that the unconditional expected value of the intensity process is $E[\lambda(t)] = \mu(1-k/\beta)^{-1}$, which is exactly the equilibrium solution of $M(t)$. Furthermore, one can readily show that the equilibrium variance of (3.2) is given by $V = k^2\beta\mu/2(\beta-k)^2$, which we find correctly predicts the variance of the intensity process from simulations.

3.1. Tracking intensity

In this experiment, we generate the times of events and "true" intensity $\lambda^*(t)$ from one simulation of the Hawkes process (1.1) with $\mu = 2, k = 1.2$,

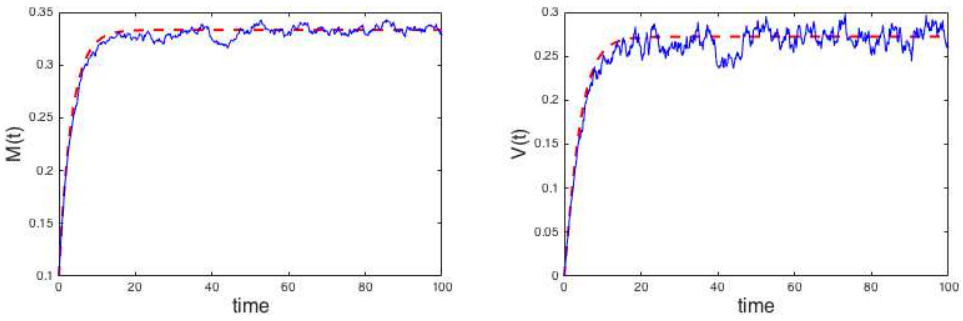

Figure 3: *(Solid line) Mean and variance emperically approximated by the sample generated from the Hawkes process* (1.1) *with parameter values* $\mu = 0.1, k = 0.7, \beta = 1$. *(Dash line) Solutions of the odes* (3.2).

and $\beta = 2$ using Ogata's algorithm [21]. The simulation is taken in the time interval $[0, 110]$ and we remove the transient stage of the intensity in the interval $[0, 10)$ and its corresponding events from the data. Thus, we will rename the time interval $[10, 110]$ to $[0, 100]$ in this experiment. We assume that all parameter values are known, but the current (or initial) intensity $\lambda^*(0)$ is estimated by a sample drawn from the distribution $\text{Ga}(36, 6)$, which has mean 6 and variance 1. We wish to test the filtering ability of EnPGF to track $\lambda^*(t)$ given the data (i.e. times of events). The model (3.1) with $\delta t = 0.1$ is used as a forecast model to generate the ensemble forecast, which empirically represents the prior distribution in the data-assimilation step. Once the data become available at the end of each timestep, EnPGF uses the data to provide a new uncertainty estimate of $\lambda(t)$. Although the true intensity is known in this controlled experiment, the filtering ability of EnPGF with a small sample size is tested by comparing the posterior summary statistics against a "gold standard" sample statistic generated by a particle filter with a large number of particles, which is 200,000 in this case. We denote the sample mean of this gold standard sample by $\lambda^\circ(t)$. As shown in Figure 4, the ensemble mean of EnPGF with 20 samples is able to accurately track the correct posterior mean of the gold standard PF, which is also very close to the true intensity. However, the particle filter with an equally small ensemble size performs poorly, particularly due

to its underestimation of the "temporal hotspots". The sample variance of EnPGF is, however, less smooth than the gold standard case due to the small sample size, and it tends to be lower except in the intervals of the temporal hotspots.

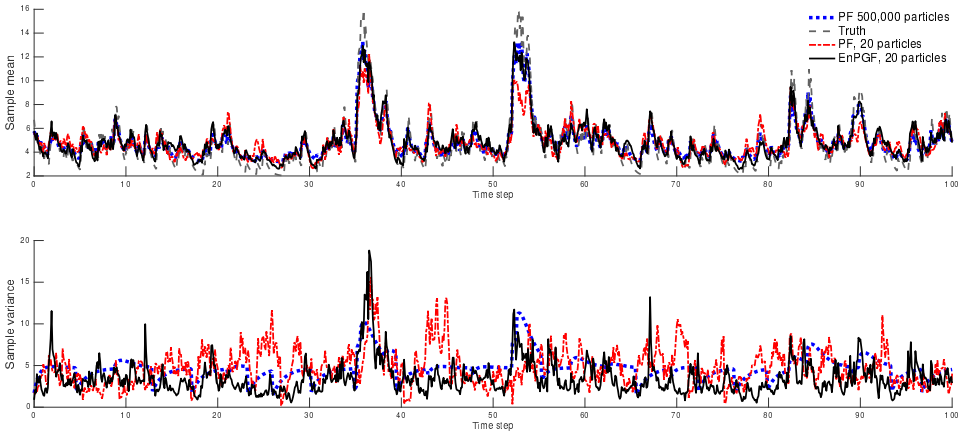

Figure 4: *(Top) The true intensity is a realization of a Hawkes process* (1.1)*. The intensity tracked by PF with 200,000 particles is used as a "gold standard". The estimates of* λ *obtained fom EnPGF and PF, both of which use 20 samples, are compared with the truth and gold standard. (Bottom) The evolutions of the sample variance are compared.*

We also demonstrate that EnPGF has much less "Monte Carlo fluctuation" caused by a small sample size. Figure 5 shows the absolute error $|\lambda(t)-\lambda^*(t)|$ as well as $|\lambda(t) - \lambda^\circ(t)|$ averaged over the time $t = 40 - 100$ since the gold standard PF starts to converge at $t = 40$. Due to the stochastic nature of the algorithm, we investigate the Monte Carlo variation by independently repeating 50 experimental runs for each sample size. We can see that the error $|\lambda(t) - \lambda^*(t)|$ as well as variation in the error for EnPGF are much smaller than PF for all sample sizes. In addition, the error of PF with respect to the gold standard, $|\lambda(t) - \lambda^\circ(t)|$, is significantly larger at a small sample size but becomes slightly better than EnPFG in the large sample size limit, which is of course expected.

To investigate the Bayesian quality of the EnPGF, the posterior density (ap-

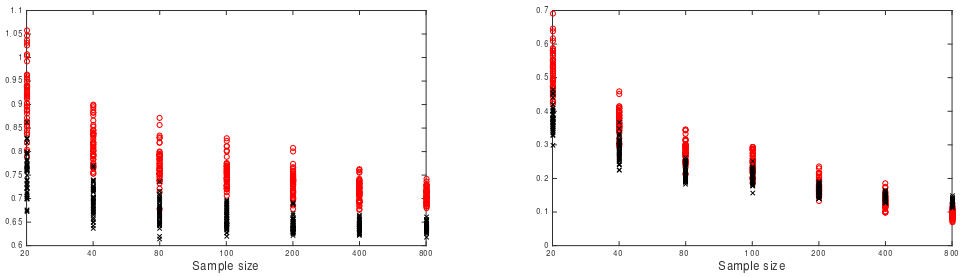

Figure 5: *(Left) absolute error* $|\lambda^*(t)-\lambda(t)|$ *and (Right) absolute error* $|\lambda^\circ(t)-\lambda(t)|$, *both of which are averaged over the time* $t = 40 - 100$. *Again,* $\lambda^*(t)$ *and* $\lambda^\circ(t)$ *are the truth and the gold standard filtered density, respectively. The plot shows the results from 50 experimental runs for each sample size.*

proximated by a probability histogram) at the final time $t = 100$ obtained from the gold standard PF is compared with EnPGF for various sample sizes, see Figure 6. The gold standard posterior histogram evidently exhibits a skewness, which is expected from using the gamma prior density, and EnPGF is able to match this feature quite well. The results also show the convergence of EnPGF to the gold standard posterior density in a large sample size limit. For a small sample size, the sample mean still accurately approximates the true mean but the density of EnPGF is less smooth and too concentrated close to the true mean, so it tends to have a smaller variance than the gold standard result as alluded to briefly above based on one experimental run.

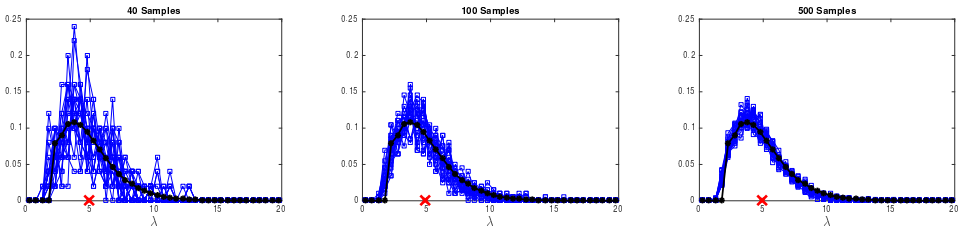

Figure 6: *The approximated density of the intensity at* $t = 100$ *obtained from the gold standard (black) and EnPGF (blue) with indicated sample size. The results from 50 experimental runs for EnPGF are plotted for each sample size. The cross mark indicates the true value of the intensity.*

3.2. Joint intensity-parameter estimation

This section develops a methodology to empirically estimate the joint posterior distribution of λ and model parameters. Thus each ensemble member is now a vector $v_k \equiv (\lambda_k, \theta_{1,k}, \ldots, \theta_{p,k})$ where $\theta_{j,k}$ is the j-th unknown parameter for $j = 1, \ldots, p$ at the time step k. The joint intensity-parameter estimation based on EnPGF follows the same format as the so-called serial-update version of EnKF introduced in [15] for a geophysical application. Again without explicitly writing the time index to avoid cluttering of notation, the process below is applied after each analysis step of the EnPGF. In particular, after obtaining the posterior particles for the intensity, $\lambda^{(i),a}$, based on EnPGF, the difference $\lambda^{(i),a} - \lambda^{(i)}$ is linearly regressed to adjust the other unknown quantities in the vector v. Thus if the i-th prior ensemble member of the j-th model parameter is denoted by $\theta_j^{(i)}$, the i-th posterior ensemble member is given by

$$\theta_j^{(i),a} = \theta_j^{(i)} + \frac{\mathrm{Cov}(\theta_j, \lambda)}{\mathrm{Var}(\lambda)}(\lambda^{(i),a} - \lambda^{(i)}), \tag{3.3}$$

where $\mathrm{Cov}(\theta_j, \lambda)$ is the sample covariance between the j-th parameter and the intensity λ and $\mathrm{Var}(\lambda)$ is the sample variance of the λ.

The joint EnPGF is tested with synthetic data for the case where μ and k in (3.1) are assumed unknown. The true parameters are $\mu = 2$, $k = 1.2$ and $\beta = 2$ while the initial sample of the vector $[\lambda(0), \mu, k]$ is randomly drawn from $N([6, 6, 6], I_3)$, where I_m is an identity matrix of size m. The estimates given by PF with 500,000 particles are used as a gold standard to examine the performance of EnPGF and PF with a small sample size. We also compare the results against the maximum likelihood estimate (MLE) where the parameter vector $[\lambda(0), \mu, k]$ is estimated for the model (3.1) but replacing the stochastic term by the observed times of events. By doing so, the model for MLE is nearly identical to the data-generating model (i.e. Hawkes model (1.1)) for a sufficiently small δt. Therefore, the model used by the MLE in this experiment is more ideal than PF and EnPGF. It is important to bear in mind that in the MLE approach the entire history of observations up to time t_k is used to calculate the estimate at time t_k

while in the filtering approach the past observation is never used again. Another difference is that the MLE uses a new estimate of $[\lambda(0), \mu, k]$ at time t_k to produce a new entire trajectory estimate of λ up to time t_k while the filtering approach updates only the ensemble of the current state λ_k (using only observation at t_k), which then becomes an ensemble of initial conditions for the next assimilation step. Figure 7 compares the results obtained from one experimental run, where both EnPGF and PF have a sample size of 300. Note that the intensity shown for the MLE is λ_k obtained by using the observation up to time t_k, not the intensity re-analysed over all observations at the final time, $t = 100$. The MLE clearly provides the most accurate parameter estimates but the gold-standard PF converges slightly faster to the truth. More importantly, the EnPGF, using a small sample size, also performs well and clearly outperform the PF at the same small sample size. However, the EnPGF produces an over-spreading ensemble for μ when compared to the gold standard PF.

In Figure 8, we show the Monte Carlo error with respect to the truth and the gold standard estimate. We perform DA for 50 experimental runs for varying sample sizes. The error is averaged over the time $t = 40 - 100$ since the gold standard PF starts to converge at $t = 40$. It can be seen that EnPGF estimates of λ and k have substantially less Monte Carlo fluctuation than PF when using small sample sizes. We also test the case where $\mu = 0.5$, $k = 1.2$, and $\beta = 2$ and find similar results, see Figure 9. This demonstrates the strength of EnPGF over PF in a small sample size. As for the MLE, the results in Figures 8 and 9 show a high accuracy of the estimate for the true parameters; again the MLE setting is more ideal than filtering in this experiment.

In Figure 10, we examine the Bayesian quality of the parameter estimates k and μ given by EnPGF in a large sample size limit. In particular, we compare the histograms of the gold standard PF and EnPGF (with 5000 samples) at the final time $t = 100$. Interestingly, in the case of $\mu = 2$, $k = 1.2$, the EnPGF gives an estimate that is closer to the truth than PF, which becomes more evident in the case of $\mu = 0.5$, $k = 1.2$. It is also clear that EnPGF produces samples of k and μ with a stronger (negative) correlation than the

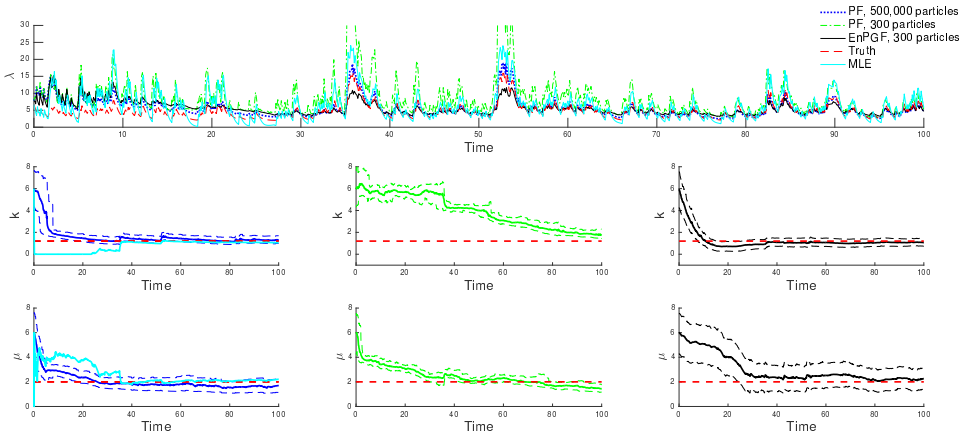

Figure 7: *A result for the joint estimation of λ, k, and μ. The sample size for EnPGF and PF is both 300 particles. (Top row) Comparison of the sample mean of $\lambda(t)$. (Middle row) Comparison of the estimates for k. The sample mean plotted in a solid curve and the 90% quantiles plotted in a dash curve for the gold standard in the left column , PF in the middle column and EnPGF in the right column. Note that the ML estimates for k is overlaid on the gold standard result in the left column. (Bottom row). Comparison of the estimates for μ, which is arranged in the same manner as the parameter k in the middle row.*

gold standard PF. This is, of course, a result of estimating μ and k through an ensemble-based linear regression through (3.3).

4. Hawkes-Cox Process

In this section, we demonstrate the EnPGF on a somewhat different crime model, the Hawkes-Cox process of [5], which combines the Hawkes process above with a Log Gaussian Cox Process (LGCP) for the background rate μ. We focus on the discrete-time version in this work as the continuous-time version can readily be transformed into discrete-time via time discretisation; see [5] for the continuous-time version. Thus we consider the time interval $[(k-1)\delta t, k\delta t)$ for $k = 1, 2, \ldots$, for a small time interval δt and the discrete-

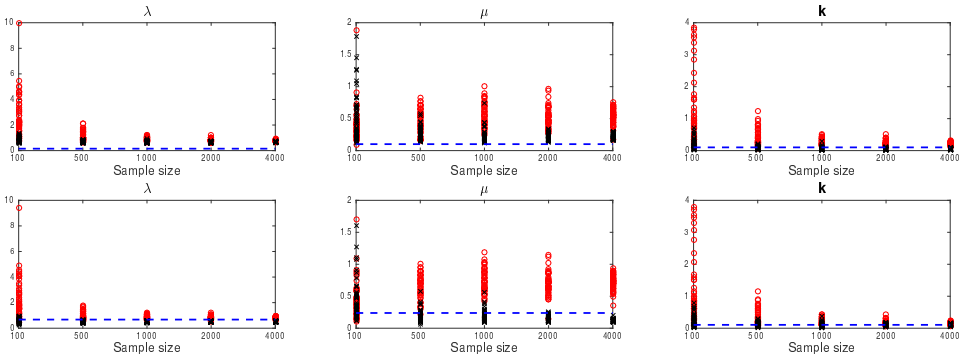

Figure 8: *(Top) absolute error* $|\lambda^*(t) - \lambda(t)|$ *and (Bottom) absolute error* $|\lambda^\circ(t) - \lambda(t)|$*, both of which are averaged over the time* $t = 40 - 100$ *for the case* $\mu = 2$*,* $k = 1.2$ *and* $\beta = 2$*. The EnPGF results are shown in the black cross mark and the results of a small sample-size PF are shown in red circle. The MLE estimate is plotted with the blue dashed line.*

time Hawkes-Cox process is given by:

$$x_{k+1} = x_k - \omega_1(x_k - \mu)\delta t + \sigma\sqrt{\delta t}Z_k \tag{4.1a}$$

$$\lambda_{k+1} = \exp(x_{k+1}) + (1 - \omega_2\delta t)(\lambda_k - \exp(x_k)) + \theta y_k, \tag{4.1b}$$

Here y_k is the number of events in the time interval $[(k-1)\delta t, k\delta t)$ and we take $y_0 = 0$. The (time-dependent) baseline of the intensity function λ_k is determined by a Gaussian process x_k. The parameters ω_2 and θ determine the decay rate of the self-excitation effect and the degree of self-excitation, respectively, similar to the Hawkes process. The stochastic process x_k is a Gaussian process with mean μ (and $x_0 = \mu$), standard deviation σ, and $Z_k \sim N(0, 1)$. The parameter ω_1 controls the decay rate of x to the mean. The parameter estimation of the model (4.1) and its application to crime and security data was demonstrated in [5] using a Metropolis adjusted Langvien algorithm (MALA) to assimilate a time-series count data (all in one large batch) for parameter estimation. It is an "off-line" algorithm that requires a path sampling of the process x_k in (4.1). However, using our EnPGF we can carry out estimation in real-time.

We will assume that the noise standard deviation σ is known and the combined state-parameter vector is $v_k = [\lambda_k, x_k, \mu, \omega_1, \omega_2, \theta]$. We now use

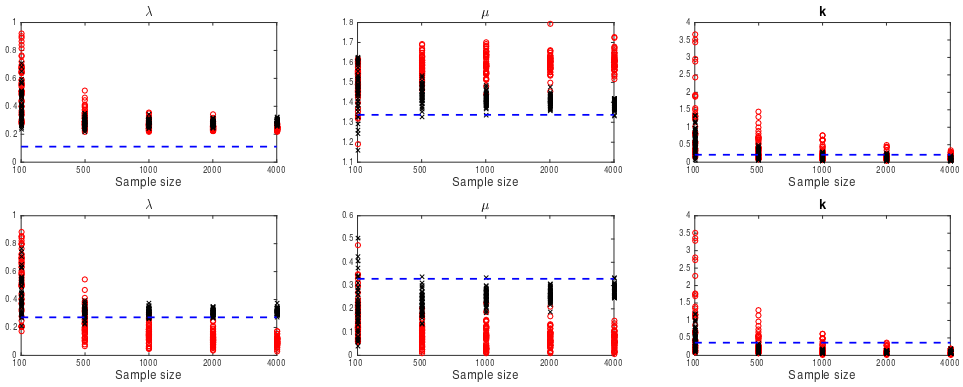

Figure 9: *(Top) absolute error* $|\lambda^*(t) - \lambda(t)|$ *and (Bottom) absolute error* $|\lambda^\circ(t) - \lambda(t)|$*, both of which are averaged over the time* $t = 40 - 100$ *for the case* $\mu = 0.5$*,* $k = 1.2$ *and* $\beta = 2$*. The EnPGF results are shown in the black cross mark and the results of a small sample-size PF are shown in red circle. The MLE estimate is plotted with the blue dashed line.*

EnPGF (with 10^3 particles) to sequentially estimate v_k given y_k. The parameter estimation for a small noise case ($\sigma = 0.1$) is shown in Figure 11 for the evolution of ensemble mean and the histogram at the end of data assimilation. These results are compared with those obtained from PF with a large sample size (10^6 particles). Both PF and EnPGF converge to the true value based on the ensemble mode for ω_1 and ensemble mean for the other parameters. The convergence rate of PF is, however, noticeably faster than EnPGF. Except for ω_1, the posterior densities of the parameters concentrate around the true value. Intuitively, we expect that the uncertainty of ω_1, the rate of decay back to μ, is large due to the smallness of σ. Therefore, we also investigate the case of a large noise ($\sigma = 1$). As shown in Figure 12, both PF and EnPGF results clearly show significant uncertainty reduction of ω_1 but noticeably less reduction for EnPGF. In Figure 13, we compare the true intensity with the tracked intensity, which is estimated by the ensemble mean of EnPGF and show that the true intensity lies mostly between the 10% and 90% quantiles of the tracked ensemble.

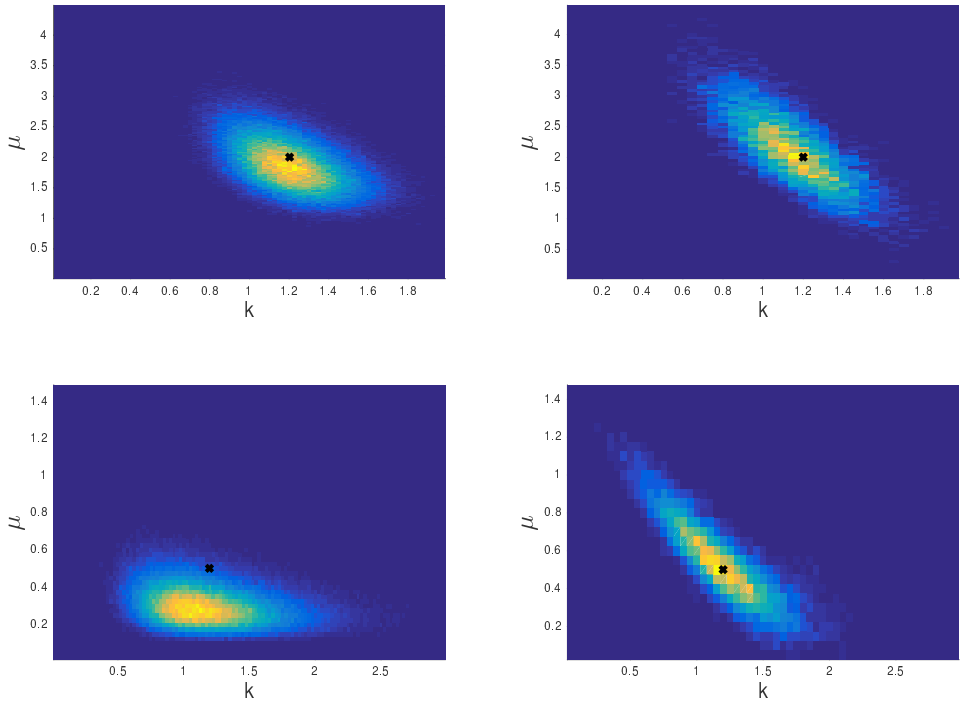

Figure 10: *Joint histograms at the final time* $t = 100$ *of the gold stadard PF (Left) and EnPGF with 5000 sample (Right) for the case* $\mu = 2$, $k = 1.2$ *(Top) and* $\mu = 0.5$, $k = 1.2$ *(Bottom). The cross mark is the true parameter values.*

5. Data Assimilation for gang violence data

As a test of our new algorithm on actual crime data, we use a dataset of over 1000 violent gang crimes from the Hollenbeck policing district of Los Angeles, CA over the years 1999-2002 and encompassing roughly 33 known gangs. The data is analyzed purely as a time series denoted by $0 < \tau_1 < \cdots < \tau_n$, though more information such as victim and suspect gang are available. The summary histograms for the time-series data are shown in Figure 14. Most consecutive violent events occurred within 6 hours and the observed frequency of zero events per day is nearly 40%.

The data under investigation here has been analyzed through the lens of a Hawkes process previously in [22]. One can check the suitability of the

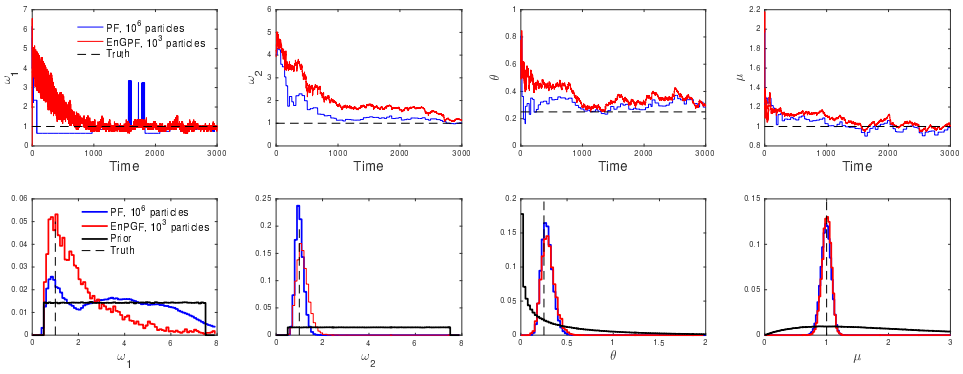

Figure 11: *Small noise case,* $\sigma = 0.1$*, for the Hawkes-Cox model. (Top) Evolution of the ensemble mean, except for the parameter* ω_1 *where the ensemble mode is plotted. (Bottom) The probability histogram of the parameter ensemble.*

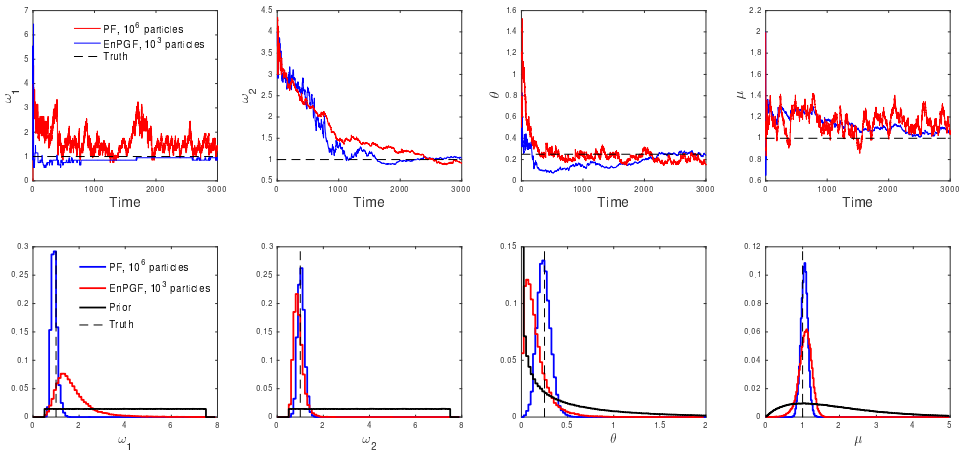

Figure 12: *Large noise case,* $\sigma = 1$ *for the Hawkes-Cox model. (Top) Evolution of the ensemble mean, except for the parameter* ω_1 *where the ensemble mode is plotted. (Bottom) The probability histogram of the parameter ensemble.*

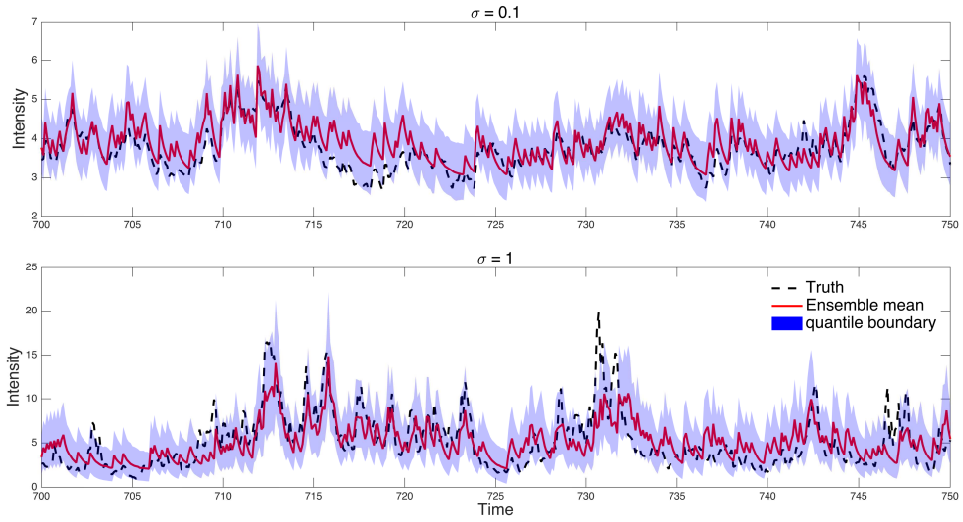

Figure 13: *The true intensity at time* $t = 700 - 750$ *is plotted against the tracked ensemble mean obtained from EnPGF. The upper and lower bounds of the shaded region are the* 90% *and* 10% *quantiles, respectively.*

Hawkes process model for this data by first defining the re-scaled time by

$$u_k := \int_0^{\tau_k} \lambda(t|H_t)dt, \tag{5.1}$$

where $u_0 = 0$. It is well known that if τ_k is a realization from a given $\lambda(t|H_t)$, then $du_k = u_k - u_{k-1}$ are independent exponential random variables with mean 1; hence $z_k = 1 - \exp(-du_k)$ has a uniform distribution $U(0, 1]$. The Kolomogrov-Smirnov (KS) test for z_k can be used to diagnose the consistency of a given model and observed time-series; more precisely, one can look for a significant difference between the empirical cumulative distribution of the z_k derived from re-scaled time and the cdf of $U(0, 1]$.

5.1. Model diagnostic

Suppose that the conditional intensity function is modelled by a Hawkes process as in (1.1), with three parameters μ, β, and $k = q\beta$. The KS test for the gang data is carried out to diagnose the consistency between the gang data and the Hawkes model with various model parameter values. Only the first

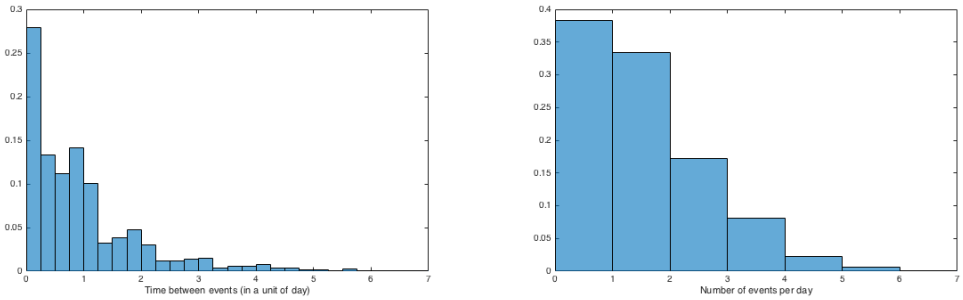

Figure 14: *(Left) Histogram of time between two events in a unit of one day. (Right) Histogram of the number of violence crime event per day*

300 events (out of 1031 events) are used for the test. In particular, we first choose a parameter vector (μ, k, β) in the rectangle $[0, 1.5] \times [0, 1.5] \times [0, 20]$, which is arbitrarily chosen so that it is large enough to contain the maximum likelihood estimates below, plotted in Figure 15. By applying the re-scaled time (5.1) to the gang data, we obtain z_k for each value of the parameter vector and the KS test is used to compare z_k and the uniform distribution as described above. The results are shown in Figure 15 and it can be seen that parameter values for which the P-value of the KS test is above 0.1 are clearly within the 95% confidence interval, which is well approximated by the formula $1.36/\sqrt{n + \sqrt{n/10}}$ for the length of observation $n > 40$ [23]. The geometry of these parameters suggests a broad range of potential values for the decay rate β. The projection of the parameters with P-value above 0.1 onto the (μ, k) plane for various values of β is plotted in Figure 16. It is quite intuitive that the correlation between μ and k is negative for all fixed values of β since having a larger μ would require a smaller k in order to explain the same data. Similarly, a large value of μ would be required for a larger value of β in order to achieve consistency with the data, given the formula for the expected mean λ of (3.2).

We also estimate the parameters using the maximum likelihood estimator (MLE) for the first 300 events in the data. The likelihood function of the Hawkes process can be found in [21]. The optimization of likelihood is computed based on a Nelder-Mead simplex algorithm in MATLAB. As shown

in Figure 15, the MLE lies within the set of parameters with P-value from the KS test above 0.1. We will later use the parameters with a large P-value as the initial knowledge of the model parameters in the Bayesian framework. However, we note that in general large P-values do not imply a higher accuracy of these parameters, but the parameter values in the small neighborhood of the true parameter should have a large P-value. Thus our initial guess based on the P-value of the historical data is not necessarily accurate but it at least contains a region of parameters that is large enough to include the optimal parameters.

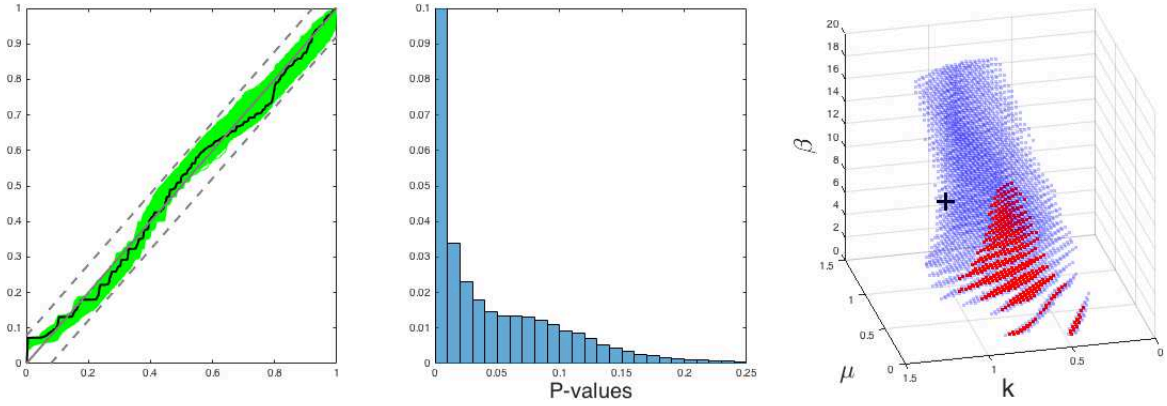

Figure 15: *(Left) KS plot of z_k for the Hawkes model. The dash lines indicate the 95% confidence bound and the green curves are the (observed) cumulative distribution obtained from simulated time-series with parameter values with P-values over 0.1. The black curve is the cumulative distribution of a time-series simulated with the MLE. (Middle) A histogram of P-values for uniform parameter grid points (k, β, μ) in $[0, 1.5] \times [0, 20] \times [0, 1.5]$. (Right) The blue dots shows those parameter with P-value between 0.1 and 0.2 and the red dots shows those with P-value over 0.2. The MLE is shown in the dark + marker ($\mu \approx 0.92, k \approx 1.00, \beta \approx 7.56$).*

5.2. Sequential Data Assimilation

We now apply EnPGF and PF to jointly track the intensity process $\lambda(t)$ and model parameters for the Hollenbeck gang violence data. Again, we use (3.1) as a forecast model for $\lambda(t)$, where we choose $\delta t = 1/6$ days, i.e., every 4 hours. This time interval is small enough in the sense that the data

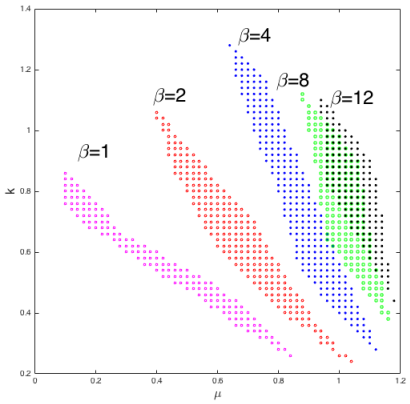

Figure 16: *Projection of the parameters with P-value above 0.1 onto the* (μ, k) *plane for various values of* β*, see also Figure 15*

contains at most 1 event for nearly all of the intervals. As discussed earlier, the range of likely values of β is very broad, so we fix its value to $\beta = 2$ and try to track only the parameters μ and k. We use two different initializations:

Initialization 1: The data during the first 300 days is used to initialize the ensemble of parameters through the KS test as already done in Section 5.1. In particular, we use the parameters on the plane $\beta = 2$ with P-value greater than 0.1, see again Figure 15, as the initial set of parameters. If a sample size M is desired, M particles are randomly drawn from the finite set of the initial parameters and then perturbed with a normal noise $N(0, 0.01I_2)$. We then choose the initial sample $\lambda^{(i)}$ to be the same as $\mu^{(i)}$ for $i = 1, \ldots, M$.
Initialization 2: We draw initial sample (λ, μ, k) from a multivariate normal distribution $N([6, 3, 3], 0.1I_3)$, which is chosen arbitrarily to be far enough from the truth in order to investigate the convergence speed.

We estimate the filtered distribution of $(\lambda(t), \mu, k)$ in the time interval $[300, 600]$ using PF with 200,000 samples for the above two initializations. We test EnPGF with 100 samples using the initialization 2. The results in Figure 17 shows that the estimation given by PF with initialization-1 parameters are relatively stable over the whole data assimilation period. In addition, the sample computed from PF with the initialization 2 converges to the sample

of the initialization-1 parameters. This suggests that the set of parameters found by KS indeed agrees well with the parameters tracked by the PF algorithm. The sample generated by EnPGF shows a similar convergence but with a noticeable discrepancy in the ensemble spread for the parameter k. The empirical distribution at the final time is compared in Figure 18. The marginal distribution of λ and joint distribution of μ and k after the final data assimilation time step are compared for the cases of PF and EnPGF, both with initialization 2. The sample supports of the two results mostly overlap but the sample of EnPGF seems to be relatively overspreading.

6. Predictability and forecast skills

6.1. Predictability

The "predictability" of an event can be defined in many ways. Here we focus on the predictability of a specific type of event and forecast system. We consider a forecast system that releases an "indicator" to alarm an upcoming event of interest. For example, in the context of police patrolling, once patrol officers complete their current assignment, a forecast system may try to suggest the location where criminal activity is most likely to happen within the next hour. In the current time-series application, this kind of forecast system is simplified to predicting whether or not the next violent crime would occur within the next H units of time. After the intensity is updated as a result of the n-th observed crime at time τ_n, we wish to use the intensity of the Hawkes process at τ_n as an indicator variable. Therefore, we may choose a threshold of the intensity, denoted by ℓ, so that whenever $\lambda(\tau_n) > \ell$, the forecast system will suggest to the user that $\tau_{n+1} - \tau_n < H$. As such, the event we wish to predict can be considered as a binary event, say, $Y = 1$ if $\tau_{n+1} - \tau_n < H$ and $Y = 0$ otherwise. The so-called "hit rate" is defined by

$$\mathcal{H}(\ell) := \mathbf{Pr}(\lambda(\tau_n) > \ell | Y = 1), \tag{6.1}$$

and the "false alarm rate" by

$$\mathcal{F}(\ell) := \mathbf{Pr}(\lambda(\tau_n) > \ell | Y = 0). \tag{6.2}$$

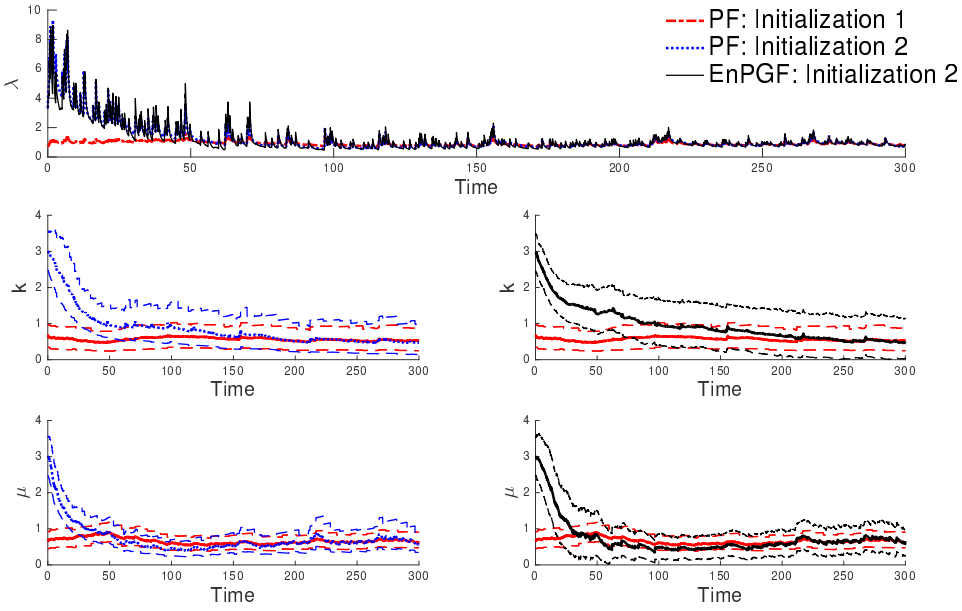

Figure 17: *A result for the joint estimation of* λ, k, *and* μ *for Holenbeck gang violence data in the time interval* $[300, 600]$. *(Top row) Comparison of the sample mean of* $\lambda(t)$. *(Middle row) Comparison of the estimate of* k. *The sample mean is plotted in a solid curve and the* 90% *quantiles is plotted in dash curve. The result for gold standard is plotted in the left column, for PF in the middle column and EnPGF in the right column. Note that the ML estimates for* μ *and* k *is overlaid on the plot of the gold standard result. (Bottom row) Comparison of the estimate of* μ, *which is arranged in the same manner of* k *in the Middle row.*

The Receiver Operating Characteristic (ROC) curve, which is a graph of the pair $(\mathcal{F}(\ell), \mathcal{H}(\ell))$ for various values of ℓ, can be used to measure the performance of a forecast system. We are interested in measuring the performance of two intensity-based forecast systems: (1) the ensemble mean of PF and (2) the Hawkes process with parameters obtained from MLE. As suggested in [24], the hit rate and false alarm rate can be empirically estimated by the observed frequencies. Suppose that we have the indicator-observation pairs $(\lambda_J, y(J))$ for $J = 1, \ldots, n$, sorted in ascending order such that $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ and $y(J)$ is the observation corresponding to λ_J. The empirical hit rate and false alarm rate are given by

$$\mathcal{H}(J) = 1 - \frac{1}{N_1} \sum_{m=1}^{J} y(J), \qquad \mathcal{F}(J) = 1 - \frac{1}{N_0} \sum_{m=1}^{J} 1 - y(J), \tag{6.3}$$

where N_1 is the number of events $y(J) = 1$ and $N_0 = n - N_1$. The ROC plot is the graph $(\mathcal{F}(J), \mathcal{H}(J))$ and its area under the curve (AUC) is usually used to diagnose the association between the indicator variable and the observation. An AUC close to 1 is desired while an AUC of 1/2 would suggest zero association. For empirical ROC, the AUC can be approximated by

$$AUC = \frac{1}{N_0 N_1} \left(\sum_{J=1}^{n} n y(J) - \frac{N_1(1 + N_1)}{2} \right). \tag{6.4}$$

To understand the accuracy of the above forecast system in different parameter regimes of the Hawkes process, we simulated 100 sample paths for various parameters and approximate the AUC in each case, assuming that all true parameters are known. The pairs $(\lambda_J, y(J))$ in this case are the self-excited intensity at the time of observation and $y(J) = 1$ if the subsequent gang-related violence occurs within a certain H unit of time. We investigate the AUC in different parameter regimes, which are determined by the ratio k/β. It is well known that the ratio k/β describes the fraction of events that are endogenously generated by previous events (i.e. being the "offspring" of a past event instead of being a new "immigrant" generated according to the baseline rate μ). Intuitively, the clustering is more pronounced when $k/\beta \to 1$. Figure 19 shows the histogram of the interarrival

time for $k/\beta = 0.1$ and $k/\beta = 0.9$ where μ is chosen for each ratio so that the intensity mean in the equilibrium state from (3.2) is $1/2$ and β is fixed to 1. It is clear that the high clustering regime has much smaller interarrival times. Also shown in Figure 19, the AUC increases as $k/\beta \to 1$ for all values of H used in the test. This is intuitive because in the highly predictable regime the event is unlikely to be generated by the baseline intensity μ and most events are actually the "offspring" of the preceding events.

6.2. Evaluation of forecast skills for the LA gang data

We now measure the performance of the PF-based and MLE-based forecast systems for the gang violence data. Again, the system releases a binary prediction according to the estimated value of $\lambda(t)$ at the time of current event. We use the data $[\tau_1, \ldots, \tau_{300}]$ as "training data" to determine the model parameters using MLE, see again Figure 15. This data size is long enough that the MLE estimate starts to converge. We then use $[\tau_{301}, \ldots, \tau_{600}]$ as a test data to evaluate the forecast skill. First, we consider the empirical ROC and let $\lambda_J \equiv \lambda(\tau_{300+J-1})$ for $J = 1, \ldots, 300$ and $y(J) = 1$ if $\tau_{300+J} - \tau_{300+J-1} < H$, otherwise $y(J) = 0$. The ROC results are shown in Figure 20 for $H = 1/4$ day, and we find that the MLE-based forecasting scheme shows a slightly better ROC performance. Next, we examine the probabilistic forecasting schemes from which the probability assignment of the next event occurring in a specific interval is estimated based on the frequency of the inter-arrival time obtained by a large number of simulations of the model (3.1) given the distributed intensities at the time of the current event, which are presented by the particles, for the PF-based system. Similarly, a large number of simulations is independently run for the MLE-based system using the Hawkes model and the proportionality of the events is used as a probability forecast. In Figure 20, the relative observed frequency (or just observed frequency from now on) of the inter-arrival time in the test data is compared with the forecast probability (averaged over all of the test data) computed by the simulation as explained above. For the interval $(0, 1/4]$, which is the most observed event, the PF-based probability is apparently closer to the observed frequencies than the MLE-based forecast.

We also calculate the Brier score (BS) [25] where the formula is given in Appendix B. In order to make a "reference" Brier score, we calculate the Brier score based on the observed frequencies of each event in the training data (i.e. using this historical frequency at every forecast). Of course, the PF-based or MLE-based forecast system are expected to have lower Brier score if we wish to say that they possess a good forecast skill. The Brier scores for these forecast systems are reported in Figure 20, which demonstrates that both PF-based and MLE-based probabilistic forecasts provide an improved probability forecast over the historical frequency and the PF-based forecast performs slightly better than the MLE-based system.

7. Conclusion

We have introduced a novel sequential data assimilation ensemble Poisson-Gamma filtering (EnPGF) algorithm for discrete-time filtering suitable for real-time crime data that observes repeat event behaviour. The algorithm is independent of the model used for the crime rate, and we demonstrated its effectiveness on two models – pure Hawkes and Hawkes-Cox. The advantage of sequentially updating the forecast in real-time while taking into account uncertainty in model parameters could have a significant impact on predictive policing algorithms that currently use MLE. Computationally, the EnPGF has the major advantage that one does not need the entire history of observations and hence we believe it is feasible to implement in practice. The ensemble mean of EnPGF is used not only to track the true signal, which is the crime intensity rate in this case, but also approximate the parameters of the process. One could then look for "step changes" in the model parameters indicating a need to investigate. These changes are currently hard to detect via MLE with windowing as such algorithms are likely to smooth out the steps. In the numerical experiments, the tracking skill is justified by comparing the estimate with the true signal and the particle filtering (PF) in the large sample size limit. The key strength of EnPGF over PF is its improved accuracy as well as less monte-carlo fluctuation in the case of relatively small sample size. For the real-world time-series of gang

violence data, the validity of EnPGF is testified by comparing with PF and the "likely" parameter region identified by the Kolmogrov-Smirnov statistics. We showed that the results from these distinct data analysis methods happen to agree very well. Nevertheless, our experimental results suggest an issue where EnPGF tends to produce over/under-spreading ensembles for some parameter estimates; hence probabilistically over/under-confidence in the parameter estimates. The implication of this in terms of forecasting would be an interesting future direction for research. Although we demonstrate the application of the new method only to time-series data, the extension to high-dimensional spatio-temporal data can be achieved by serially processing M grid cell time-series analysis one at a time and use the ensemble updated through data assimilation of the first grid cell as a new prior for data assimilation in the second grid cell and so on. This serially-updated formulation is one of the commonly used implementations for EnKF [15]. Work is currently underway to develop this high-dimensional extension of EnPGF and study its effectiveness with burglary data.

In this paper we have also examined the forecast skills provided by the time-series Hawkes model in the perfect model scenario, where all model parameters are known. We studied the forecast system whereby the elevated risk of criminal activities is alerted whenever the crime intensity rate predicted by the data assimilation exceeds a given threshold. Thus, the ROC analysis is a suitable tool to study the ability of such a forecast system. We show that even in the ideal situation, the ROC results vary with parameter regimes. The high clustering regime tends to have a better AUC due to the high probability of generating offspring. We also investigated the impact of using data assimilation in the real-world data in comparison with MLE. We carried out sequential data assimilation using a particle filter with a large sample size to construct the ensemble forecast system. In the gang violence data, our results show that data assimilation and MLE give similar performance with respect to ROC curves, but data assimilation gives a significant improvement in the probabilistic forecast skill based on the Brier score.

Acknowledgments. NS gratefully acknowledges the support of the UK

Engineering and Physical Sciences Research Council for programme grant EP/P030882/1. MBS gratefully acknowledges the support of the US National Science Foundation grant DMS-1737925.

Appendix A: Resampling

In the resampling step, particles with low weights are removed with high probabilities and particles with high weights are multiplied. Thus the computation can be focused on those particles that are relevant to the observations. There are a number of resampling algorithms and most common algorithms are unbiased; hence the key difference in performance lies in the variance reduction, see [26] for a review and comparison of common resampling schemes. The most basic algorithm is the so-called simple random resampling introduced in Gordon, which is also known as multinomial resampling. Suppose that the original set of weighted particles is $\{w_j, v_j\}$ for $j = 1, \ldots, M$. The simple resampling generates a new set of particles $\{1/M, v_k^*\}$ for $k = 1, \ldots, M$ based on the inverse cumulative density function (CDF):

Step 1 Simulate a uniform random number $u_k \sim U[0, 1)$ for $k = 1, \ldots, M$

Step 2 Assign $v_k^* = v_i$ if $u_k \in (q_{i-1}, q_i]$, where $q_i = \sum_{s=1}^{i} w_s$.

In this work, we use the residual resampling algorithm introduced in [27] to reduce the Monte Carlo variance of the simple random resampling. In this approach, we replicate N_j exact copies of v_j according to

$$N_j = \lfloor M w_j \rfloor + \tilde{N}_j,$$

where $\lfloor \rfloor$ denotes the integer part and $\tilde{N}_i$ for $j = 1, \ldots, M$ are distributed according to the multinomial distribution with the number of trials $M - \sum_{j=1}^{M} \lfloor M w_j \rfloor$ and probability of success

$$p_j = \frac{M w_j - \sum_{j=1}^{M} \lfloor M w_j \rfloor}{M - \sum_{j=1}^{M} \lfloor M w_j \rfloor}.$$

The simple resampling scheme can be used to select the remaining $M - \sum_{j=1}^{M} \lfloor M w_j \rfloor$ particles; hence obtaining $\tilde{N}_j$.

The resampling scheme should be used only when it is necessary since by selecting out only high-weight particles, it causes particle depravation. A criterion to activate the resampling step in particle filtering is usually done by setting a certain threshold for the effective sample size, defined by

$$N_{eff} = \frac{1}{\sum_{i=1}^{M} w_i^2}.$$

In this work, we use the resampling step only when $N_{eff} < 1/8$.

Appendix B: Brier Score

Consider r categories of events, and assume each of the observations can occur only in one of these categories. The probability forecast of the i-th event is, therefore, denoted by p_{ij} for $j = 1, \ldots, r$, where $\sum_j p_{ij} = 1$. Given n observations of such categorial data, the Brier score (BS) is defined by

$$BS = \sum_{i=1}^{n} \sum_{j=1}^{r} (p_{ij} - \delta_{ij})^2,$$

where $\delta_{ij} = 1$ if the i-th event occurs in the category j and $\delta_{ij} = 0$ otherwise. In Figure 20, we have $r = 5$ for the events defined by the inter-arrival time within the intervals $((k-1)/4, k/4]$ for $k = 1, \ldots, 4$, and $(1, \infty)$.

References

[1] G. O. Mohler, M. B. Short, P. J. Brantingham, F. P. Schoenberg, G. E. Tita, Self-exciting point process modeling of crime, J. Am. Stat. Assoc. 106 (493) (2011) 100–108.

[2] G. O. Mohler, M. B. Short, S. Malinowski, M. Johnson, G. E. Tita, A. L. Bertozzi, P. J. Brantingham, Randomized controlled field trials of predictive policing, J. Am. Stat. Assoc. 110 (512) (2015) 1399–1411.

[3] S. Menard, Coefficients of determinant of multiple logistic regression analysis, Am. Stat. 54.

[4] S. Shirota, A. E. Gelfand, Space and circular time log Gaussian Cox process with application to crime event data, Ann. Appl. Stat. 11 (2) (2017) 481–503.

[5] G. Mohler, Modeling and estimation of multi-source clustering in crime and security data, Ann. Appl. Stat. 7 (3) (2013) 1525–1539.

[6] M. A. Taddy, Autoregressive mixture models for dynamic spatial Poisson processes: Application to tracking intensity of violent crime, J. Am. Stat. Assoc. 105 (492) (2010) 1403–1417.

[7] N. J. Gordon, D. J. Salmond, A. F. M. Smith, Novel approach to nonlinear/non-Gaussian Bayesian state estimation, Rad. and Sig. Pro., IEE Proc. F 140 (2) (1993) 107–113.

[8] A. Doucet, N. de Freitas, N. Gordon, Sequential Monte-Carlo methods in practice, Springer-Verlag, 2001.

[9] D. Crisan, A. Doucet, A survey of convergence results on particle filtering for practitioners, IEEE Trans. Signal Process. 50 (3) (2002) 736–746.

[10] P. J. V. Leeuwen, Y. Cheng, S. Reich, Nonlinear data assimilation, Frontier in Applied Dynamical Systems: Reviews and Tutorial, Springer, 2015.

[11] K. Law, A. Stuart, K. Zygalakis, Data assimilation: Mathematical introduction, Springer-Verlag, 2015.

[12] G. Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, J. Geophys. Res. 99 (C5) (1994) 10143–10162.

[13] G. Burgers, P. J. van Leeuwen, G. Evensen, Analysis scheme in the ensemble Kalman filter, Mon. Weather Rev. 126 (1998) 1719–1724.

[14] H. Guillaud, Police prédictive : la prédiction des banalités, http://internetactu.blog.lemonde.fr/2015/06/27/police-predictive-la-prediction-des-banalites/, online; accessed 21-Sept-2017 (June 2015).

[15] J. L. Anderson, A local least squares framework for ensemble filtering, M. Weather Rev. 131 (doi: 10.1175/1520-0493) (2003) 634–642.

[16] P. L. Houtekamer, H. L. Mitchell, Data assimilation using an ensemble Kalman filter technique, Mon. Weather Rev. 126 (1998) 796–811.

[17] J. L. Anderson, An ensemble adjustment Kalman filter for data assimilation, Mon. Weather Rev. 129 (2001) 2884–2903.

[18] C. H. Bishop, B. J. Etherton, S. J. Majumdar, Adaptive sampling with the ensemble transform Kalman filter. part i: Theoretical aspects, Mon. Weather Rev. 129 (2001) 420–436.

[19] C. H. Bishop, The GIGG-EnKF: ensemble kalman filtering for highly skewed non-negative uncerstianty distribution, Q. J. R. Meteorol. Soc. 142 (2016) 1395–1412.

[20] J. Fonseca, R. Zaatour, Hawkes process: fast calibration, application to trade clustering, and diffustive limit, J. Futures Mark. 34 (6) (2014) 548–579.

[21] Y. Ogata, On Lewis' simulation method for point process, IEEE Trans. Inf. Theory IT-27 (1) (1981) 23–31.

[22] M. Egesdal, C. Fathauer, K. Louie, J. Neuman, G. Mohler, E. Lewis, Statistical and stochastic modeling of gang rivalries in Los Angeles, SIAM Undergraduate Research Online (2010) 72–94.

[23] W. J. Conover, Practical nonparametric statistics, John Wiley & Sons, 1999.

[24] J. Broecker, Forecast Verification: A Practitioner's Guide in Atmospheric Science, John Wiley & Sons, Ltd., 2012, Ch. 7, pp. 119–139.

[25] G. W. Brier, Verification of forecasts expressed in terms of probability, M. Weather Rev. 78 (1) (1950) 1–3.

[26] R. Douc, O. Cappé, Comparison of resampling schemes for particle filtering, Proc. 4th Int. Symp. on Image and Signal Processing and Analysis, 2005, pp. 64–69.

[27] J. Liu, R. Chen, Sequential Monte Carlo methods for dynamic systems, J. Roy. Statist. Soc. Ser. B 93 (1998) 1032–1044.

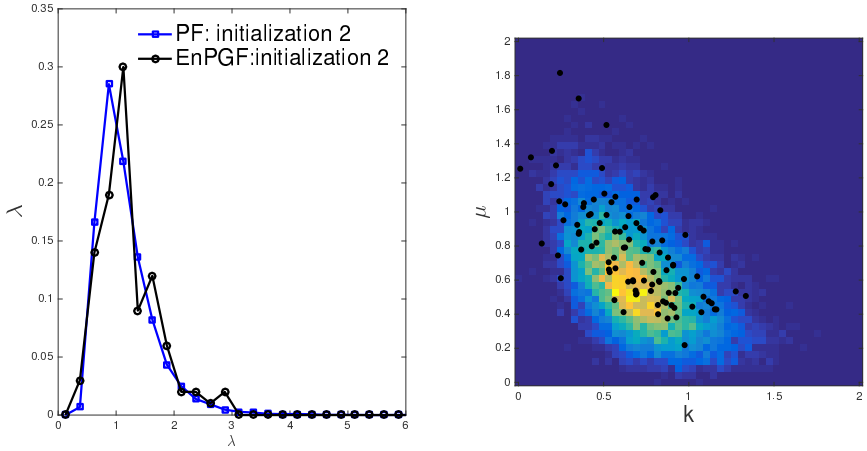

Figure 18: *(Left) Comparing the histogram of λ for PF (200,000 particles) with EnPGF (100 particles), both of which use the initialization 2. (Right) Empirical joint distribution of k and μ at the final step for the gang violence data, which is initialized by using the sample spread obtained from the KS test with the p-values above* 0.1. *The sample of 100-member EnPGF with the initialization 2 is shown in the solid black dots on top of the approximated density obtained from PF*

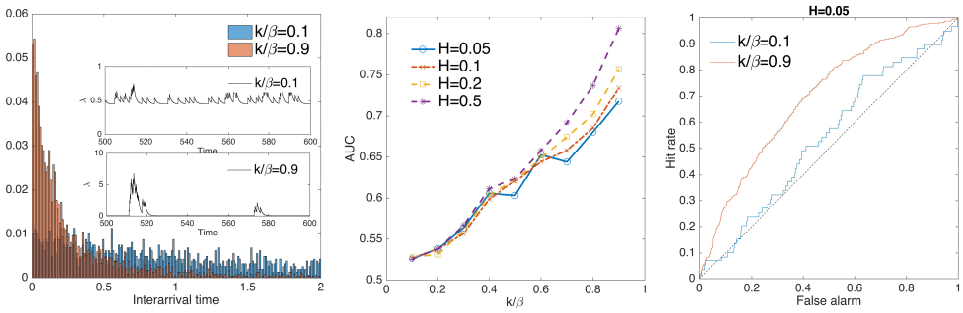

Figure 19: *(Left) Histogram of the interarrival time for $k/\beta = 0.1$ and $k/\beta = 0.9$, where μ is chosen so that the equilibrium mean is 1/2 in both cases. The inserted pictures show a portion of the intensity process generated according to these parameters. In the low clustering scheme ($k/\beta = 0.1$), the events spread out more evenly than in the high clustering case, which show two tight clusters here. (Middle) AUC as a function of k/β for various values of H. (Right) The ROC curve associated with $H = 0.05$; $k/\beta = 0.9$.*

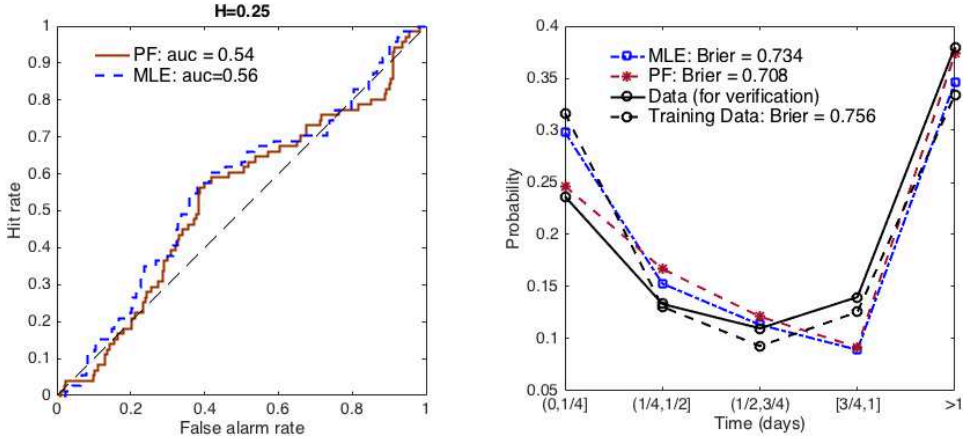

Figure 20: *Forecast skill analysis of the gang data. (Left) Empirical ROC of the PF-based and MLE-based inferences. (Right) Comparison between the observed frequencies of test data (i.e.* $[\tau_{301}, \ldots, \tau_{600}]$*), observed frequencies of training data (i.e.* $[\tau_1, \ldots, \tau_{300}]$*), PF-based average probability forecast for test data, and MLE-based average probability forecast. The Brier scores are also reported in the labels.*