2nd Symposium on Advances in Approximate Bayesian Inference, 2019 1–11

Global Approximate Inference via Local Linearisation for Temporal Gaussian Processes

William J. Wilkinson WILLIAM.WILKINSON@AALTO.FI
Aalto University
Paul E. Chang PAUL.CHANG@AALTO.FI
Aalto University
Michael Riis Andersen MIRI@DTU.DK
Technical University of Denmark
Arno Solin ARNO.SOLIN@AALTO.FI
Aalto University

Abstract

The extended Kalman filter (EKF) is a classical signal processing algorithm which performs efficient approximate Bayesian inference in non-conjugate models by linearising the local measurement function, avoiding the need to compute intractable integrals when calculating the posterior. In some cases the EKF outperforms methods which rely on cubature to solve such integrals, especially in time-critical real-world problems. The drawback of the EKF is its local nature, whereas state-of-the-art methods such as variational inference or expectation propagation (EP) are considered global approximations. We formulate power EP as a nonlinear Kalman filter, before showing that linearisation results in a globally iterated algorithm that exactly matches the EKF on the first pass through the data, and iteratively improves the linearisation on subsequent passes. An additional benefit is the ability to calculate the limit as the EP power tends to zero, which removes the instability of the EP-like algorithm. The resulting inference scheme solves non-conjugate temporal Gaussian process models in linear time, $\mathcal{O}(n)$, and in closed form.

1. Introduction

Temporal Gaussian process (GP, Rasmussen and Williams, 2006) models can be solved in linear computational scaling, $\mathcal{O}(n)$, in the number of data n (Hartikainen and Särkkä, 2010). However, non-conjugate (*i.e.*, non-Gaussian likelihood) GP models introduce a computational problem in that they generally involve approximating intractable integrals in order to update the posterior distribution when data is observed. The most common numerical method used in such scenarios is sigma-point integration (Kokkala et al., 2016), with Gauss–Hermite cubature being a popular way to choose the sigma-point locations and weights. A drawback of this method is that the number of cubature points scales exponentially with the dimensionality d. Lower-order sigma-point methods allow accuracy to be traded off for scalability, for example the unscented transform (which forms the basis for the unscented Kalman filter, see Särkkä, 2013) requires only $2d + 1$ cubature points.

One significant alternative to cubature methods is linearisation. Although such an approach has gone out of fashion lately, García-Fernández et al. (2015) showed that a globally

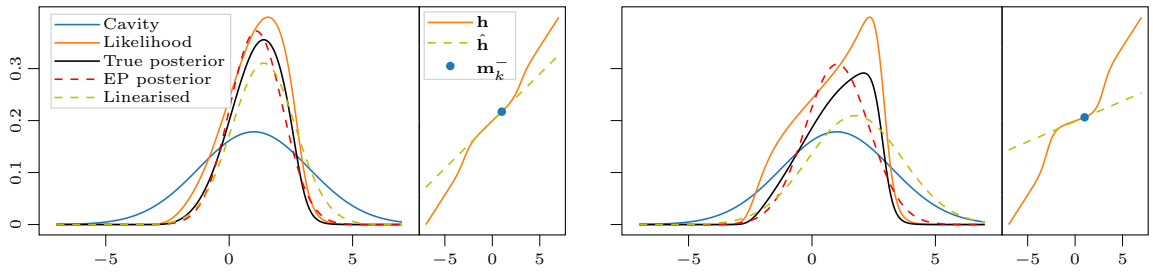

Figure 1: When the measurement function $\mathbf{h}$ is approximately linear in the region of the cavity $\mathrm{N}(\mathbf{m}_k^-, \mathbf{P}_k^-)$, (**left**), linearisation $\hat{\mathbf{h}}$ results in a good approximation to the posterior. When $\mathbf{h}$ is highly nonlinear (**right**), linearisation results in a crude posterior approximation.

iterated version of the statistically linearised filter (SLF, Särkkä, 2013), which performs linearisation w.r.t. the posterior rather than the prior, performs in line with expectation propagation (EP, Minka, 2001) in many modelling scenarios, whilst also providing local convergence guarantees (Appendix D explains the connection to our proposed method). Crucially, linearisation guarantees that the integrals required to calculate the posterior have a closed form solution, which results in significant computational savings if d is large.

Motivated by these observations, and with the aim of illustrating the connections between classical filtering methods and EP, we formulate power EP (PEP, Minka, 2004) as a Gaussian filter parametrised by a set of local likelihood approximations. The linearisations used to calculate these approximations are then refined during multiple passes through the data. We show that a single iteration of our approach is identical to the extended Kalman filter (EKF, Jazwinski, 1970), and furthermore that we are able to calculate exactly the limit as the EP power tends to zero, since there are no longer any intractable integrals that depend on the power. The result is a global approximate inference algorithm for temporal GPs that is efficient and stable, easy to implement, scales to problems with large data and high-dimensional latent states, and consistently outperforms the EKF.

2. Unifying power EP and the extended Kalman filter for temporal GPs

We consider non-conjugate (*i.e.*, non-Gaussian likelihood) Gaussian process models with one-dimensional inputs t (*i.e.*, time) which have a dual kernel (*left*) and discrete state space (*right*) form (Särkkä et al., 2013),

$$\begin{aligned} \mathbf{f}(t) &\sim \mathcal{GP}\big(\mathbf{0},\, \mathbf{K}_{\boldsymbol{\theta}}(t,t')\big), & \mathbf{x}_k &= \mathbf{A}_{\boldsymbol{\theta},k}\mathbf{x}_{k-1} + \mathbf{q}_k, \\ \mathbf{y}_k &\sim p(\mathbf{y}_k \,|\, \mathbf{f}(t_k)) & \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{r}_k) \end{aligned} \tag{1}$$

$\mathbf{y}_k \in \mathbb{R}^a$ are observations, $\mathbf{f}(t) = \big(f^{(1)}(t), \ldots, f^{(d)}(t)\big)^\top \in \mathbb{R}^d$ are GPs, $\mathbf{x}_k = \big(\mathbf{x}_k^{(1)}, \ldots, \mathbf{x}_k^{(d)}\big)^\top \in \mathbb{R}^s$ is the latent state vector containing the GP dynamics. Each $\mathbf{x}_k^{(i)}$ contains the state dynamics for one latent GP, for example a Matérn-$5/2$ GP prior is modelled with $\mathbf{x}_k^{(i)} = \big(f^{(i)}(t_k),\, \dot{f}^{(i)}(t_k),\, \ddot{f}^{(i)}(t_k)\big)^\top$. The hyerparameters $\boldsymbol{\theta}$ of the kernel $\mathbf{K}_{\boldsymbol{\theta}}$ determine the state transition matrix $\mathbf{A}_{\boldsymbol{\theta},k}$ and the process noise $\mathbf{q}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}_{\boldsymbol{\theta},k})$. The measurement model $\mathbf{h}(\mathbf{x}_k, \mathbf{r}_k)$ is a (nonlinear) function of $\mathbf{x}_k$ and the observation noise $\mathbf{r}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{R}_k)$.

Our aim is to calculate the posterior over the latent states, $p(\mathbf{x}_k \,|\, \mathbf{y}_1, \ldots, \mathbf{y}_n)$ for $k < n$, otherwise known as the *smoothing* solution, which can be obtained via application of a

Gaussian filter (to obtain the *filtering* solution $p(\mathbf{x}_k \,|\, \mathbf{y}_1, \ldots, \mathbf{y}_k)$) followed by a Gaussian smoother. If $\mathbf{h}(\cdot)$ is linear then the Kalman filter and Rauch–Tung–Striebel (RTS, Särkkä, 2013) smoother return the optimal solution.

Gaussian filtering and smoothing As with most approximate inference methods, we approximate the filtering distributions with Gaussians, $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:k}) \approx \mathrm{N}(\mathbf{x}_k;\, \mathbf{m}_k, \mathbf{P}_k)$. The prediction step remains the same as in the standard Kalman filter, with the resulting distribution acting as the EP cavity on the forward (filtering) pass: $\mathbf{m}_k^{(\mathrm{cav})} = \mathbf{A}_{\boldsymbol{\theta},k}\mathbf{m}_{k-1}$, and $\mathbf{P}_k^{(\mathrm{cav})} = \mathbf{A}_{\boldsymbol{\theta},k}\mathbf{P}_{k-1}\mathbf{A}_{\boldsymbol{\theta},k}^\top + \mathbf{Q}_{\boldsymbol{\theta},k}$.

To account for the non-Gaussian likelihood in the update step we follow Nickisch et al. (2018), introducing an intermediary step in which the parameters of the approximate likelihoods, $\mathrm{N}(\mathbf{x}_k; \mathbf{m}_k^{(\mathrm{site})}, \mathbf{P}_k^{(\mathrm{site})}) \approx p(\mathbf{y}_k \,|\, \mathbf{f}(t_k))$, are set via a *moment matching* procedure and stored before continuing with the Kalman updates. This PEP formulation, with power α, makes use of the fact that the required moments can be calculated via the derivatives of the log-normaliser, $\mathcal{Z}_k$, of the tilted distribution (see Seeger, 2005), giving

$$\begin{aligned}
&\mathcal{L}_k = \log \mathcal{Z}_k = \log \mathbb{E}_{\mathrm{N}(\mathbf{x}_k;\, \mathbf{m}_k^{(\mathrm{cav})}, \mathbf{P}_k^{(\mathrm{cav})})}\left[p(\mathbf{y}_k \,|\, \mathbf{f}(t_k))^\alpha\right], \\
&\mathbf{m}_k^{(\mathrm{site})} = \mathbf{m}_k^{(\mathrm{cav})} - \left(\frac{\mathrm{d}^2\mathcal{L}_k}{\mathrm{d}\mathbf{m}_k^{(\mathrm{cav})^2}}\right)^{-1} \frac{\mathrm{d}\mathcal{L}_k}{\mathrm{d}\mathbf{m}_k^{(\mathrm{cav})}}, \quad \mathbf{P}_k^{(\mathrm{site})} = \alpha\left(-\mathbf{P}_k^{(\mathrm{cav})} - \left(\frac{\mathrm{d}^2\mathcal{L}_k}{\mathrm{d}\mathbf{m}_k^{(\mathrm{cav})^2}}\right)^{-1}\right)
\end{aligned} \tag{2}$$

After the mean and covariance of our new likelihood approximation have been calculated, we can proceed with a modified set of linear Kalman filter updates,

$$\begin{aligned}
&\mathbf{S}_k = \mathbf{P}_k^{(\mathrm{cav})} + \mathbf{P}_k^{(\mathrm{site})}, && \mathbf{K}_k = \mathbf{P}_k^{(\mathrm{cav})}\mathbf{S}_k^{-1}, \\
&\mathbf{m}_k = \mathbf{m}_k^{(\mathrm{cav})} + \mathbf{K}_k(\mathbf{m}_k^{(\mathrm{site})} - \mathbf{m}_k^{(\mathrm{cav})}), && \mathbf{P}_k = \mathbf{P}_k^{(\mathrm{cav})} - \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^\top.
\end{aligned} \tag{3}$$

As in Wilkinson et al. (2019), we augment the standard RTS smoother with another moment matching step where the cavity distribution is calculated by removing (a fraction α of) the local likelihood from the marginal smoothing distribution $p(\mathbf{x}_k \,|\, \mathbf{y}_{1:n}) = \mathrm{N}(\mathbf{m}_k^s, \mathbf{P}_k^s)$,

$$\mathbf{P}_k^{(\mathrm{cav})} = \left(\left(\mathbf{P}_k^s\right)^{-1} - \alpha\left(\mathbf{P}_k^{(\mathrm{site})}\right)^{-1}\right)^{-1}, \quad \mathbf{m}_k^{(\mathrm{cav})} = \mathbf{P}_k^{(\mathrm{cav})}\left(\left(\mathbf{P}_k^s\right)^{-1}\mathbf{m}_k^s - \alpha\left(\mathbf{P}_k^{(\mathrm{site})}\right)^{-1}\mathbf{m}_k^{(\mathrm{site})}\right) \tag{4}$$

Moment matching is again performed via Eq. (2) using this new cavity. The likelihood parameters, $\mathbf{m}_k^{(\mathrm{site})}$, $\mathbf{P}_k^{(\mathrm{site})}$, are stored to be used on the next forward (filtering) pass.

Moment matching has a closed form solution after linearisation The computational saving in our approach comes from noticing that when $\mathbf{h}(\cdot)$ is linear, $\mathcal{Z}_k$ can be calculated in closed form. Using a first-order Taylor series expansion about the mean $\mathbf{m}_k^{(\mathrm{cav})}$ we obtain the approximation $\mathbf{h}(\mathbf{x}_k, \mathbf{r}_k) \approx \mathbf{J}_{\mathbf{x}_k}(\mathbf{x}_k - \mathbf{m}_k^{(\mathrm{cav})}) + \mathbf{h}(\mathbf{m}_k^{(\mathrm{cav})}, \mathbf{0}) + \mathbf{J}_{\mathbf{r}_k}\mathbf{r}_k$, which is a linear function of the state, $\mathbf{x}_k$, and Gaussian noise, $\mathbf{r}_k$, such that $p(\mathbf{y}_k \,|\, \mathbf{f}(t_k)) \approx \mathrm{N}(\mathbf{y}_k; \mathbf{g}(\mathbf{x}_k), \hat{\mathbf{R}}_k)$, where $\hat{\mathbf{R}}_k = \mathbf{J}_{\mathbf{r}_k}\mathbf{R}_k\mathbf{J}_{\mathbf{r}_k}^\top$ and $\mathbf{g}(\mathbf{x}_k) = \mathbf{J}_{\mathbf{x}_k}(\mathbf{x}_k - \mathbf{m}_k^{(\mathrm{cav})}) + \mathbf{h}(\mathbf{m}_k^{(\mathrm{cav})}, \mathbf{0})$. Here $\mathbf{J}_{\mathbf{x}_k} = \mathbf{J}_{\mathbf{x}}|_{\mathbf{m}_k^{(\mathrm{cav})}, \mathbf{0}} \in$

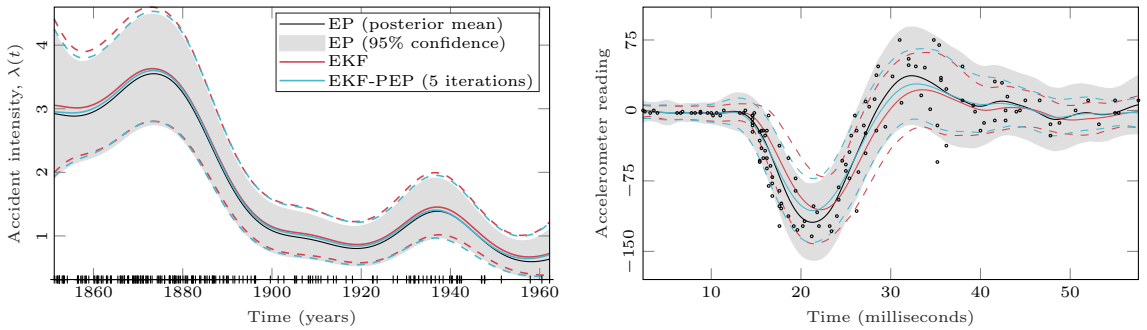

Figure 2: Two tasks with non-conjugate GP models. The coal mining accident task (log-Gaussian Cox process, **left**) is well approximated via linearisation, and iterating improves the match to the EP posterior. Linearisation in the motorcycle crash task (heteroscedastic noise, **right**) is a crude approximation, but iterating still improves the posterior.

$\mathbb{R}^{a\times s}$ and $\mathbf{J}_{\mathbf{r}_k} = \mathbf{J}_{\mathbf{r}}|_{\mathbf{m}_k^{(\text{cav})},\mathbf{0}} \in \mathbb{R}^{a\times a}$ are the Jacobian of $\mathbf{h}(\cdot)$ evaluated at the mean w.r.t. $\mathbf{x}_k$ and $\mathbf{r}_k$ respectively. This new Gaussian form means the moment matching step becomes,

$$\mathcal{L}_k = \log \mathbb{E}_{\mathrm{N}(\mathbf{x}_k;\, \mathbf{m}_k^{(\text{cav})}, \mathbf{P}_k^{(\text{cav})})}\left[\mathrm{N}\big(\mathbf{y}_k;\, \mathbf{g}(\mathbf{x}_k), \hat{\mathbf{R}}_k\big)^{\alpha}\right] = c + \log \mathrm{N}\big(\mathbf{y}_k;\, \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0}), \boldsymbol{\Sigma}_k\big), \quad (5)$$

where $\boldsymbol{\Sigma}_k = \frac{1}{\alpha}\hat{\mathbf{R}}_k + \mathbf{J}_{\mathbf{x}_k}\mathbf{P}_k^{(\text{cav})}\mathbf{J}_{\mathbf{x}_k}^{\top}$. $\mathbf{J}_{\mathbf{x}_k}$ represents the slope of a linear function, hence its derivative w.r.t. $\mathbf{m}_k^{(\text{cav})}$ is zero (see Deisenroth and Mohamed, 2012, for discussion). Therefore,

$$\frac{\mathrm{d}\mathcal{L}_k}{\mathrm{d}\mathbf{m}_k^{(\text{cav})}} = \mathbf{J}_{\mathbf{x}_k}^{\top}\boldsymbol{\Sigma}_k^{-1}(\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})), \qquad \frac{\mathrm{d}^2\mathcal{L}_k}{\mathrm{d}\mathbf{m}_k^{(\text{cav})2}} = -\mathbf{J}_{\mathbf{x}_k}^{\top}\boldsymbol{\Sigma}_k^{-1}\mathbf{J}_{\mathbf{x}_k}. \quad (6)$$

Now we update the approximate likelihood in closed form (Appendix B gives the derivation),

$$\begin{aligned}
\mathbf{P}_k^{(\text{site})} &= \left(\mathbf{J}_{\mathbf{x}_k}^{\top}\hat{\mathbf{R}}_k^{-1}\mathbf{J}_{\mathbf{x}_k}\right)^{-1}, \\
\mathbf{m}_k^{(\text{site})} &= \mathbf{m}_k^{(\text{cav})} + \left(\mathbf{P}_k^{(\text{site})} + \alpha\mathbf{P}_k^{(\text{cav})}\right)\mathbf{J}_{\mathbf{x}_k}^{\top}\left(\hat{\mathbf{R}}_k + \alpha\mathbf{J}_{\mathbf{x}_k}\mathbf{P}_k^{(\text{cav})}\mathbf{J}_{\mathbf{x}_k}^{\top}\right)^{-1}(\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})).
\end{aligned} \quad (7)$$

The result when we use Eq. (7) (with $\alpha = 1$) to modify the filter updates, Eq. (3), is *exactly* the EKF (see Appendix C for the proof). Additionally, since these updates are now available in closed form, a variational free energy method ($\alpha \to 0$, see Bui et al., 2017) is simple to implement and doesn't require any matrix subtractions and inversions in Eq. (4), which can be costly and unstable. Taking $\alpha \to 0$ prior to linearisation is not possible because the intractable integrals also depend on α. Appendix A describes our full iterative algorithm.

3. Empirical analysis and discussion

In Fig. 2, we compare our approach (EKF-PEP, $\alpha = 1$) to EP and the EKF on two non-conjugate GP tasks (see Appendix E for the full formulations). Whilst our method is suited to large datasets, we focus here on small time series for ease of comparison. In the left-hand

plot, a log-Gaussian Cox process (approximated with a Poisson model for 200 equal time interval bins) is used to model the intensity of coal mining accidents. EKF-PEP and the EKF match the EP posterior well, with EKF-PEP obtaining an even tighter match to both the mean and marginal variances. The right-hand plot shows a similar comparison for 133 accelerometer readings in a simulated motorcycle crash, using a heteroscedastic noise model. Linearisation in this model is a crude approximation to the true likelihood, but we observe that iteratively refining the linearisation vastly improves the posterior is some regions.

This new perspective on linearisation in approximate inference unifies the PEP and EKF paradigms for temporal data, and provides an improvement to the EKF that requires no additional implementation effort. Key areas for further exploration are the effect of adjusting α (*i.e.*, changing the cavity and the linearisation point), and the use of statistical linearisation as an alternative method for obtaining the local approximations.

Acknowledgments

We acknowledge funding from the Academy of Finland (grant numbers 308640 and 324345) and from Innovation Fund Denmark (grant number 8057-00036A).

References

Thang D. Bui, Josiah Yan, and Richard E. Turner. A unifying framework for Gaussian process pseudo-point approximations using power expectation propagation. *Journal of Machine Learning Research*, 2017.

Marc Deisenroth and Shakir Mohamed. Expectation propagation in Gaussian process dynamical systems. In *Advances in Neural Information Processing Systems (NIPS)*, 2012.

Ángel F. García-Fernández, Lennart Svensson, Mark R Morelande, and Simo Särkkä. Posterior linearization filter: Principles and implementation using sigma points. *IEEE transactions on signal processing*, 2015.

Jouni Hartikainen and Simo Särkkä. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2010.

Andrew H. Jazwinski. *Stochastic Processes and Filtering Theory*. Academic Press, 1970.

Juho Kokkala, Arno Solin, and Simo Särkkä. Sigma-point filtering and smoothing based parameter estimation in nonlinear dynamic systems. *Journal of Advances in Information Fusion*, 2016.

Thomas P. Minka. Expectation propagation for approximate Bayesian inference. In *Uncertainty in Artificial Intelligence (UAI)*, 2001.

Thomas P. Minka. Power EP. Technical report, 2004.

Hannes Nickisch, Arno Solin, and Alexander Grigorievskiy. State space Gaussian processes with non-Gaussian likelihood. In *International Conference on Machine Learning (ICML)*, 2018.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Simo Särkkä. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

Simo Särkkä, Arno Solin, and Jouni Hartikainen. Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing. *IEEE Signal Processing Magazine*, 2013.

Matthias Seeger. Expectation propagation for exponential families. Technical report, 2005.

William J. Wilkinson, Michael Riis Andersen, Joshua D. Reiss, Dan Stowell, and Arno Solin. End-to-end probabilistic inference for nonstationary audio analysis. In *International Conference on Machine Learning (ICML)*, 2019.

Appendix A. The proposed globally iterated EKF-PEP algorithm

Algorithm 1 Globally iterated extended Kalman filter with power EP-style updates

Input: $\{t_k, \mathbf{y}_k\}_{k=1}^n$, $\mathbf{A}_k$, $\mathbf{Q}_k$, $\mathbf{R}_k$, $\mathbf{P}_\infty$ — data and discretised state space model
$\mathbf{h}$, $\mathbf{H}$, $\mathbf{J}_\mathbf{x}$, $\mathbf{J}_\mathbf{r}$, α — measurement model, Jacobian and EP power
$\mathbf{m}_0 \leftarrow \mathbf{0}$, $\mathbf{P}_0 \leftarrow \mathbf{P}_\infty$, $\mathbf{e}_{1:n} = \mathbf{0}$ — initial state
while not converged **do** — iterated EP-style loop
  for $k = 1$ **to** n **do** — forward pass (FILTERING)
    $\mathbf{m}_k \leftarrow \mathbf{A}_k \, \mathbf{m}_{k-1}$; $\quad \mathbf{P}_k \leftarrow \mathbf{A}_k \, \mathbf{P}_{k-1} \, \mathbf{A}_k^\top + \mathbf{Q}_k$ — predict
    if has label $\mathbf{y}_k$ **then**
      $\mathbf{v}_k = \mathbf{y}_k - \mathbf{h}(\mathbf{m}_k, \mathbf{0})$ — residual
      if first filter iteration **then**
        $\mathbf{J}_{\mathbf{x}_k} \leftarrow \mathbf{J}_\mathbf{x}|_{\mathbf{m}_k, \mathbf{0}}$, $\quad \mathbf{J}_{\mathbf{r}_k} \leftarrow \mathbf{J}_\mathbf{r}|_{\mathbf{m}_k, \mathbf{0}}$ — evaluate Jacobian
        $\mathbf{P}_k^{(\text{site})} \leftarrow \left(\mathbf{J}_{\mathbf{x}_k}^\top \left(\mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top \right)^{-1} \mathbf{J}_{\mathbf{x}_k} \right)^{-1}$ — match moments
        $\mathbf{m}_k^{(\text{site})} \leftarrow \mathbf{m}_k + \left(\mathbf{P}_k^{(\text{site})} + \alpha \mathbf{P}_k \right) \mathbf{J}_{\mathbf{x}_k}^\top \left(\mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top + \alpha \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k \mathbf{J}_{\mathbf{x}_k}^\top \right)^{-1} \mathbf{v}_k$
      end if
      $\mathbf{S}_k \leftarrow \mathbf{P}_k + \mathbf{P}_k^{(\text{site})}$; $\quad \mathbf{K}_k \leftarrow \mathbf{P}_k \mathbf{S}_k^{-1}$ — innovation & gain
      $\mathbf{m}_k \leftarrow \mathbf{m}_k + \mathbf{K}_k (\mathbf{m}_k^{(\text{site})} - \mathbf{m}_k)$; $\quad \mathbf{P}_k \leftarrow \mathbf{P}_k - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top$ — update
      $\mathbf{E}_k = \mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top + \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k \mathbf{J}_{\mathbf{x}_k}^\top$
      $\mathbf{e}_k = \frac{1}{2} \log |2\pi \mathbf{E}_k| + \frac{1}{2} \mathbf{v}_k^\top \mathbf{E}_k^{-1} \mathbf{v}_k$ — energy
    end if
  end for
  for $k = n - 1$ **to** 1 **do** — backward pass (SMOOTHING)
    $\mathbf{G}_k \leftarrow \mathbf{P}_k \, \mathbf{A}_{k+1}^\top \left(\mathbf{A}_{k+1} \, \mathbf{P}_k \, \mathbf{A}_{k+1}^\top + \mathbf{Q}_{k+1} \right)^{-1}$ — smoothing gain
    $\mathbf{m}_k \leftarrow \mathbf{m}_k + \mathbf{G}_k \left(\mathbf{m}_{k+1} - \mathbf{A}_{k+1} \, \mathbf{m}_k \right)$ — update
    $\mathbf{P}_k \leftarrow \mathbf{P}_k + \mathbf{G}_k \left(\mathbf{P}_{k+1} - \mathbf{A}_{k+1} \, \mathbf{P}_k \, \mathbf{A}_{k+1}^\top - \mathbf{Q}_{k+1} \right) \mathbf{G}_k^\top$
    if has label $\mathbf{y}_k$ **then**
      $\mathbf{P}_k^{(\text{cav})} = \left(\mathbf{P}_k^{-1} - \alpha \left(\mathbf{P}_k^{(\text{site})} \right)^{-1} \right)^{-1}$ — remove site to get cavity
      $\mathbf{m}_k^{(\text{cav})} = \mathbf{P}_k^{(\text{cav})} \left(\mathbf{P}_k^{-1} \mathbf{m}_k - \alpha \left(\mathbf{P}_k^{(\text{site})} \right)^{-1} \mathbf{m}_k^{(\text{site})} \right)$
      $\mathbf{J}_{\mathbf{x}_k} \leftarrow \mathbf{J}_\mathbf{x}|_{\mathbf{m}_k^{(\text{cav})}, \mathbf{0}}$, $\quad \mathbf{J}_{\mathbf{r}_k} \leftarrow \mathbf{J}_\mathbf{r}|_{\mathbf{m}_k^{(\text{cav})}, \mathbf{0}}$ — evaluate Jacobian
      $\mathbf{v}_k = \mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})$ — residual
      $\mathbf{P}_k^{(\text{site})} \leftarrow \left(\mathbf{J}_{\mathbf{x}_k}^\top \left(\mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top \right)^{-1} \mathbf{J}_{\mathbf{x}_k} \right)^{-1}$ — match moments
      $\mathbf{m}_k^{(\text{site})} \leftarrow \mathbf{m}_k^{(\text{cav})} + \left(\mathbf{P}_k^{(\text{site})} + \alpha \mathbf{P}_k^{(\text{cav})} \right) \mathbf{J}_{\mathbf{x}_k}^\top \left(\mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top + \alpha \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top \right)^{-1} \mathbf{v}_k$
    end if
  end for
end while
Return: $\mathbb{E}[\mathbf{f}(t_k)] = \mathbf{H}\mathbf{m}_k$; $\mathbb{V}[\mathbf{f}(t_k)] = \mathbf{H}\mathbf{P}_k\mathbf{H}^\top$ — $\mathbf{H}$ extracts 1st-order terms
$\log p(\mathbf{y} \mid \boldsymbol{\theta}) \simeq -\sum_{k=1}^n \mathbf{e}_k$ — log marginal likelihood

Appendix B. Closed form site updates

Here we derive in full the closed form site updates after linearisation. Plugging the derivatives from Eq. (6) into the updates in Eq. (2) we get,

$$\begin{aligned}\mathbf{m}_k^{(\text{site})} &= \mathbf{m}_k^{(\text{cav})} + \left(\mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} \mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})), \\ \mathbf{P}_k^{(\text{site})} &= \alpha \left(-\mathbf{P}_k^{(\text{cav})} + \left(\mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1}\right).\end{aligned} \tag{8}$$

By the matrix inversion lemma, and with $\hat{\mathbf{R}}_k = \mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top$,

$$\boldsymbol{\Sigma}_k^{-1} = \alpha \hat{\mathbf{R}}_k^{-1} - \alpha \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k} \left(\mathbf{P}_k^{(\text{cav})^{-1}} + \mathbf{J}_{\mathbf{x}_k}^\top \alpha \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} \mathbf{J}_{\mathbf{x}_k}^\top \alpha \mathbf{R}_k^{-1}, \tag{9}$$

so that

$$\mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{J}_{\mathbf{x}_k} = \mathbf{W}_k - \mathbf{W}_k \left(\mathbf{P}_k^{(\text{cav})^{-1}} + \mathbf{W}_k\right)^{-1} \mathbf{W}_k, \tag{10}$$

where $\mathbf{W}_k = \mathbf{J}_{\mathbf{x}_k}^\top \alpha \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}$. Applying the matrix inversion lemma for a second time we obtain

$$\begin{aligned}\left(\mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} &= \mathbf{W}_k^{-1} - \mathbf{W}_k^{-1} \mathbf{W}_k \left(\mathbf{W}_k \mathbf{W}_k^{-1} \mathbf{W}_k - \left(\mathbf{P}_k^{(\text{cav})^{-1}} + \mathbf{W}_k\right)\right)^{-1} \mathbf{W}_k \mathbf{W}_k^{-1} \\ &= \mathbf{W}_k^{-1} + \mathbf{P}_k^{(\text{cav})} \\ &= \frac{1}{\alpha} \left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} + \mathbf{P}_k^{(\text{cav})}.\end{aligned} \tag{11}$$

We can also write

$$\begin{aligned}\left(\mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} \mathbf{J}_{\mathbf{x}_k}^\top \boldsymbol{\Sigma}_k^{-1} &= \left(\frac{1}{\alpha} \left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} + \mathbf{P}_k^{(\text{cav})}\right) \mathbf{J}_{\mathbf{x}_k}^\top \left(\frac{1}{\alpha} \hat{\mathbf{R}}_k + \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top\right)^{-1} \\ &= \left(\left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} + \alpha \mathbf{P}_k^{(\text{cav})}\right) \mathbf{J}_{\mathbf{x}_k}^\top \left(\hat{\mathbf{R}}_k + \alpha \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top\right)^{-1}.\end{aligned} \tag{12}$$

Together the above calculations give the approximate site mean and covariance as

$$\begin{aligned}\mathbf{P}_k^{(\text{site})} &= \left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1}, \\ \mathbf{m}_k^{(\text{site})} &= \mathbf{m}_k^{(\text{cav})} + \left(\mathbf{P}_k^{(\text{site})} + \alpha \mathbf{P}_k^{(\text{cav})}\right) \mathbf{J}_{\mathbf{x}_k}^\top \left(\hat{\mathbf{R}}_k + \alpha \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top\right)^{-1} (\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})).\end{aligned} \tag{13}$$

Appendix C. Analytical linearisation in EP ($\alpha = 1$) results in an iterated version of the EKF

Here we prove that a single pass of the proposed EP-style algorithm with linearisation is exactly equivalent to the EKF. Plugging the closed form site updates, Eq. (7), with $\alpha = 1$

(since the filter predictions can be interpreted as the cavity with the full site removed), into our modified Kalman filter update equations, Eq. (3), we get a new set of Kalman updates in which the latent noise terms are determined by scaling the observation noise with the Jacobian of the state:

$$\begin{aligned}
\mathbf{S}_k &= \mathbf{P}_k^{(\text{cav})} + \left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1}, \\
\mathbf{K}_k &= \mathbf{P}_k^{(\text{cav})} \mathbf{S}_k^{-1}, \\
\mathbf{m}_k &= \mathbf{m}_k^{(\text{cav})} + \mathbf{K}_k \mathbf{S}_k \mathbf{J}_{\mathbf{x}_k}^\top \left(\hat{\mathbf{R}}_k + \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top\right)^{-1} (\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})), \\
\mathbf{P}_k &= \mathbf{P}_k^{(\text{cav})} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top.
\end{aligned} \tag{14}$$

This can be rewritten to explicitly show that there are two innovation covariance terms, $\mathbf{S}_k$ and $\hat{\mathbf{S}}_k$, which act on the state mean and covariance separately:

Linearised update step:

$$\begin{aligned}
\hat{\mathbf{S}}_k &= \mathbf{P}_k^{(\text{cav})} + \left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1}, \\
\mathbf{S}_k &= \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top + \hat{\mathbf{R}}_k, \\
\hat{\mathbf{K}}_k &= \mathbf{P}_k^{(\text{cav})} \hat{\mathbf{S}}_k^{-1}, \\
\mathbf{K}_k &= \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top \mathbf{S}_k^{-1}, \\
\mathbf{m}_k &= \mathbf{m}_k^{(\text{cav})} + \mathbf{K}_k (\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})), \\
\mathbf{P}_k &= \mathbf{P}_k^{(\text{cav})} - \hat{\mathbf{K}}_k \hat{\mathbf{S}}_k \hat{\mathbf{K}}_k^\top.
\end{aligned} \tag{15}$$

Now we calculate the inverse of $\hat{\mathbf{S}}_k$:

$$\begin{aligned}
\hat{\mathbf{S}}_k^{-1} &= \left(\mathbf{P}_k^{(\text{cav})} + \left(\mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1}\right)^{-1} \\
&= \mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k} - \mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k} \left(\mathbf{P}_k^{(\text{cav})^{-1}} + \mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} \mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}
\end{aligned} \tag{16}$$

and the inverse of $\mathbf{S}_k$:

$$\begin{aligned}
\mathbf{S}_k^{-1} &= \left(\mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top + \hat{\mathbf{R}}_k\right)^{-1} \\
&= \hat{\mathbf{R}}_k^{-1} - \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k} \left(\mathbf{P}_k^{(\text{cav})^{-1}} + \mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1} \mathbf{J}_{\mathbf{x}_k}\right)^{-1} \mathbf{J}_{\mathbf{x}_k}^\top \hat{\mathbf{R}}_k^{-1}
\end{aligned} \tag{17}$$

which shows that

$$\hat{\mathbf{S}}_k^{-1} = \mathbf{J}_{\mathbf{x}_k}^\top \mathbf{S}_k^{-1} \mathbf{J}_{\mathbf{x}_k}, \tag{18}$$

and hence, recalling that $\hat{\mathbf{R}}_k = \mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top$, Eq. (15) simplifies to give exactly the extended Kalman filter updates:

EKF update step:

$$
\begin{aligned}
\mathbf{S}_k &= \mathbf{J}_{\mathbf{x}_k} \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top + \mathbf{J}_{\mathbf{r}_k} \mathbf{R}_k \mathbf{J}_{\mathbf{r}_k}^\top, \\
\mathbf{K}_k &= \mathbf{P}_k^{(\text{cav})} \mathbf{J}_{\mathbf{x}_k}^\top \mathbf{S}_k^{-1}, \\
\mathbf{m}_k &= \mathbf{m}_k^{(\text{cav})} + \mathbf{K}_k (\mathbf{y}_k - \mathbf{h}(\mathbf{m}_k^{(\text{cav})}, \mathbf{0})), \\
\mathbf{P}_k &= \mathbf{P}_k^{(\text{cav})} - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^\top.
\end{aligned}
\tag{19}
$$

Appendix D. Connection to posterior linearisation

Posterior linearisation (García-Fernández et al., 2015) is a filtering algorithm that iteratively refines local posterior approximations based on statistical linear regression (SLR), and can be seen as a globally iterated extension of the SLR filter (Särkkä, 2013). The idea is that the measurement function is linearised with respect to the posterior, rather than the prior, which is particularly beneficial when the measurement noise is small, such that the prior and posterior can have very different locations and variance. One drawback of using SLR is that it does not generally result in closed form updates, however it does provide local convergence guarantees.

We have shown in Section 2 that on the first filtering pass our proposed algorithm is equivalent to the EKF. However, the power EP formulation of the smoothing pass, Eq. (4), iteratively refines the approximate likelihood parameters in the *context* of the posterior (with a fraction of the local likelihood removed). Letting $\alpha \to 0$ during the cavity calculation in Eq. (4) implies that the expectations are now with respect to the full marginal posterior. This shows that PLF is a version of our algorithm in which $\alpha = 0$ and analytical linearisation is replaced with SLR.

This motivates the following observation: posterior linearisation is a variational free energy method in which the intractable integrals required for posterior calculation are solved via linearisation of the likelihood mean function. This is intuitive since the formulation of the PLF is based on minimizing local KL divergences.

The local convergence analysis in García-Fernández et al. (2015) depends on using SLR as the linearisation method and initialising the state sufficiently close to a fixed point. However, it now becomes apparent why both the PLF and our algorithm are generally more stable than EP: no covariance subtractions and inversions are necessary in calculating the cavity distribution, which avoids the possibility of negative-definite covariance matrices.

Appendix E. Full model formulations for Section 3

Log-Gaussian Cox process The coal mining dataset contains the dates of 191 coal mine explosions in Britain between the years 1851–1962, discretised into $n = 200$ equal time interval bins. We use a log-Gaussian Cox process to model this count data. Assuming the process has locally constant intensity in the subregions allows a Poisson likelihood to be used for each bin, $p(y_k \,|\, f_k^{(1)}) \approx \text{Poisson}\big(y_k \,|\, \lambda_k = \exp\big(f_k^{(1)}\big)\big)$, where we define $f_k^{(i)} = f^{(i)}(t_k)$. However, the Poisson is a discrete probability distribution and the EKF applies to continuous observations. Therefore we use a Gaussian approximation to the Poisson likelihood, noticing that the first

two moments of the Poisson distribution are equal to the intensity $\lambda_k = \exp\big(f_k^{(1)}\big)$,

$$\begin{aligned} f^{(1)}(t) &\sim \mathcal{GP}\big(0, \kappa_{\boldsymbol{\theta}_1}(t,t')\big), \\ y_k &\sim p(y_k \,|\, f_k^{(1)}) = \mathrm{N}\big(\exp\big(f_k^{(1)}\big),\, \exp\big(f_k^{(1)}\big)\big), \\ h(\mathbf{x}_k, r_k) &= \exp\big(f_k^{(1)}\big) + \exp\big(f_k^{(1)}/2\big) r_k. \end{aligned} \tag{20}$$

Heteroscedastic noise model The motorcycle crash experiment consists of 131 simulated readings from an accelerometer on a motorcycle helmet during impact. A single GP is not a good model for this data due to the heteroscedasticity of the observation noise, therefore it is common to model the noise separately. We model the process with one GP for the mean and another for the time varying observation noise. Letting $r_k \sim \mathrm{N}(0,1)$, we place a GP prior over $f^{(1)}$ and $f^{(2)}$, both with Matern-$3/2$ kernels,

$$\begin{aligned} &f^{(1)}(t) \sim \mathcal{GP}\big(0, \kappa_{\boldsymbol{\theta}_1}(t,t')\big), \qquad f^{(2)}(t) \sim \mathcal{GP}\big(0, \kappa_{\boldsymbol{\theta}_2}(t,t')\big), \\ &y_k \sim p(y_k \,|\, f_k^{(1)}, f_k^{(2)}) = \mathrm{N}(f_k^{(1)}, \phi\big(f_k^{(2)}\big)^2), \\ &h(\mathbf{x}_k, r_k) = f_k^{(1)} + \phi\big(f_k^{(2)}\big) r_k, \end{aligned} \tag{21}$$

where $\phi(z) = \log(1 + \exp(z))$. In practice a problem arises when linearising this likelihood model. Since the mean of $r_k = 0$, the Jacobian of the noise term disappears when evaluated at the mean regardless of the value of $f^{(2)}$. Hence we reformulate the model to improve identifiability,

$$\bar{h}(\mathbf{x}_k, r_k) = (y_k - f_k^{(1)})\phi\big(f_k^{(2)}\big)^{-1} - r_k = 0. \tag{22}$$

Fig. 3 plots a breakdown of the different components of the model, showing that our linearisation method performs similarly to PEP.

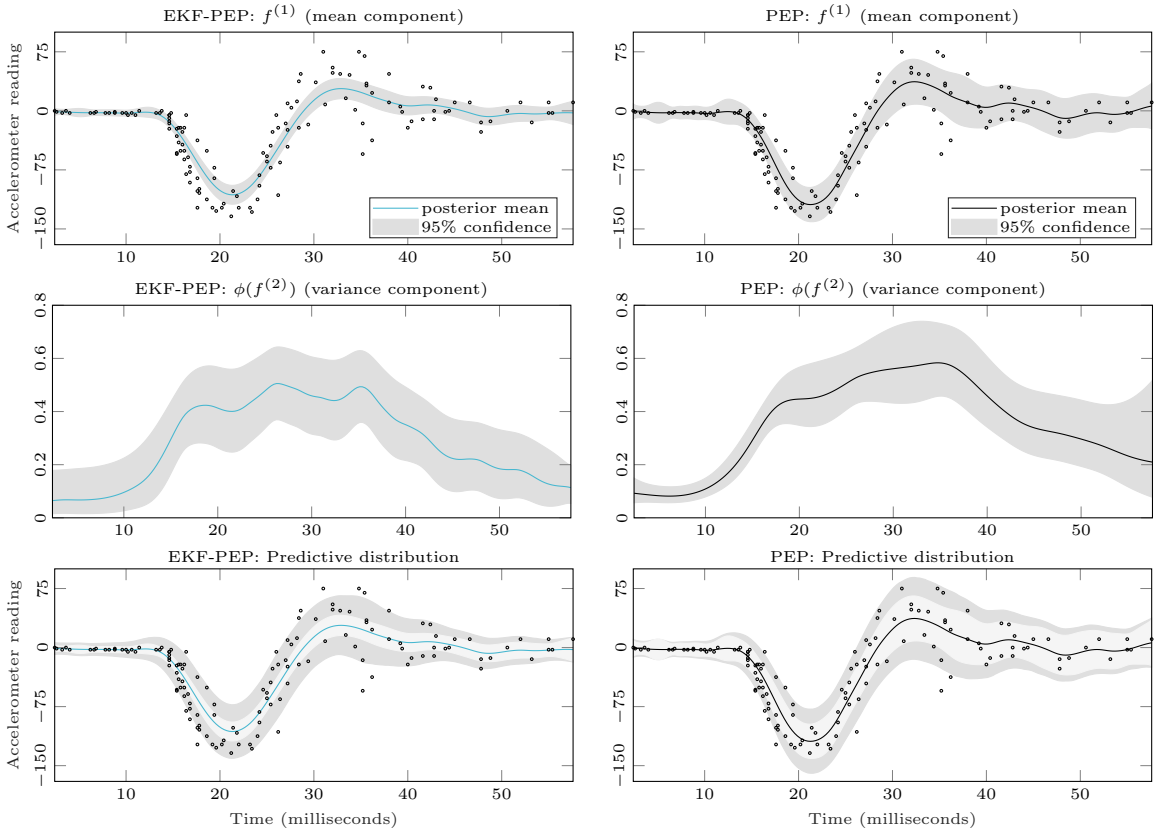

Figure 3: Results of the motorcycle crash experiment. **Left** is the EKF-PEP method and **right** is the PEP equivalent. The **top** plots are the posterior for $f^{(1)}(t)$ (the mean process), the **middle** plots show the posterior for $f^{(2)}(t)$ (the observation noise process), and the **bottom** plots are the full model.