Aalto University
School of Electrical Engineering

Stochastic (partial) differential equations and Gaussian processes

Simo Särkkä

Aalto University, Finland

Contents

Contents

Contents

Contents

Contents

Kernel vs. SPDE representations of GPs

GP model $\mathbf{x} \in \mathbb{R}^d, t \in \mathbb{R}$	Equivalent S(P)DE model
Spatial $k(\mathbf{x}, \mathbf{x}')$	SPDE model ($\mathcal{L}$ is an operator) $$\mathcal{L} \, f(\mathbf{x}) = w(\mathbf{x})$$
Temporal $k(t, t')$	State-space/SDE model $$\frac{d\mathbf{f}(t)}{dt} = \mathbf{A} \, \mathbf{f}(t) + \mathbf{L} \, w(t)$$
Spatio-temporal $k(\mathbf{x}, t; \mathbf{x}', t')$	Stochastic evolution equation $$\frac{\partial}{\partial t} \mathbf{f}(\mathbf{x}, t) = \mathcal{A}_x \, \mathbf{f}(\mathbf{x}, t) + \mathbf{L} \, w(\mathbf{x}, t)$$

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Why use S(P)DE solvers for GPs?

- The $O(n^3)$ computational complexity is a challenge.
- What do we get:
  - $O(n)$ state-space methods for SDEs/SPDEs.
  - Sparse approximations developed for SPDEs.
  - Reduced rank Fourier/basis function approximations.
  - Path to non-Gaussian processes.
- Downsides:
  - We often need to approximate.
  - Mathematics can become messy.

Contents

Ornstein-Uhlenbeck process

- The mean and covariance functions:

$$m(x) = 0$$
$$k(x, x') = \sigma^2 \exp(-\lambda |x - x'|)$$

- This has a *path representation* as a stochastic differential equation (SDE):

$$\frac{df(t)}{dt} = -\lambda f(t) + w(t).$$

where $w(t)$ is a white noise process with x relabeled as t.

- Ornstein–Uhlenbeck process is a Markov process.
- What does this actually mean $\Longrightarrow$ white board.

Ornstein-Uhlenbeck process

- The mean and covariance functions:

$$\begin{aligned} m(x) &= 0 \\ k(x, x') &= \sigma^2 \exp(-\lambda |x - x'|) \end{aligned}$$

- This has a *path representation* as a stochastic differential equation (SDE):

$$\frac{df(t)}{dt} = -\lambda\, f(t) + w(t).$$

  where $w(t)$ is a white noise process with x relabeled as t.
- Ornstein–Uhlenbeck process is a Markov process.
- What does this actually mean $\Longrightarrow$ white board.

Ornstein-Uhlenbeck process

- The mean and covariance functions:

$$
\begin{aligned}
m(x) &= 0 \\
k(x, x') &= \sigma^2 \exp(-\lambda |x - x'|)
\end{aligned}
$$

- This has a *path representation* as a stochastic differential equation (SDE):

$$
\frac{df(t)}{dt} = -\lambda\, f(t) + w(t).
$$

  where $w(t)$ is a white noise process with x relabeled as t.
- Ornstein–Uhlenbeck process is a Markov process.
- What does this actually mean $\Longrightarrow$ white board.

Ornstein-Uhlenbeck process

- The mean and covariance functions:

$$m(x) = 0$$
$$k(x, x') = \sigma^2 \exp(-\lambda |x - x'|)$$

- This has a *path representation* as a stochastic differential equation (SDE):

FT: $(i\omega)\hat{f} = -\lambda \hat{f} + \hat{w}$

$\hat{f} = \frac{\hat{w}}{\lambda + i\omega}$

$$\frac{df(t)}{dt} = -\lambda f(t) + w(t).$$

Spectral Density:

$S(\omega) = \frac{E[|\hat{w}|^2]}{\omega^2 + \lambda^2} = \frac{q}{\omega^2 + \lambda^2}$

where $w(t)$ is a white noise process with x relabeled as t.

$h(\tau) = \frac{1}{2\pi} \int \frac{q}{\omega^2 + \lambda^2} \exp(i\omega\tau) \, d\omega$

$\tau = t - t'$

$= \frac{q}{2\lambda} \exp(-\lambda|\tau|)$

Residual Transform.

$=$

- Ornstein–Uhlenbeck process is a Markov process.
- What does this actually mean $\Longrightarrow$ white board.

Ornstein-Uhlenbeck process (cont.)

- Consider a Gaussian process regression problem

$$
\begin{aligned}
f(x) &\sim \mathrm{GP}(0, \sigma^2 \exp(-\lambda |x - x'|)) \\
y_k &= f(x_k) + \varepsilon_k
\end{aligned}
$$

- This is equivalent to the state-space model

$$
\begin{aligned}
\frac{df(t)}{dt} &= -\lambda f(t) + w(t) \\
y_k &= f(t_k) + \varepsilon_k
\end{aligned}
$$

that is, with $f_k = f(t_k)$ we have a Gauss-Markov model

$$
\begin{aligned}
f_{k+1} &\sim p(f_{k+1} \mid f_k) \\
y_k &\sim p(y_k \mid f_k)
\end{aligned}
$$

- Solvable in $O(n)$ time using Kalman filter/smoother.

Ornstein-Uhlenbeck process (cont.)

- Consider a Gaussian process regression problem

$$\begin{aligned} f(x) &\sim \mathrm{GP}(0, \sigma^2 \exp(-\lambda |x - x'|)) \\ y_k &= f(x_k) + \varepsilon_k \end{aligned}$$

- This is equivalent to the state-space model

$$\begin{aligned} \frac{df(t)}{dt} &= -\lambda\, f(t) + w(t) \\ y_k &= f(t_k) + \varepsilon_k \end{aligned}$$

that is, with $f_k = f(t_k)$ we have a Gauss-Markov model

$$\begin{aligned} f_{k+1} &\sim p(f_{k+1} \mid f_k) \\ y_k &\sim p(y_k \mid f_k) \end{aligned}$$

- Solvable in $O(n)$ time using Kalman filter/smoother.

Ornstein-Uhlenbeck process (cont.)

- Consider a Gaussian process regression problem

$$f(x) \sim \mathrm{GP}(0, \sigma^2 \exp(-\lambda |x - x'|))$$
$$y_k = f(x_k) + \varepsilon_k$$

- This is equivalent to the state-space model

$$\frac{df(t)}{dt} = -\lambda\, f(t) + w(t)$$
$$y_k = f(t_k) + \varepsilon_k$$

$\times e^{\lambda T}$ (integrating factor)

$e^{\lambda T} \cdot \frac{df(t)}{dt} + \lambda f(t) e^{\lambda t} = w \cdot e^{\lambda t}$

$\frac{d}{dt}(e^{\lambda t} \cdot f) = e^{\lambda t} \cdot w \quad \Big| \int_{t_0}^{t}$

$e^{\lambda t} f(t) - e^{\lambda t_0} f(t_0) = \int_{t_0}^{t} e^{\lambda s} w\, ds$

that is, with $f_k = f(t_k)$ we have a Gauss-Markov model

$\Rightarrow f(t) = \underbrace{e^{-\lambda(t-t_0)}}_{\to A} f(t_0) + \underbrace{\int_{t_0}^{t} e^{-\lambda(t-s)} w(s)\, ds}_{\to Q}$

$$f_{k+1} \sim p(f_{k+1} \mid f_k)$$
$$y_k \sim p(y_k \mid f_k)$$

KF.

- Solvable in $O(n)$ time using Kalman filter/smoother.

$\boxed{f(t_{k+1}) = A \cdot f_k + q \qquad q \sim N(0, Q)}$

$\cancel{\frac{q}{2\lambda}(1 - e^{-2\lambda(t-t_0)})}$

$\Rightarrow p(f(t) \mid f(t_0)) = N(f(t) \mid A f(t_0), Q)$

$A = e^{-\lambda(t-t_0)}$

$Q = \int_{t_0}^{t} e^{-2\lambda(t-s)} q\, ds$

State Space Form of Linear Time-Invariant SDEs

- Consider a Nth order LTI SDE of the form

$$\frac{d^N f}{dt^N} + a_{N-1} \frac{d^{N-1} f}{dt^{N-1}} + \cdots + a_0 f = w(t).$$

- If we define $\mathbf{f} = (f, \ldots, d^{N-1} f / dt^{N-1})$, we get a state space model:

$$\frac{d\mathbf{f}}{dt} = \underbrace{\begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & -a_1 & \ldots & -a_{N-1} \end{pmatrix}}_{\mathbf{A}} \mathbf{f} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}}_{\mathbf{L}} w(t)$$

$$f(t) = \underbrace{\begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}}_{\mathbf{H}} \mathbf{f}.$$

- The vector process $\mathbf{f}(t)$ is Markovian although $f(t)$ isn't.

State Space Form of Linear Time-Invariant SDEs

- Consider a *N*th order LTI SDE of the form

$$\frac{d^N f}{dt^N} + a_{N-1} \frac{d^{N-1} f}{dt^{N-1}} + \cdots + a_0 f = w(t).$$

- If we define $\mathbf{f} = (f, \ldots, d^{N-1} f / dt^{N-1})$, we get a state space model:

$$\frac{d\mathbf{f}}{dt} = \underbrace{\begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & -a_1 & \ldots & -a_{N-1} \end{pmatrix}}_{\mathbf{A}} \mathbf{f} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}}_{\mathbf{L}} w(t)$$

$$f(t) = \underbrace{\begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}}_{\mathbf{H}} \mathbf{f}.$$

- The vector process $\mathbf{f}(t)$ is Markovian although $f(t)$ isn't.

State Space Form of Linear Time-Invariant SDEs

- Consider a Nth order LTI SDE of the form

$$\frac{d^N f}{dt^N} + a_{N-1} \frac{d^{N-1} f}{dt^{N-1}} + \cdots + a_0 f = w(t).$$

- If we define $\mathbf{f} = (f, \ldots, d^{N-1} f / dt^{N-1})$, we get a state space model:

$$\frac{d\mathbf{f}}{dt} = \underbrace{\begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & 0 & 1 \\ -a_0 & -a_1 & \ldots & -a_{N-1} \end{pmatrix}}_{\mathbf{A}} \mathbf{f} + \underbrace{\begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}}_{\mathbf{L}} w(t)$$

$$f(t) = \underbrace{\begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}}_{\mathbf{H}} \mathbf{f}.$$

- The vector process $\mathbf{f}(t)$ is Markovian although $f(t)$ isn't.

Spectra of Linear Time-Invariant SDEs

- By taking the Fourier transform of the LTI SDE, we can derive the spectral density which has the form:

$$S(\omega) = \frac{\text{(constant)}}{\text{(polynomial in } \omega^2)} \qquad \frac{q}{\omega^2 + \lambda^2}$$

- We can also do this conversion to the other direction:
  - With certain parameter values, the Matérn has the form:

$$S(\omega) \propto (\lambda^2 + \omega^2)^{-(p+1)}.$$

  - Many non-rational spectral densities can be approximated:

$$S(\omega) = \sigma^2 \sqrt{\frac{\pi}{\kappa}} \exp\left(-\frac{\omega^2}{4\kappa}\right) \approx \frac{\text{(const)}}{N!/0!(4\kappa)^N + \cdots + \omega^{2N}}$$

- For the conversion of a rational spectral density to a Markovian (state-space) model, we can use the spectral factorization.

Spectra of Linear Time-Invariant SDEs

- By taking the Fourier transform of the LTI SDE, we can derive the spectral density which has the form:

$$S(\omega) = \frac{\text{(constant)}}{\text{(polynomial in } \omega^2\text{)}}$$

- We can also do this conversion to the other direction:
  - With certain parameter values, the Matérn has the form:

$$S(\omega) \propto (\lambda^2 + \omega^2)^{-(p+1)}.$$

  - Many non-rational spectral densities can be approximated:

$$S(\omega) = \sigma^2 \sqrt{\frac{\pi}{\kappa}} \exp\left(-\frac{\omega^2}{4\kappa}\right) \approx \frac{\text{(const)}}{N!/0!(4\kappa)^N + \cdots + \omega^{2N}}$$

- For the conversion of a rational spectral density to a Markovian (state-space) model, we can use the spectral factorization.

Spectra of Linear Time-Invariant SDEs

- By taking the Fourier transform of the LTI SDE, we can derive the spectral density which has the form:

$$S(\omega) = \frac{\text{(constant)}}{\text{(polynomial in } \omega^2\text{)}}$$

- We can also do this conversion to the other direction:
  - With certain parameter values, the Matérn has the form:

$$S(\omega) \propto (\lambda^2 + \omega^2)^{-(p+1)}.$$

  - Many non-rational spectral densities can be approximated:

$$S(\omega) = \sigma^2 \sqrt{\frac{\pi}{\kappa}} \exp\left(-\frac{\omega^2}{4\kappa}\right) \approx \frac{\text{(const)}}{N!/0!(4\kappa)^N + \cdots + \omega^{2N}}$$

- For the conversion of a rational spectral density to a Markovian (state-space) model, we can use the spectral factorization.

Spectra of Linear Time-Invariant SDEs

- By taking the Fourier transform of the LTI SDE, we can derive the spectral density which has the form:

$$S(\omega) = \frac{\text{(constant)}}{\text{(polynomial in } \omega^2)}$$

- We can also do this conversion to the other direction:
  - With certain parameter values, the Matérn has the form:

$$S(\omega) \propto (\lambda^2 + \omega^2)^{-(p+1)}.$$

  - Many non-rational spectral densities can be approximated:

$$S(\omega) = \sigma^2 \sqrt{\frac{\pi}{\kappa}} \exp\left(-\frac{\omega^2}{4\kappa}\right) \approx \frac{\text{(const)}}{N!/0!(4\kappa)^N + \cdots + \omega^{2N}}$$

- For the conversion of a rational spectral density to a Markovian (state-space) model, we can use the spectral factorization.

Spectra of Linear Time-Invariant SDEs

- By taking the Fourier transform of the LTI SDE, we can derive the spectral density which has the form:

$$S(\omega) = \frac{\text{(constant)}}{\text{(polynomial in } \omega^2)}$$

- We can also do this conversion to the other direction:
  - With certain parameter values, the Matérn has the form:

$$S(\omega) \propto (\lambda^2 + \omega^2)^{-(p+1)}.$$

  - Many non-rational spectral densities can be approximated:

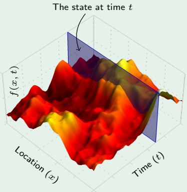
$$S(\omega) = \sigma^2 \sqrt{\frac{\pi}{\kappa}} \exp\left(-\frac{\omega^2}{4\kappa}\right) \approx \frac{\text{(const)}}{N!/0!(4\kappa)^N + \cdots + \omega^{2N}}$$

- For the conversion of a rational spectral density to a Markovian (state-space) model, we can use the spectral factorization.

State-space methods for Gaussian processes

- Approximation:

$$S(\omega) \approx \frac{b_0 + b_1\,\omega^2 + \cdots + b_M\,\omega^{2M}}{a_0 + a_1\,\omega^2 + \cdots + a_N\,\omega^{2N}}$$



- Results in a linear stochastic differential equation (SDE)

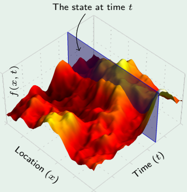
$$d\mathbf{f}(t) = \mathbf{A}\,\mathbf{f}(t)\,dt + \mathbf{L}\,d\mathbf{W}$$

- More generally stochastic evolution equations.
- $O(n)$ GP regression with Kalman filters and smoothers.
- Parallel block-sparse precision methods $\longrightarrow O(\log n)$.

State-space methods for Gaussian processes

- Approximation:

$$S(\omega) \approx \frac{b_0 + b_1\,\omega^2 + \cdots + b_M\,\omega^{2M}}{a_0 + a_1\,\omega^2 + \cdots + a_N\,\omega^{2N}}$$



- Results in a linear stochastic differential equation (SDE)

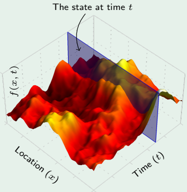
$$d\mathbf{f}(t) = \mathbf{A}\,\mathbf{f}(t)\,dt + \mathbf{L}\,d\mathbf{W}$$

- More generally stochastic evolution equations.
- $O(n)$ GP regression with Kalman filters and smoothers.
- Parallel block-sparse precision methods $\longrightarrow O(\log n)$.

State-space methods for Gaussian processes

- Approximation:

$$S(\omega) \approx \frac{b_0 + b_1\,\omega^2 + \cdots + b_M\,\omega^{2M}}{a_0 + a_1\,\omega^2 + \cdots + a_N\,\omega^{2N}}$$



- Results in a linear stochastic differential equation (SDE)

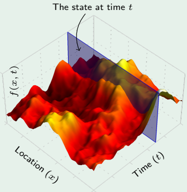
$$d\mathbf{f}(t) = \mathbf{A}\,\mathbf{f}(t)\,dt + \mathbf{L}\,d\mathbf{W}$$

- More generally stochastic evolution equations.
- $O(n)$ GP regression with Kalman filters and smoothers.
- Parallel block-sparse precision methods $\longrightarrow O(\log n)$.

State-space methods for Gaussian processes

- Approximation:

$$S(\omega) \approx \frac{b_0 + b_1\,\omega^2 + \cdots + b_M\,\omega^{2M}}{a_0 + a_1\,\omega^2 + \cdots + a_N\,\omega^{2N}}$$



- Results in a linear stochastic differential equation (SDE)

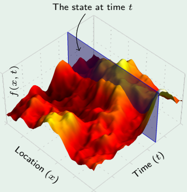
$$d\mathbf{f}(t) = \mathbf{A}\,\mathbf{f}(t)\,dt + \mathbf{L}\,d\mathbf{W}$$

- More generally stochastic evolution equations.
- $O(n)$ GP regression with Kalman filters and smoothers.
- Parallel block-sparse precision methods $\longrightarrow O(\log n)$.

State-space methods for Gaussian processes

- Approximation:

$$S(\omega) \approx \frac{b_0 + b_1\,\omega^2 + \cdots + b_M\,\omega^{2M}}{a_0 + a_1\,\omega^2 + \cdots + a_N\,\omega^{2N}}$$



- Results in a linear stochastic differential equation (SDE)

$$d\mathbf{f}(t) = \mathbf{A}\,\mathbf{f}(t)\,dt + \mathbf{L}\,d\mathbf{W}$$

- More generally stochastic evolution equations.
- $O(n)$ GP regression with Kalman filters and smoothers. NTS
- Parallel block-sparse precision methods $\longrightarrow O(\log n)$.

State-space methods – temporal example

Example (Matérn class 1d)

The Matérn class of covariance functions is

$$k(t, t') = \sigma^2 \, \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{\ell} |t - t'| \right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}}{\ell} |t - t'| \right).$$

When, e.g., $\nu = 3/2$, we have

$$\begin{aligned} d\mathbf{f}(t) &= \begin{pmatrix} 0 & 1 \\ -\lambda^2 & -2\lambda \end{pmatrix} \mathbf{f}(t) \, dt + \begin{pmatrix} 0 \\ q^{1/2} \end{pmatrix} dW(t), \\ f(t) &= \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{f}(t). \end{aligned}$$

State-space methods – spatio-temporal example

Example (2D Matérn covariance function)

- Consider a space-time Matérn covariance function

$$k(x,t;x',t') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\,\frac{\rho}{l}\right)^{\nu} K_{\nu}\left(\sqrt{2\nu}\,\frac{\rho}{l}\right).$$

  where we have $\rho = \sqrt{(t-t')^2 + (x-x')^2}$, $\nu = 1$ and $d = 2$.

- We get the following representation:

$$d\mathbf{f}(x,t) = \begin{pmatrix} 0 & 1 \\ \frac{\partial^2}{\partial x^2} - \lambda^2 & -2\sqrt{\lambda^2 - \frac{\partial^2}{\partial x^2}} \end{pmatrix} \mathbf{f}(x,t)\,dt + \begin{pmatrix} 0 \\ 1 \end{pmatrix} dW(x,t).$$

State-space methods – spatio-temporal example

Example (2D Matérn covariance function)

- Consider a space-time Matérn covariance function

$$k(x,t;x',t') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\,\frac{\rho}{l}\right)^{\nu} K_{\nu}\left(\sqrt{2\nu}\,\frac{\rho}{l}\right).$$

  where we have $\rho = \sqrt{(t-t')^2 + (x-x')^2}$, $\nu = 1$ and $d = 2$.

- We get the following representation:

$$d\mathbf{f}(x,t) = \begin{pmatrix} 0 & 1 \\ \frac{\partial^2}{\partial x^2} - \lambda^2 & -2\sqrt{\lambda^2 - \frac{\partial^2}{\partial x^2}} \end{pmatrix} \mathbf{f}(x,t)\,dt + \begin{pmatrix} 0 \\ 1 \end{pmatrix} dW(x,t).$$

Contents

Basic idea of SPDE inference on GPs [1/2]

- Consider e.g. the stochastic partial differential equation:

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} - \lambda^2 f(x,y) = w(x,y)$$

- Fourier transforming gives the spectral density:

$$S(\omega_x, \omega_y) \propto \left(\lambda^2 + \omega_x^2 + \omega_y^2\right)^{-2}.$$

- Inverse Fourier transform gives the covariance function:

$$k(x,y;x',y') = \frac{\sqrt{(x-x')^2 + (y-y')^2}}{2\lambda} K_1(\lambda \sqrt{(x-x')^2 + (y-y')^2})$$

- But this is just the Matérn covariance function.

Basic idea of SPDE inference on GPs [1/2]

- Consider e.g. the stochastic partial differential equation:

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} - \lambda^2 f(x,y) = w(x,y)$$

- Fourier transforming gives the spectral density:

$$S(\omega_x, \omega_y) \propto \left(\lambda^2 + \omega_x^2 + \omega_y^2\right)^{-2}.$$

- Inverse Fourier transform gives the covariance function:

$$k(x,y;x',y') = \frac{\sqrt{(x-x')^2 + (y-y')^2}}{2\lambda} K_1(\lambda \sqrt{(x-x')^2 + (y-y')^2})$$

- But this is just the Matérn covariance function.

Basic idea of SPDE inference on GPs [1/2]

- Consider e.g. the stochastic partial differential equation:

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} - \lambda^2 f(x,y) = w(x,y)$$

- Fourier transforming gives the spectral density:

$$S(\omega_x, \omega_y) \propto \left(\lambda^2 + \omega_x^2 + \omega_y^2\right)^{-2}.$$

- Inverse Fourier transform gives the covariance function:

$$k(x,y;x',y') = \frac{\sqrt{(x-x')^2 + (y-y')^2}}{2\lambda} K_1(\lambda \sqrt{(x-x')^2 + (y-y')^2})$$

- But this is just the Matérn covariance function.

Basic idea of SPDE inference on GPs [1/2]

- Consider e.g. the stochastic partial differential equation:

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} - \lambda^2 f(x,y) = w(x,y)$$

- Fourier transforming gives the spectral density:

$$S(\omega_x,\omega_y) \propto \left(\lambda^2 + \omega_x^2 + \omega_y^2\right)^{-2}.$$

- Inverse Fourier transform gives the covariance function:

$$k(x,y;x',y') = \frac{\sqrt{(x-x')^2+(y-y')^2}}{2\lambda} K_1(\lambda\sqrt{(x-x')^2+(y-y')^2})$$

- But this is just the Matérn covariance function.

Basic idea of SPDE inference on GPs [2/2]

- More generally, SPDE for some linear operator $\mathcal{L}$:

$$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$

- Now f is a GP with precision and covariance operators:

$$\mathcal{K}^{-1} = \mathcal{L}^* \, \mathcal{L}$$
$$\mathcal{K} = (\mathcal{L}^* \, \mathcal{L})^{-1}$$

- Idea: approximate $\mathcal{L}$ or $\mathcal{L}^{-1}$ using PDE/ODE methods:
  1. Finite-differences/FEM methods lead to sparse precision approximations.
  2. Fourier/basis-function methods lead to reduced rank covariance approximations.
  3. Spectral factorization leads to state-space (Kalman) methods which are time-recursive (or sparse in precision).

Basic idea of SPDE inference on GPs [2/2]

- More generally, SPDE for some linear operator $\mathcal{L}$:

$$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$

- Now f is a GP with precision and covariance operators:

$$\mathcal{K}^{-1} = \mathcal{L}^* \, \mathcal{L}$$
$$\mathcal{K} = (\mathcal{L}^* \, \mathcal{L})^{-1}$$

- Idea: approximate $\mathcal{L}$ or $\mathcal{L}^{-1}$ using PDE/ODE methods:
  1. Finite-differences/FEM methods lead to sparse precision approximations.
  2. Fourier/basis-function methods lead to reduced rank covariance approximations.
  3. Spectral factorization leads to state-space (Kalman) methods which are time-recursive (or sparse in precision).

Basic idea of SPDE inference on GPs [2/2]

- More generally, SPDE for some linear operator $\mathcal{L}$:

$$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$

- Now f is a GP with precision and covariance operators:

$$\mathcal{K}^{-1} = \mathcal{L}^*\, \mathcal{L}$$
$$\mathcal{K} = (\mathcal{L}^*\, \mathcal{L})^{-1}$$

- Idea: approximate $\mathcal{L}$ or $\mathcal{L}^{-1}$ using PDE/ODE methods:
  1. Finite-differences/FEM methods lead to sparse precision approximations.
  2. Fourier/basis-function methods lead to reduced rank covariance approximations.
  3. Spectral factorization leads to state-space (Kalman) methods which are time-recursive (or sparse in precision).

Basic idea of SPDE inference on GPs [2/2]

- More generally, SPDE for some linear operator $\mathcal{L}$:

$$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$

- Now f is a GP with precision and covariance operators:

$$\mathcal{K}^{-1} = \mathcal{L}^* \,\mathcal{L}$$
$$\mathcal{K} = (\mathcal{L}^* \,\mathcal{L})^{-1}$$

- Idea: approximate $\mathcal{L}$ or $\mathcal{L}^{-1}$ using PDE/ODE methods:
  1. Finite-differences/FEM methods lead to sparse precision approximations.
  2. Fourier/basis-function methods lead to reduced rank covariance approximations.
  3. Spectral factorization leads to state-space (Kalman) methods which are time-recursive (or sparse in precision).

Basic idea of SPDE inference on GPs [2/2]

- More generally, SPDE for some linear operator $\mathcal{L}$:

$$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$

- Now f is a GP with precision and covariance operators:

$$\mathcal{K}^{-1} = \mathcal{L}^* \, \mathcal{L}$$
$$\mathcal{K} = (\mathcal{L}^* \, \mathcal{L})^{-1}$$

- Idea: approximate $\mathcal{L}$ or $\mathcal{L}^{-1}$ using PDE/ODE methods:
  1. Finite-differences/FEM methods lead to sparse precision approximations.
  2. Fourier/basis-function methods lead to reduced rank covariance approximations.
  3. Spectral factorization leads to state-space (Kalman) methods which are time-recursive (or sparse in precision).

Basic idea of SPDE inference on GPs [2/2]

- More generally, SPDE for some linear operator $\mathcal{L}$:

$$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$

- Now f is a GP with precision and covariance operators:

$$\mathcal{K}^{-1} = \mathcal{L}^*\, \mathcal{L}$$
$$\mathcal{K} = (\mathcal{L}^*\, \mathcal{L})^{-1}$$

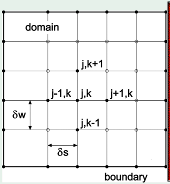
- Idea: approximate $\mathcal{L}$ or $\mathcal{L}^{-1}$ using PDE/ODE methods:
  1. Finite-differences/FEM methods lead to sparse precision approximations.
  2. Fourier/basis-function methods lead to reduced rank covariance approximations.
  3. Spectral factorization leads to state-space (Kalman) methods which are time-recursive (or sparse in precision).

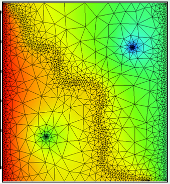
Finite-differences/FEM – sparse precision

- Basic idea:

$$\frac{\partial f(x)}{\partial x} \approx \frac{f(x+h) - f(x)}{h}$$

$$\frac{\partial^2 f(x)}{\partial x^2} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$





- We get an SPDE approximation $\mathcal{L} \approx \mathbf{L}$, where $\mathbf{L}$ is sparse
- The precision operator approximation is then sparse:

$$\mathcal{K}^{-1} \approx \mathbf{L}^{\mathsf{T}} \mathbf{L} = \text{sparse}$$

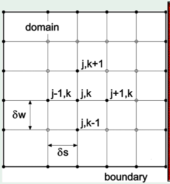
- $\mathcal{L}$ need to be approximated as integro-differential operator.
- Requires formation of a grid, but parallelizes well.

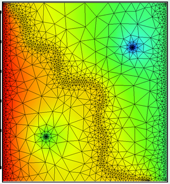
Finite-differences/FEM – sparse precision

- Basic idea:

$$\frac{\partial f(x)}{\partial x} \approx \frac{f(x+h) - f(x)}{h}$$

$$\frac{\partial^2 f(x)}{\partial x^2} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$





- We get an SPDE approximation $\mathcal{L} \approx \mathbf{L}$, where $\mathbf{L}$ is sparse
- The precision operator approximation is then sparse:

$$\mathcal{K}^{-1} \approx \mathbf{L}^{\mathsf{T}} \mathbf{L} = \text{sparse}$$

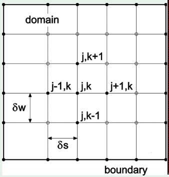
- $\mathcal{L}$ need to be approximated as integro-differential operator.
- Requires formation of a grid, but parallelizes well.

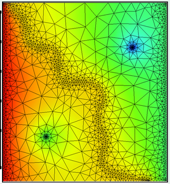
Finite-differences/FEM – sparse precision

- Basic idea:

$$\frac{\partial f(x)}{\partial x} \approx \frac{f(x+h) - f(x)}{h}$$

$$\frac{\partial^2 f(x)}{\partial x^2} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$





- We get an SPDE approximation $\mathcal{L} \approx \mathbf{L}$, where $\mathbf{L}$ is sparse
- The precision operator approximation is then sparse:

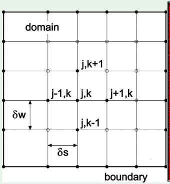
$$\mathcal{K}^{-1} \approx \mathbf{L}^\mathsf{T} \mathbf{L} = \text{sparse}$$

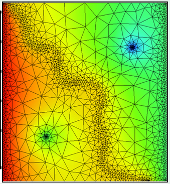
- $\mathcal{L}$ need to be approximated as integro-differential operator.
- Requires formation of a grid, but parallelizes well.

Finite-differences/FEM – sparse precision

- Basic idea:

$$\frac{\partial f(x)}{\partial x} \approx \frac{f(x+h) - f(x)}{h}$$
$$\frac{\partial^2 f(x)}{\partial x^2} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$





- We get an SPDE approximation $\mathcal{L} \approx \mathbf{L}$, where $\mathbf{L}$ is sparse
- The precision operator approximation is then sparse:

$$\mathcal{K}^{-1} \approx \mathbf{L}^{\mathsf{T}} \mathbf{L} = \text{sparse}$$

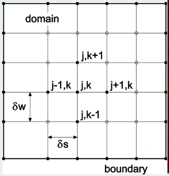
- $\mathcal{L}$ need to be approximated as integro-differential operator.
- Requires formation of a grid, but parallelizes well.

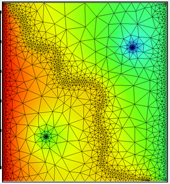
Finite-differences/FEM – sparse precision

- Basic idea:

$$\frac{\partial f(x)}{\partial x} \approx \frac{f(x+h) - f(x)}{h}$$

$$\frac{\partial^2 f(x)}{\partial x^2} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$





- We get an SPDE approximation $\mathcal{L} \approx \mathbf{L}$, where $\mathbf{L}$ is sparse
- The precision operator approximation is then sparse:

$$\mathcal{K}^{-1} \approx \mathbf{L}^\mathsf{T}\,\mathbf{L} = \text{sparse}$$

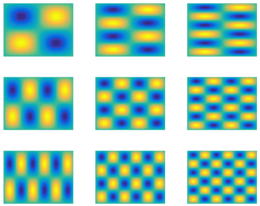
- $\mathcal{L}$ need to be approximated as integro-differential operator.
- Requires formation of a grid, but parallelizes well.

Classical and random Fourier methods – reduced rank approximations and FFT

- Approximation:

$$f(\mathbf{x}) \approx \sum_{\mathbf{k} \in \mathbb{N}^d} c_{\mathbf{k}} \exp\left(2\pi\, \mathrm{i}\, \mathbf{k}^\mathsf{T} \mathbf{x}\right)$$

$$c_{\mathbf{k}} \sim \text{Gaussian}$$



- We use less coefficients $c_{\mathbf{k}}$ than the number of data points.
- Leads to reduced-rank covariance approximations

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{|\mathbf{k}| \leq N} \sigma_{\mathbf{k}}^2 \exp\left(2\pi\, \mathrm{i}\, \mathbf{k}^\mathsf{T} \mathbf{x}\right) \exp\left(2\pi\, \mathrm{i}\, \mathbf{k}^\mathsf{T} \mathbf{x}'\right)^*$$

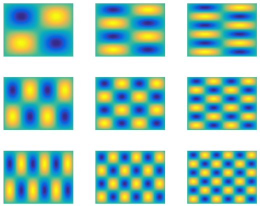
- Truncated series, random frequencies, FFT, ...

Classical and random Fourier methods – reduced rank approximations and FFT

- Approximation:

$$f(\mathbf{x}) \approx \sum_{\mathbf{k} \in \mathbb{N}^d} c_{\mathbf{k}} \exp\left(2\pi\, \mathrm{i}\, \mathbf{k}^\mathsf{T} \mathbf{x}\right)$$

$$c_{\mathbf{k}} \sim \text{Gaussian}$$



- We use less coefficients $c_{\mathbf{k}}$ than the number of data points.
- Leads to reduced-rank covariance approximations

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{|\mathbf{k}| \le N} \sigma_{\mathbf{k}}^2 \exp\left(2\pi\, \mathrm{i}\, \mathbf{k}^\mathsf{T} \mathbf{x}\right) \exp\left(2\pi\, \mathrm{i}\, \mathbf{k}^\mathsf{T} \mathbf{x}'\right)^*$$

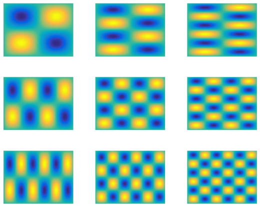
- Truncated series, random frequencies, FFT, ...

Classical and random Fourier methods – reduced rank approximations and FFT

- Approximation:

$$f(\mathbf{x}) \approx \sum_{\mathbf{k} \in \mathbb{N}^d} c_{\mathbf{k}} \exp\left(2\pi \, \mathrm{i} \, \mathbf{k}^{\mathsf{T}} \mathbf{x}\right)$$

$$c_{\mathbf{k}} \sim \text{Gaussian}$$



- We use less coefficients $c_{\mathbf{k}}$ than the number of data points.
- Leads to reduced-rank covariance approximations

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{|\mathbf{k}| \le N} \sigma_{\mathbf{k}}^2 \exp\left(2\pi \, \mathrm{i} \, \mathbf{k}^{\mathsf{T}} \mathbf{x}\right) \exp\left(2\pi \, \mathrm{i} \, \mathbf{k}^{\mathsf{T}} \mathbf{x}'\right)^*$$

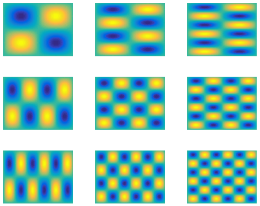
- Truncated series, random frequencies, FFT, ...

Classical and random Fourier methods – reduced rank approximations and FFT

- Approximation:

$$f(\mathbf{x}) \approx \sum_{\mathbf{k} \in \mathbb{N}^d} c_{\mathbf{k}} \exp\left(2\pi \, i \, \mathbf{k}^\mathsf{T} \, \mathbf{x}\right)$$

$$c_{\mathbf{k}} \sim \text{Gaussian}$$



- We use less coefficients $c_{\mathbf{k}}$ than the number of data points.
- Leads to reduced-rank covariance approximations

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{|\mathbf{k}| \le N} \sigma_{\mathbf{k}}^2 \exp\left(2\pi \, i \, \mathbf{k}^\mathsf{T} \, \mathbf{x}\right) \exp\left(2\pi \, i \, \mathbf{k}^\mathsf{T} \, \mathbf{x}'\right)^*$$

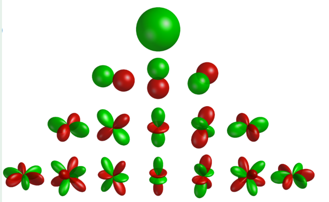
- Truncated series, random frequencies, FFT, ...

Hilbert-space/Galerkin methods – reduced rank approximations

- Approximation:

$$f(\mathbf{x}) \approx \sum_i c_i \, \phi_i(\mathbf{x})$$

$$\langle \phi_i, \phi_j \rangle_H \approx \delta_{ij}, \text{ e.g. } \nabla^2 \phi_i = -\lambda_i \, \phi_i$$



- Again, use less coefficients than the number of data points.
- Reduced-rank covariance approximations such as

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{i=1}^{N} \sigma_i^2 \, \phi_i(\mathbf{x}) \, \phi_i(\mathbf{x}').$$

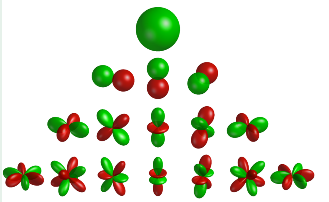
- Wavelets, Galerkin, finite elements, …

Hilbert-space/Galerkin methods – reduced rank approximations

- Approximation:

$$f(\mathbf{x}) \approx \sum_i c_i \, \phi_i(\mathbf{x})$$

$$\langle \phi_i, \phi_j \rangle_H \approx \delta_{ij}, \text{ e.g. } \nabla^2 \phi_i = -\lambda_i \, \phi_i$$



- Again, use less coefficients than the number of data points.
- Reduced-rank covariance approximations such as

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{i=1}^{N} \sigma_i^2 \, \phi_i(\mathbf{x}) \, \phi_i(\mathbf{x}').$$

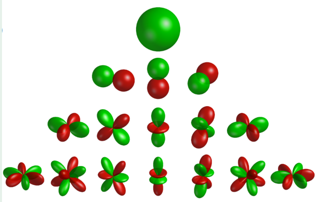
- Wavelets, Galerkin, finite elements, …

Hilbert-space/Galerkin methods – reduced rank approximations

- Approximation:

$$f(\mathbf{x}) \approx \sum_i c_i \, \phi_i(\mathbf{x})$$

$$\langle \phi_i, \phi_j \rangle_H \approx \delta_{ij}, \text{ e.g. } \nabla^2 \phi_i = -\lambda_i \, \phi_i$$



- Again, use less coefficients than the number of data points.
- Reduced-rank covariance approximations such as

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{i=1}^{N} \sigma_i^2 \, \phi_i(\mathbf{x}) \, \phi_i(\mathbf{x}').$$

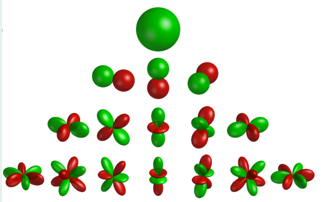
- Wavelets, Galerkin, finite elements, ...

Hilbert-space/Galerkin methods – reduced rank approximations

- Approximation:

$$f(\mathbf{x}) \approx \sum_i c_i \, \phi_i(\mathbf{x})$$

$$\langle \phi_i, \phi_j \rangle_H \approx \delta_{ij}, \text{ e.g. } \nabla^2 \phi_i = -\lambda_i \, \phi_i$$



- Again, use less coefficients than the number of data points.
- Reduced-rank covariance approximations such as

$$k(\mathbf{x}, \mathbf{x}') \approx \sum_{i=1}^{N} \sigma_i^2 \, \phi_i(\mathbf{x}) \, \phi_i(\mathbf{x}').$$

- Wavelets, Galerkin, finite elements, ...

Contents

Back to SPDE representations of GPs

GP model $\mathbf{x} \in \mathbb{R}^d$, $t \in \mathbb{R}$	Equivalent S(P)DE model
Spatial $k(\mathbf{x}, \mathbf{x}')$	SPDE model ($\mathcal{L}$ is an operator) $$\mathcal{L}\, f(\mathbf{x}) = w(\mathbf{x})$$
Temporal $k(t, t')$	State-space/SDE model $$\frac{d\mathbf{f}(t)}{dt} = \mathbf{A}\,\mathbf{f}(t) + \mathbf{L}\,w(t)$$
Spatio-temporal $k(\mathbf{x}, t; \mathbf{x}', t')$	Stochastic evolution equation $$\frac{\partial}{\partial t}\mathbf{f}(\mathbf{x}, t) = \mathcal{A}_x\,\mathbf{f}(\mathbf{x}, t) + \mathbf{L}\,w(\mathbf{x}, t)$$

What then?

- Exchange and map approximations between the fields:
  - Inducing points $\leftrightarrow$ point-collocation; spectral methods $\leftrightarrow$ Galerkin methods; finite-differences $\leftrightarrow$ GMRFs;
- Non-Gaussian processes: Student's-t processes, non-linear Itô processes, jump processes, hybrid point/Gaussian processes.
- Hierarchical (deep) SPDE models: we stack SPDEs on top of each other – the SPDE just becomes non-linear.
- Combined first-principles and nonparametric models – latent force models (LFM), also non-linear and non-Gaussian LFMs.
- Inverse problems – operators in the measurement model.

What then?

- Exchange and map approximations between the fields:
  - Inducing points $\leftrightarrow$ point-collocation; spectral methods $\leftrightarrow$ Galerkin methods; finite-differences $\leftrightarrow$ GMRFs;
- Non-Gaussian processes: Student's-t processes, non-linear Itô processes, jump processes, hybrid point/Gaussian processes.
- Hierarchical (deep) SPDE models: we stack SPDEs on top of each other – the SPDE just becomes non-linear.
- Combined first-principles and nonparametric models – latent force models (LFM), also non-linear and non-Gaussian LFMs.
- Inverse problems – operators in the measurement model.

What then?

- Exchange and map approximations between the fields:
  - Inducing points $\leftrightarrow$ point-collocation; spectral methods $\leftrightarrow$ Galerkin methods; finite-differences $\leftrightarrow$ GMRFs;
- Non-Gaussian processes: Student's-t processes, non-linear Itô processes, jump processes, hybrid point/Gaussian processes.
- Hierarchical (deep) SPDE models: we stack SPDEs on top of each other – the SPDE just becomes non-linear.
- Combined first-principles and nonparametric models – latent force models (LFM), also non-linear and non-Gaussian LFMs.
- Inverse problems – operators in the measurement model.

What then?

- Exchange and map approximations between the fields:
  - Inducing points $\leftrightarrow$ point-collocation; spectral methods $\leftrightarrow$ Galerkin methods; finite-differences $\leftrightarrow$ GMRFs;
- Non-Gaussian processes: Student's-t processes, non-linear Itô processes, jump processes, hybrid point/Gaussian processes.
- Hierarchical (deep) SPDE models: we stack SPDEs on top of each other – the SPDE just becomes non-linear.
- Combined first-principles and nonparametric models – latent force models (LFM), also non-linear and non-Gaussian LFMs.
- Inverse problems – operators in the measurement model.

What then?

- Exchange and map approximations between the fields:
  - Inducing points $\leftrightarrow$ point-collocation; spectral methods $\leftrightarrow$ Galerkin methods; finite-differences $\leftrightarrow$ GMRFs;
- Non-Gaussian processes: Student's-t processes, non-linear Itô processes, jump processes, hybrid point/Gaussian processes.
- Hierarchical (deep) SPDE models: we stack SPDEs on top of each other – the SPDE just becomes non-linear.
- Combined first-principles and nonparametric models – latent force models (LFM), also non-linear and non-Gaussian LFMs.
- Inverse problems – operators in the measurement model.

What then?

- Exchange and map approximations between the fields:
  - Inducing points $\leftrightarrow$ point-collocation; spectral methods $\leftrightarrow$ Galerkin methods; finite-differences $\leftrightarrow$ GMRFs;
- Non-Gaussian processes: Student's-t processes, non-linear Itô processes, jump processes, hybrid point/Gaussian processes.
- Hierarchical (deep) SPDE models: we stack SPDEs on top of each other – the SPDE just becomes non-linear.
- Combined first-principles and nonparametric models – latent force models (LFM), also non-linear and non-Gaussian LFMs.
- Inverse problems – operators in the measurement model.

Summary

- Gaussian processes (GPs) are nice, but the computational scaling is bad.
- GPs have representations as solutions to SPDEs.
- In temporal models we can use Kalman/Bayesian filters and smoothers.
- SPDE methods can be used to speed up GP inference.
- New paths towards non-linear GP models.
- Work nicely in latent force models.

Summary

- Gaussian processes (GPs) are nice, but the computational scaling is bad.
- GPs have representations as solutions to SPDEs.
- In temporal models we can use Kalman/Bayesian filters and smoothers.
- SPDE methods can be used to speed up GP inference.
- New paths towards non-linear GP models.
- Work nicely in latent force models.

Summary

- Gaussian processes (GPs) are nice, but the computational scaling is bad.
- GPs have representations as solutions to SPDEs.
- In temporal models we can use Kalman/Bayesian filters and smoothers.
- SPDE methods can be used to speed up GP inference.
- New paths towards non-linear GP models.
- Work nicely in latent force models.

Summary

- Gaussian processes (GPs) are nice, but the computational scaling is bad.
- GPs have representations as solutions to SPDEs.
- In temporal models we can use Kalman/Bayesian filters and smoothers.
- SPDE methods can be used to speed up GP inference.
- New paths towards non-linear GP models.
- Work nicely in latent force models.

Summary

- Gaussian processes (GPs) are nice, but the computational scaling is bad.
- GPs have representations as solutions to SPDEs.
- In temporal models we can use Kalman/Bayesian filters and smoothers.
- SPDE methods can be used to speed up GP inference.
- New paths towards non-linear GP models.
- Work nicely in latent force models.

Summary

- Gaussian processes (GPs) are nice, but the computational scaling is bad.
- GPs have representations as solutions to SPDEs.
- In temporal models we can use Kalman/Bayesian filters and smoothers.
- SPDE methods can be used to speed up GP inference.
- New paths towards non-linear GP models.
- Work nicely in latent force models.

Useful references

- **N. Wiener (1950).** *Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications*. John Wiley & Sons, Inc.
- **R. L. Stratonovich (1963).** *Topics in the Theory of Random Noise.* Gordon and Breach.
- **J. Hartikainen and S. Särkkä (2010).** *Kalman Filtering and Smoothing Solutions to Temporal Gaussian Process Regression Models.* Proc. MLSP.
- **S. Särkkä, A. Solin, and J. Hartikainen (2013).** *Spatio-Temporal Learning via Infinite-Dimensional Bayesian Filtering and Smoothing*. IEEE Sig.Proc.Mag., 30(5):51–61.
- **S. Särkkä (2013).** *Bayesian Filtering and Smoothing*. Cambridge University Press.
- **S. Särkkä and A. Solin (2017, to appear)** *Applied Stochastic Differential Equations*. Cambridge University Press.