Adaptive deep learning PDE surrogates to accelerate Bayesian inference with guaranteed accuracy

with an application in turbo-machinery

Teo Deveney (joint work with Tony Shardlow and Eike Mueller)
University of Bath

Motivation

We want to perform Bayesian inference for models of the form

$$\hat{z}_i = f(\hat{\boldsymbol{x}}_i; \boldsymbol{\alpha}) + \epsilon_i$$

for data indexed by $i = 1, \ldots, M$, where $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$, and with prior $p(\boldsymbol{\theta}) = p(\boldsymbol{\alpha}, \sigma)$ chosen as appropriate.

This model gives rise to a normal likelihood function

$$p(\hat{\boldsymbol{z}} \,|\, \boldsymbol{\theta}, \hat{\boldsymbol{x}}) = \frac{1}{\left(2\pi\sigma^2\right)^{M/2}} \exp\left(-\frac{1}{2\sigma^2} \sum_{i=1}^{M} (\hat{z}_i - f(\hat{\boldsymbol{x}}_i; \boldsymbol{\alpha}))^2 \right)$$

The mean $f(\hat{\boldsymbol{x}}; \boldsymbol{\alpha})$ is assumed to depend on the solution $u(\hat{\boldsymbol{x}}; \boldsymbol{\alpha})$ of a physical model (e.g. a PDE), typically non-trivial to evaluate and not easily differentiable wrt $\boldsymbol{\alpha}$.

Classical methods

Classical methods to approximate the posterior distribution

$$p(\boldsymbol{\theta} \,|\, \hat{\boldsymbol{x}}, \hat{\boldsymbol{z}}) \propto p(\hat{\boldsymbol{z}} \,|\, \boldsymbol{\theta}, \hat{\boldsymbol{x}}) p(\boldsymbol{\theta})$$

typically follow one of two approaches:

Markov chain Monte Carlo

- Empirically sample from $p(\boldsymbol{\theta} \,|\, \hat{\boldsymbol{x}}, \hat{\boldsymbol{z}})$
- Simple schemes require many evaluations of $u(\hat{\boldsymbol{x}}; \boldsymbol{\alpha})$
- More sophisticated schemes also require $\nabla_{\alpha} u(\hat{\boldsymbol{x}}; \boldsymbol{\alpha})$
- Asymptotically exact (weak convergence as sample size $\rightarrow \infty$)

Reduced order approximation

- Represent $p(\boldsymbol{\theta} \,|\, \hat{\boldsymbol{x}}, \hat{\boldsymbol{z}})$ by a parameterised distribution $q_{\lambda}(\boldsymbol{\theta})$, forming a reduced class Q or representable distributions
- Optimise an objective $L(\boldsymbol{\lambda})$ such that $q_{\lambda}(\boldsymbol{\theta}) \approx p(\boldsymbol{\theta} \,|\, \hat{\boldsymbol{x}}, \hat{\boldsymbol{z}})$ (ELBO,MAP)
- Typically requires fewer evaluations of $u(\hat{\boldsymbol{x}}; \boldsymbol{\alpha})$ than MCMC but requires $\nabla_{\alpha} u(\hat{\boldsymbol{x}}; \boldsymbol{\alpha})$
- Does not converge to $p(\boldsymbol{\theta} \,|\, \hat{\boldsymbol{x}}, \hat{\boldsymbol{z}})$ (converges to optimal element of Q)

Deep surrogate methodology

Alleviating the computational cost of working with $u(\boldsymbol{x};\boldsymbol{\alpha})$

Assume $u(\boldsymbol{x};\boldsymbol{\alpha})$ solves some PDE parameterised by $\boldsymbol{\alpha}$

$$\begin{aligned}\mathscr{L}(u,\boldsymbol{x};\boldsymbol{\alpha}) &= 0, \qquad && \boldsymbol{x}\in\Omega, \boldsymbol{\alpha}\in A\\ \mathscr{B}(u,\boldsymbol{x};\boldsymbol{\alpha}) &= 0, && \boldsymbol{x}\in\partial\Omega, \boldsymbol{\alpha}\in A\,.\end{aligned}$$

Basic premise: Approximate the parametric solution map $u(\boldsymbol{x};\boldsymbol{\alpha}) : \Omega\times A\to\mathbb{R}$ with a neural network $\hat{u}(\boldsymbol{x};\boldsymbol{\alpha})$ by using gradient descent to minimise the loss

$$Loss(\hat{u}) = \|\mathscr{L}(\hat{u},\boldsymbol{x};\boldsymbol{\alpha})\|^2_{L_2(\Omega\times A,\pi^{\Omega}\otimes\pi^{A})} + \|\mathscr{B}(\hat{u},\boldsymbol{x};\boldsymbol{\alpha})\|^2_{L_2(\partial\Omega\times A,\pi^{\partial\Omega}\otimes\pi^{A})}$$

Once trained using appropriate measures π^{Ω}, $\pi^{\partial\Omega}$, π^{A} the neural network $\hat{u}(\boldsymbol{x};\boldsymbol{\alpha})$ is a cheap, differentiable function approximating the parametric PDE solution.

Deep surrogate methodology

Practical considerations

The loss function

$$Loss(\hat{u}) = \|\mathscr{L}(\hat{u}, \boldsymbol{x}; \boldsymbol{\alpha})\|^2_{L_2(\Omega \times A, \pi^{\Omega} \otimes \pi^{A})} + \|\mathscr{B}(\hat{u}, \boldsymbol{x}; \boldsymbol{\alpha})\|^2_{L_2(\partial\Omega \times A, \pi^{\partial\Omega} \otimes \pi^{A})}$$

is intractable. In practice draw randomised collocation points from π^{Ω}, $\pi^{\partial\Omega}$, π^{A} and minimise a Monte Carlo approximation

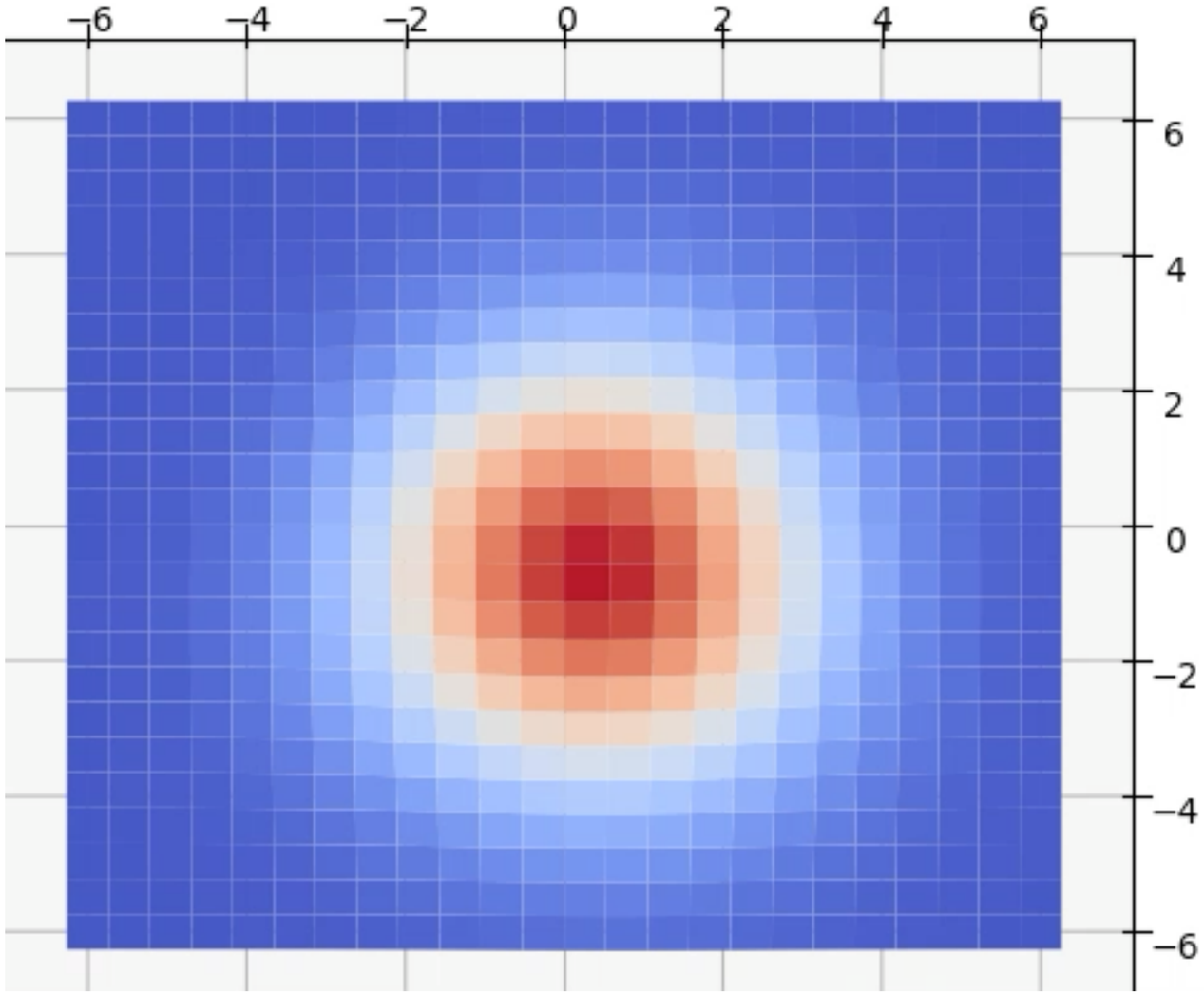
$$Loss(\hat{u}) = \sum_{n=1}^{N} \mathscr{L}(\hat{u}, \boldsymbol{x}_n^{\Omega}; \boldsymbol{\alpha}_n)^2 + \sum_{n=1}^{N} \mathscr{B}(\hat{u}, \boldsymbol{x}_n^{\partial\Omega}; \boldsymbol{\alpha}_n)^2$$

These points are re-sampled after each gradient descent iteration (resulting in SGD).

To solve Bayesian problems the trained surrogate is then used to replace a numerical solver in order to accelerate MCMC sampling.

Example

Parametric advection diffusion equation



An application in turbo machinery

We seek to understand the evolution of heat fluxes $\tilde{q}(t, r)$ within compressor cavities through the model

$$\hat{u}_i = u(t_i, r_i; \boldsymbol{\alpha}) + \epsilon_i$$

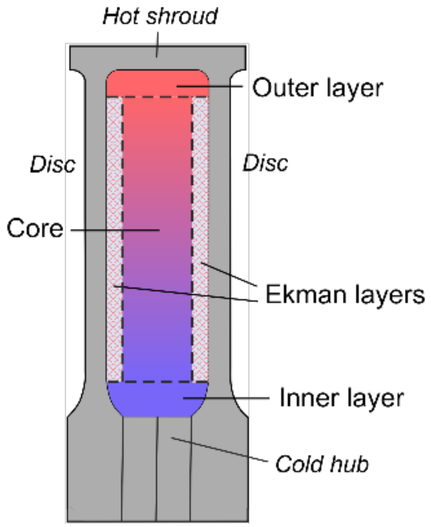
We have temperature measurements $\hat{u}_1, \ldots, \hat{u}_M$, at times $\hat{t}_1, \ldots, \hat{t}_M$, and locations $\hat{r}_1, \ldots, \hat{r}_M$, and the dynamics

$$\frac{\partial u(t,r)}{\partial t} = \frac{\partial^2 u(t,r)}{\partial r^2} + \frac{1}{r}\frac{\partial u(t,r)}{\partial r} - \tilde{q}(t,r)$$

$$u(0,r) = u_0(r)$$

$$u(t,a) = u_a(t)$$

$$u(t,b) = u_b(t)$$



This problem is ill-conditioned (without regularisation small changes in temperature indicate large changes in flux).

Key requirements

This method may influence future aircraft engine design so our approach must:

- **Allow interpretable priors**
- **Be accurate and efficient to train**
- **Have theoretical guarantees**

Interpretable priors

What happens with a poor prior on $\tilde{q}(t, r)$

Using fixed basis functions $\phi_i(t, r)$, represent $\tilde{q}(t, r)$ by the expansion

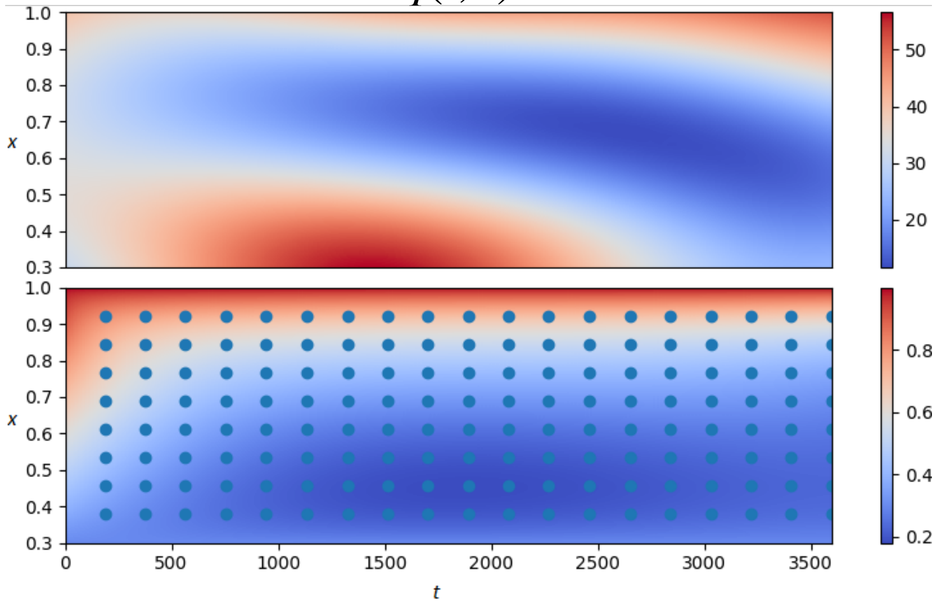
$$q(t, r) = \sum_{i=1}^{D} \alpha_i \phi_i(t, r) \, .$$

We choose the Chebyshev polynomials $\phi_i(t, r) = T_n(t) T_m(r)$ of degree $n + m \leq 10$, so $D = 66$.

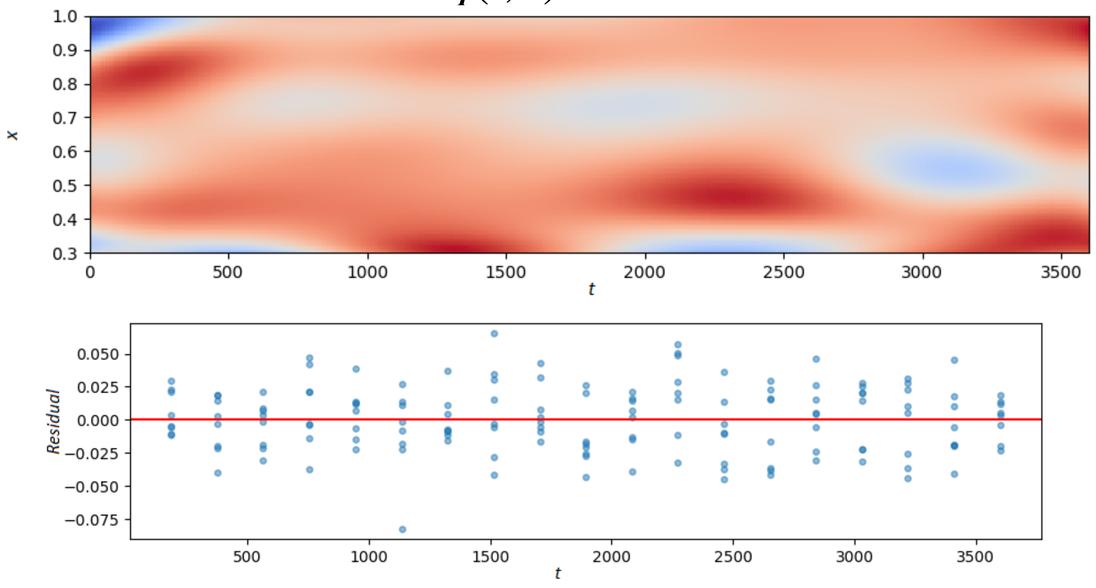
The problem is then to infer the coefficients $\boldsymbol{\alpha}$ using the data. This includes choosing an appropriate prior.

A poor prior(e.g. $\boldsymbol{\alpha} \sim$ Unif(A)), leads to physically unreasonable inferences

True $\tilde{q}(t, r)$



Inferred $q(t, r)$



Interpretable priors

Approximating a Gaussian process prior over $\tilde{q}(t, r)$

Using fixed basis functions $\phi_i(t, r)$, represent $\tilde{q}(t, r)$ by the truncated basis expansion

$$q(t, r) = \sum_{i=1}^{D} \alpha_i \phi_i(t, r).$$

Gaussian processes are a natural choice of prior, as their behaviour can be controlled through the specification of their mean and covariance functions.

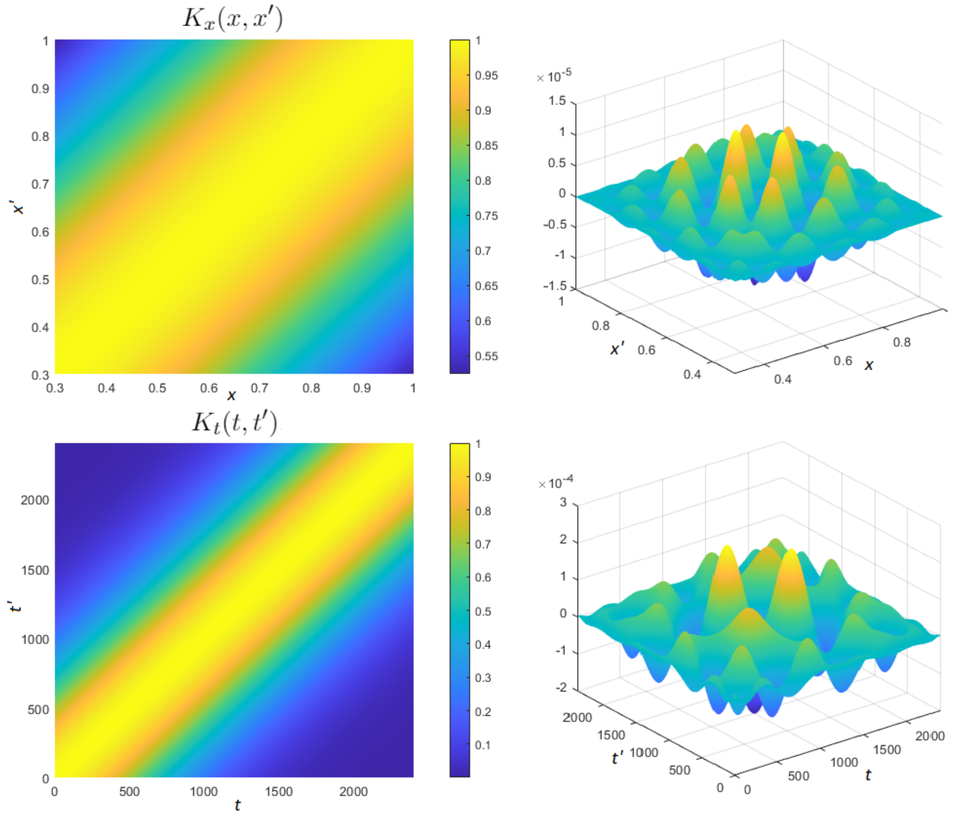
Suppose we want the prior:

$$\tilde{q}(t, r) \sim \mathcal{GP}(\mu(t, r), K([t, r], [t', r']))$$

We can approximate this with $q(t, r)$ if the coefficients are $\boldsymbol{\alpha} \sim MVN(\boldsymbol{m}, \Sigma)$, with:

$$\sum_{i=1}^{D} m_i \phi_i(t, x) \approx \mu(t, r)$$

$$\sum_{i,j=1}^{D} \phi_i(t, x) \Sigma_{i,j} \phi_j(t', x') \approx K([t, x], [t', x'])$$



Interpretable priors

Approximating a Gaussian process prior over $\tilde{q}(t, r)$

Using fixed basis functions $\phi_i(t, r)$, represent $\tilde{q}(t, r)$ by the truncated basis expansion

$$q(t, r) = \sum_{i=1}^{D} \alpha_i \phi_i(t, r).$$

Gaussian processes are a natural choice of prior, as their behaviour can be controlled through the specification of their mean and covariance functions.

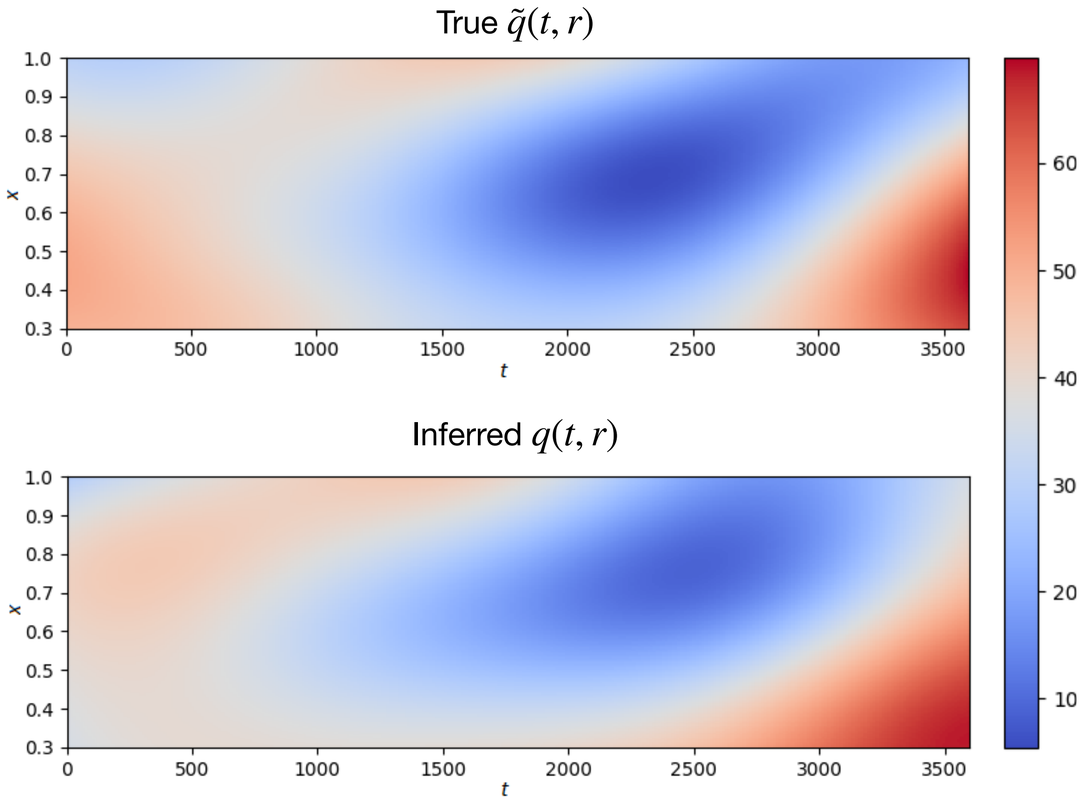
Suppose we want the prior:

$$\tilde{q}(t, r) \sim \mathcal{GP}(\mu(t, r), K([t, r], [t', r']))$$

We can approximate this with $q(t, r)$ if the coefficients are $\boldsymbol{\alpha} \sim MVN(\boldsymbol{m}, \Sigma)$, with:

$$\sum_{i=1}^{D} m_i \phi_i(t, x) \approx \mu(t, r)$$

$$\sum_{i,j=1}^{D} \phi_i(t, x) \Sigma_{i,j} \phi_j(t', x') \approx K([t, x], [t', x'])$$



Accuracy and efficiency

Tailoring the training procedure for Bayesian inversion

Assuming estimates of $\boldsymbol{\alpha}$ are not known a-priori, the standard approach to training a surrogate is to choose π^A (the training measure over the parameters) as something general e.g. $\pi^A \sim \text{Unif}(\bar{A})$.

During inference with MCMC we produce an empirical approximation to the posterior

$$\hat{p}(\boldsymbol{\theta} \,|\, z, \boldsymbol{x}) = \frac{1}{S} \sum_{n=1}^{S} \delta_{\boldsymbol{\theta}_n}(\boldsymbol{\theta})$$

$u(\boldsymbol{x}; \boldsymbol{\alpha})$ is only evaluated close to the typical set of $\hat{p}(\boldsymbol{\theta} \,|\, z, \boldsymbol{x}) \approx p(\boldsymbol{\theta} \,|\, z, \boldsymbol{x})$ (recall $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \sigma)$).

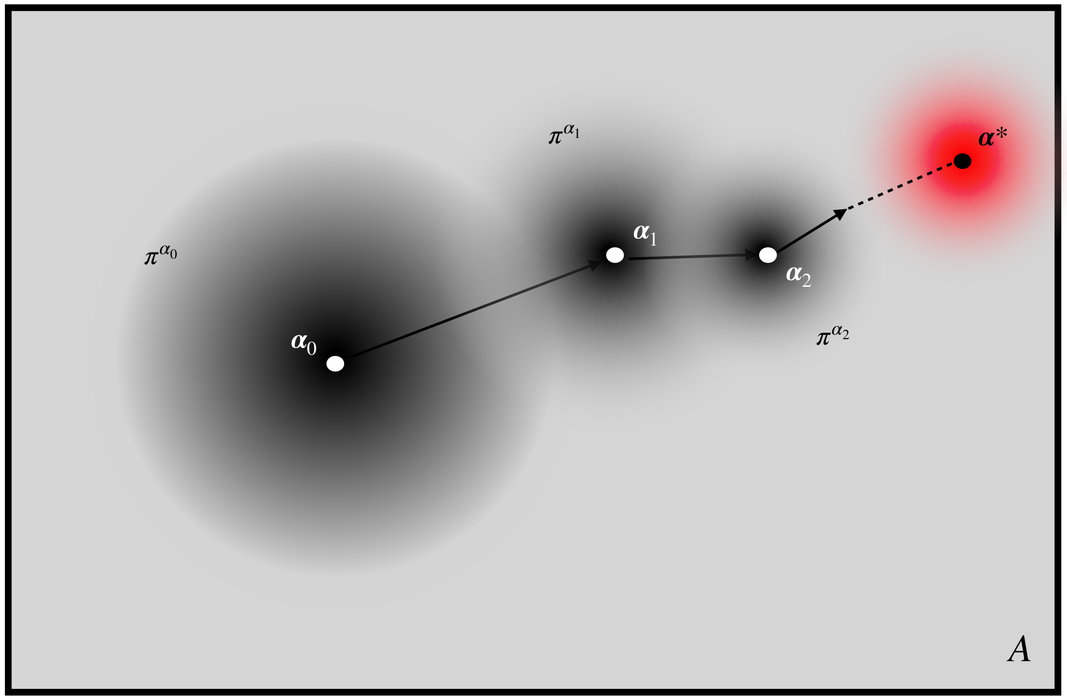
Ideally we would choose π^A such that its PDF is $p(\boldsymbol{\theta} \,|\, z, \boldsymbol{x})$, however π^A is specified before training when no approximation to $p(\boldsymbol{\theta} \,|\, z, \boldsymbol{x})$ is known.

Accuracy and efficiency

Tailoring the training procedure for Bayesian inversion

An adaptive training procedure akin to a trust-region optimisation scheme for $\boldsymbol{\theta}^* = \underset{\theta \in \Theta}{\text{argmax}}(p(\boldsymbol{\theta} \,|\, \boldsymbol{z}, \boldsymbol{x}))$

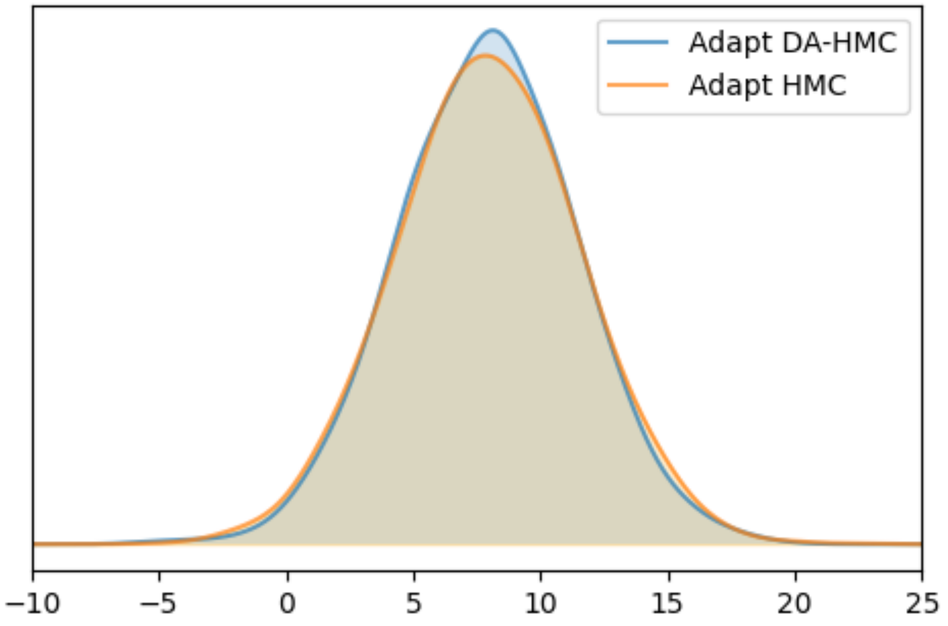
utilises a sequence of localised training measure $\pi^{\alpha_0}, \pi^{\alpha_1}, \pi^{\alpha_2}, \ldots$ to produce local models, and updates $\boldsymbol{\theta}$ based on local gradient information.

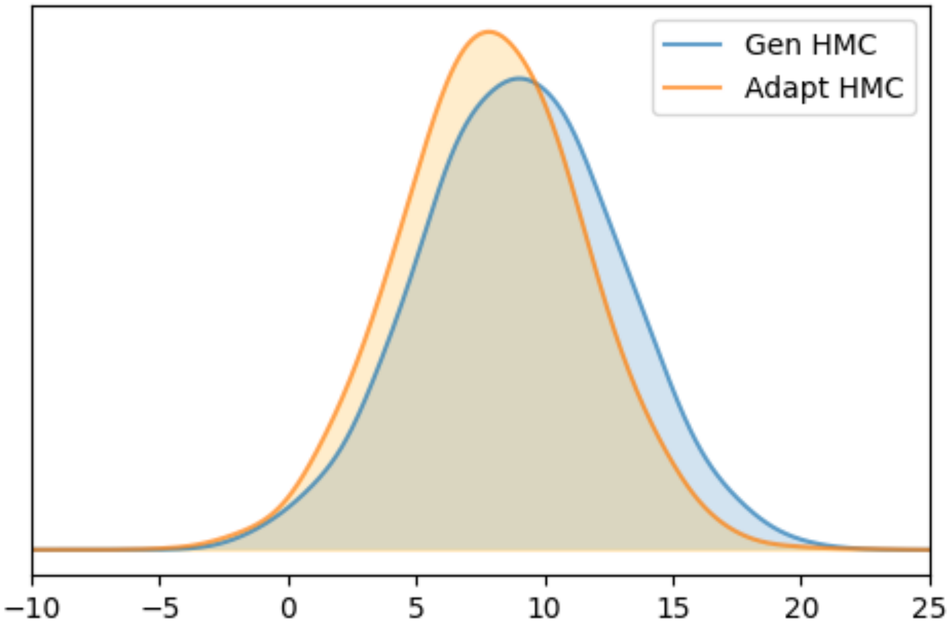


- Once $\boldsymbol{\theta}^*$ is found we train the surrogate over the Laplace approximation to the posterior and use this to commence MCMC sampling
- Samples from the MCMC are additionally used as training points throughout a warm-up period
- The final training measure π^A is $\hat{p}(\boldsymbol{\theta} \,|\, \boldsymbol{z}, \boldsymbol{x})$
- Speed of adaptive training is 925s vs 15,874 for a general π^A
- Accuracy of approximation is 5.02×10^{-4} vs 1.14×10^{-2}

Accuracy and efficiency

Comparing the adaptive surrogate induced posterior for a coefficient





Accuracy and efficiency

Relating the training loss to the posterior

Theorem: Let ψ denote the posterior distribution and $\hat{\psi}$ denote the surrogate induced approximation with training measure π^A. Under suitable assumptions, the error between ψ and $\hat{\psi}$ in the Hellinger distance satisfies

$$d_{\text{Hell}}(\psi, \hat{\psi})^2 \leq C \left\| \frac{d\psi}{d\pi^A} \right\|_\infty \int \frac{1}{\sigma^2} F(\hat{u}(\,\cdot\,,\cdot\,,\boldsymbol{\alpha}))\, d\pi^A(\boldsymbol{\alpha}, \sigma^2)$$
$$+ C \left\| \frac{d\psi}{d\pi^A} \right\|_\infty \int \frac{1}{\sigma^6} F(\hat{u}(\,\cdot\,,\cdot\,,\boldsymbol{\alpha}))^3\, d\pi^A(\boldsymbol{\alpha}, \sigma^2)\,.$$

Here $F(\hat{u}) = \|\mathscr{L}(\hat{u}, \boldsymbol{x}; \boldsymbol{\alpha})\|^2_{L_2(\Omega, \pi^\Omega)} + \|\mathscr{B}(\hat{u}, \boldsymbol{x}; \boldsymbol{\alpha})\|^2_{L_2(\partial\Omega, \pi^{\partial\Omega})}$ so the RHS is related to the loss function.

Notably, the Radon-Nykodym derivative $\left\| \frac{d\psi}{d\pi^A} \right\|_\infty \geq 1$ with equality only occurring when $\pi^A = \psi$.

This bounds the posterior accuracy based on the training loss, however the constants are unknown and training is not guaranteed to be successful a-priori.

Theoretical guarantees

Ensuring guaranteed accuracy of the posterior: Delayed acceptance

Delayed acceptance MCMC allows us to guarantee the accuracy of the posterior a-priori through the incorporation of a traditional solver. Suppose $\hat{\psi}$ is the surrogate posterior and $\tilde{\psi}$ is the posterior wrt a traditional solver. In this sampling scheme:

- The surrogate is used to propose and accept the next Monte Carlo sample standard

  Metropolis criteria: $$A_M(\boldsymbol{\theta}_{prop}, \boldsymbol{\theta}) = \min\left\{1, \frac{q(\boldsymbol{\theta}\,|\,\boldsymbol{\theta}_{prop})\hat{\psi}(\boldsymbol{\theta}_{prop})}{q(\boldsymbol{\theta}_{prop}\,|\,\boldsymbol{\theta})\hat{\psi}(\boldsymbol{\theta})}\right\}$$

- If accepted by the surrogate, a traditional solver is used to ‘validate’ the proposal via a

  secondary acceptance: $$A(\boldsymbol{\theta}_{prop}, \boldsymbol{\theta}) = \min\left\{1, \frac{\tilde{\psi}(\boldsymbol{\theta}_{prop})\hat{\psi}(\boldsymbol{\theta})}{\tilde{\psi}(\boldsymbol{\theta})\hat{\psi}(\boldsymbol{\theta}_{prop})}\right\}$$

- Detailed balance is preserved wrt $\tilde{\psi}$

Theoretical guarantees

Delayed acceptance: Why is this useful?

Many of the most efficient MCMC proposal distributions require information that is not readily returned by numerical solvers, (e.g. a Hamiltonian Monte Carlo iteration may require 50 or so gradients of the PDE solution wrt its parameters). The surrogate can handle these efficiently

	Surrogate only			Delayed-acceptance		
Proposal	**Time**	**ESS**	**Cost**	**Time**	**ESS**	**Cost**
RWMH	70.76	32.90	2.151	673.62	20.28	33.216
MALA	110.84	427.34	0.259	799.51	364.97	2.191
HMC	379.96	11263.20	0.034	1337.94	6497.04	0.206

Estimated cost with a full numerical solver: RWMH - 30.395

MALA - 4.680

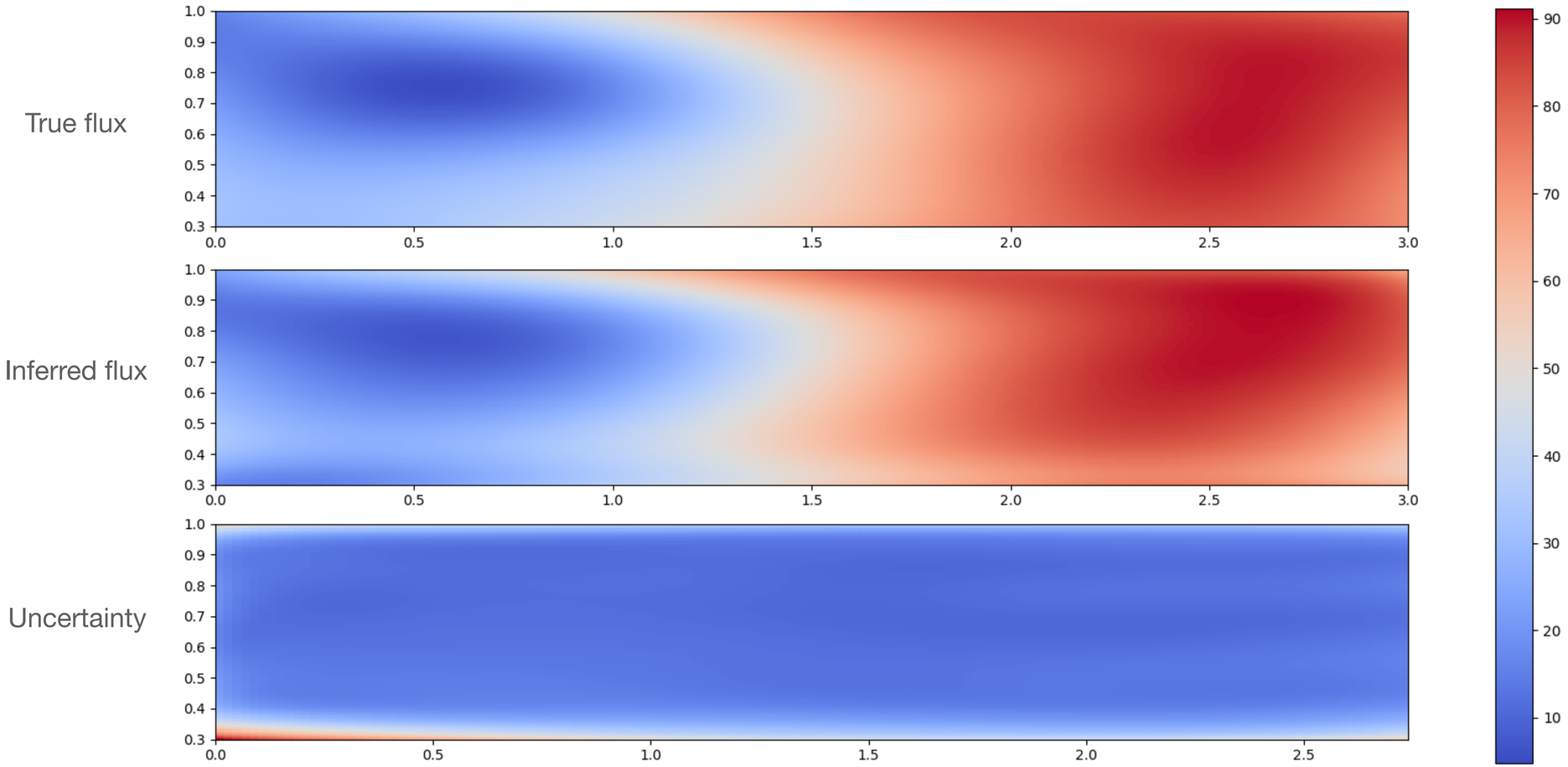
HMC - 8.967

Key requirements

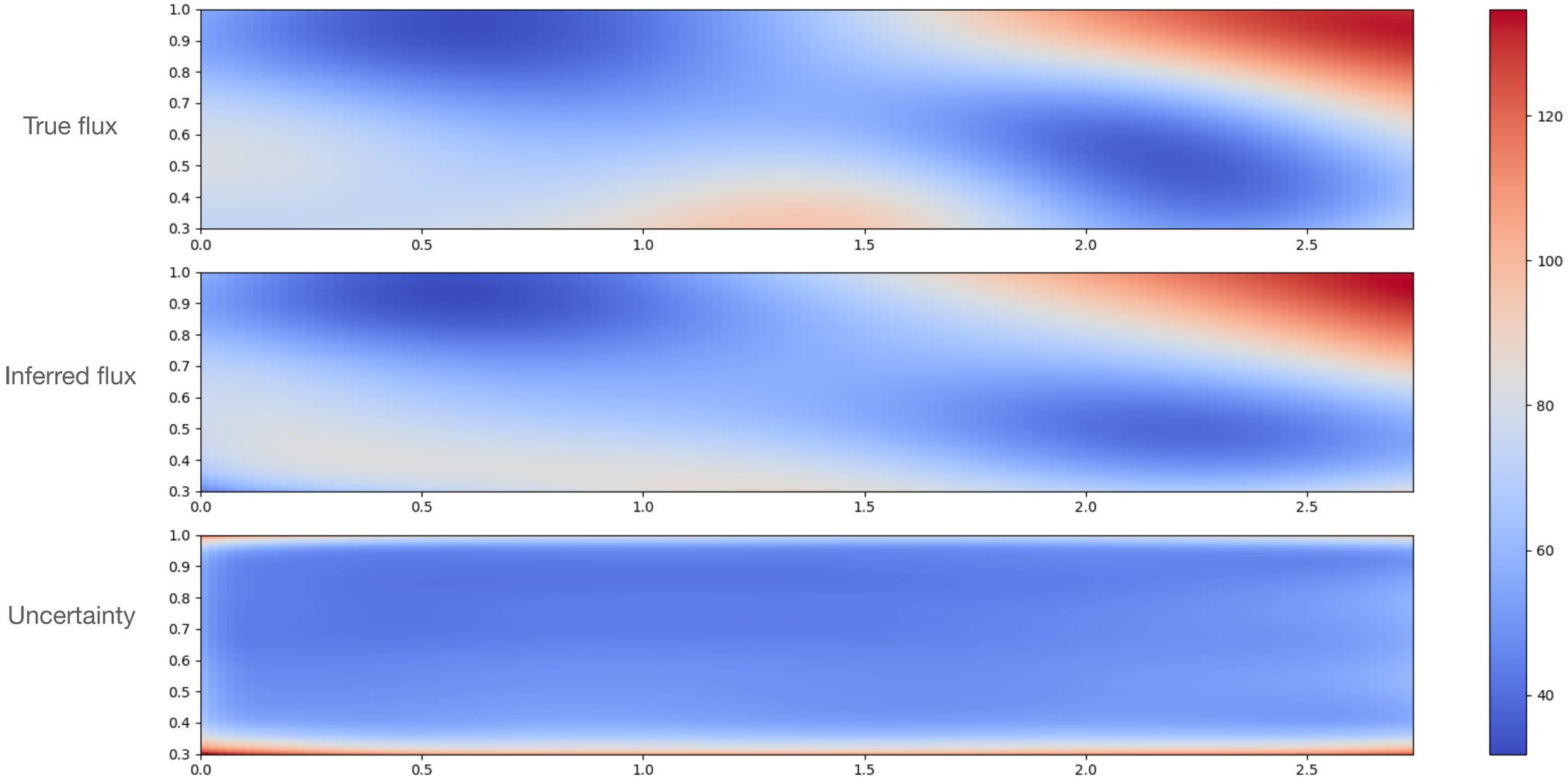
This method may influence future aircraft engine design so our approach must:

- **Allow interpretable priors -> Gaussian processes**
- **Accurate and efficient to train -> Adaptive training**
- **Theoretical guarantees -> Delayed acceptance**

Example


True flux
Inferred flux
Uncertainty
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.0
0.5
1.0
1.5
2.0
2.5
3.0
90
80
70
60
50
40
30
20
10

Example


True flux
Inferred flux
Uncertainty
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.0
0.5
1.0
1.5
2.0
2.5
120
100
80
60
40

Thank you!