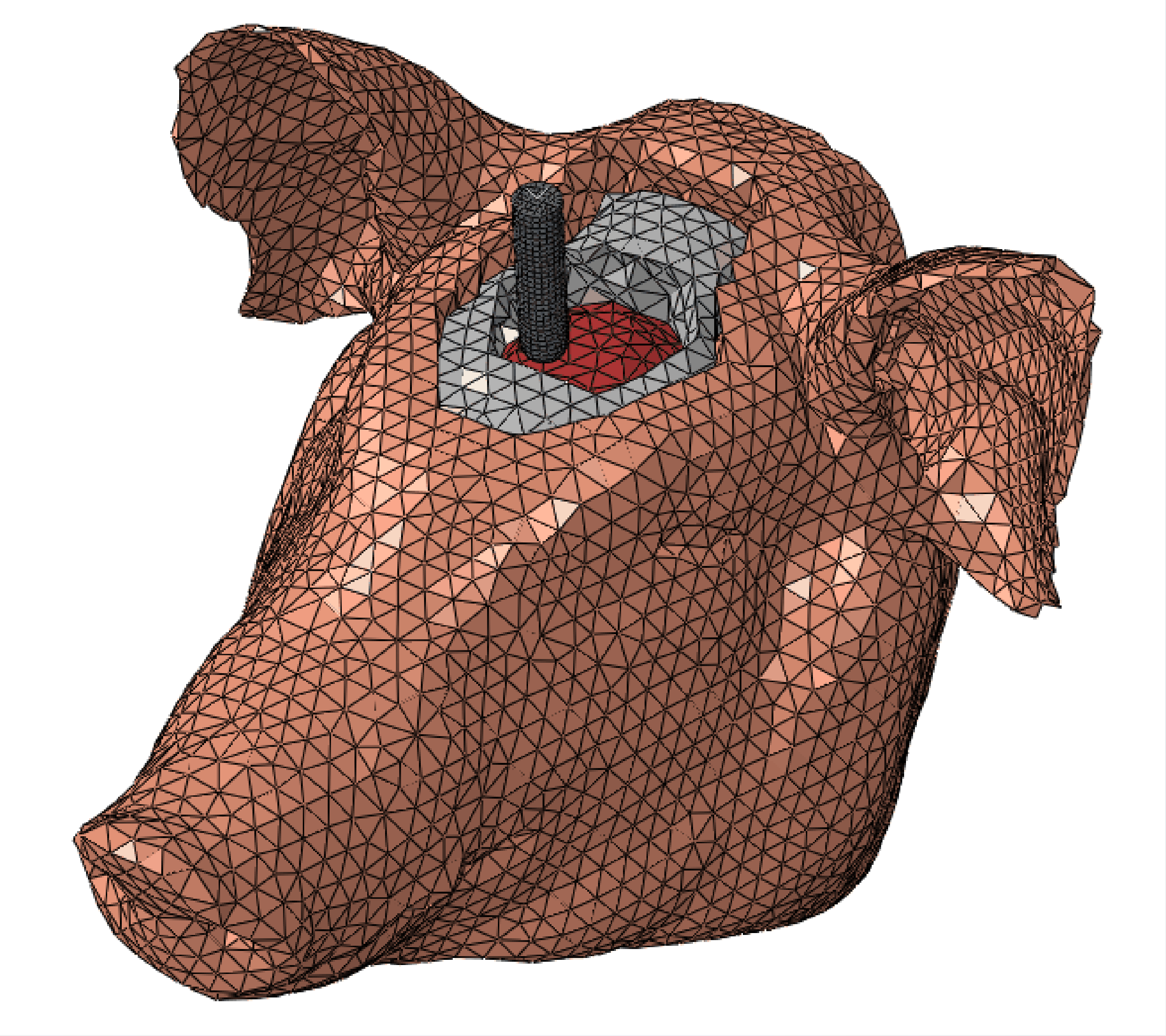
Multi-fidelity modeling and propagation of uncertainty in multiscale biological systems

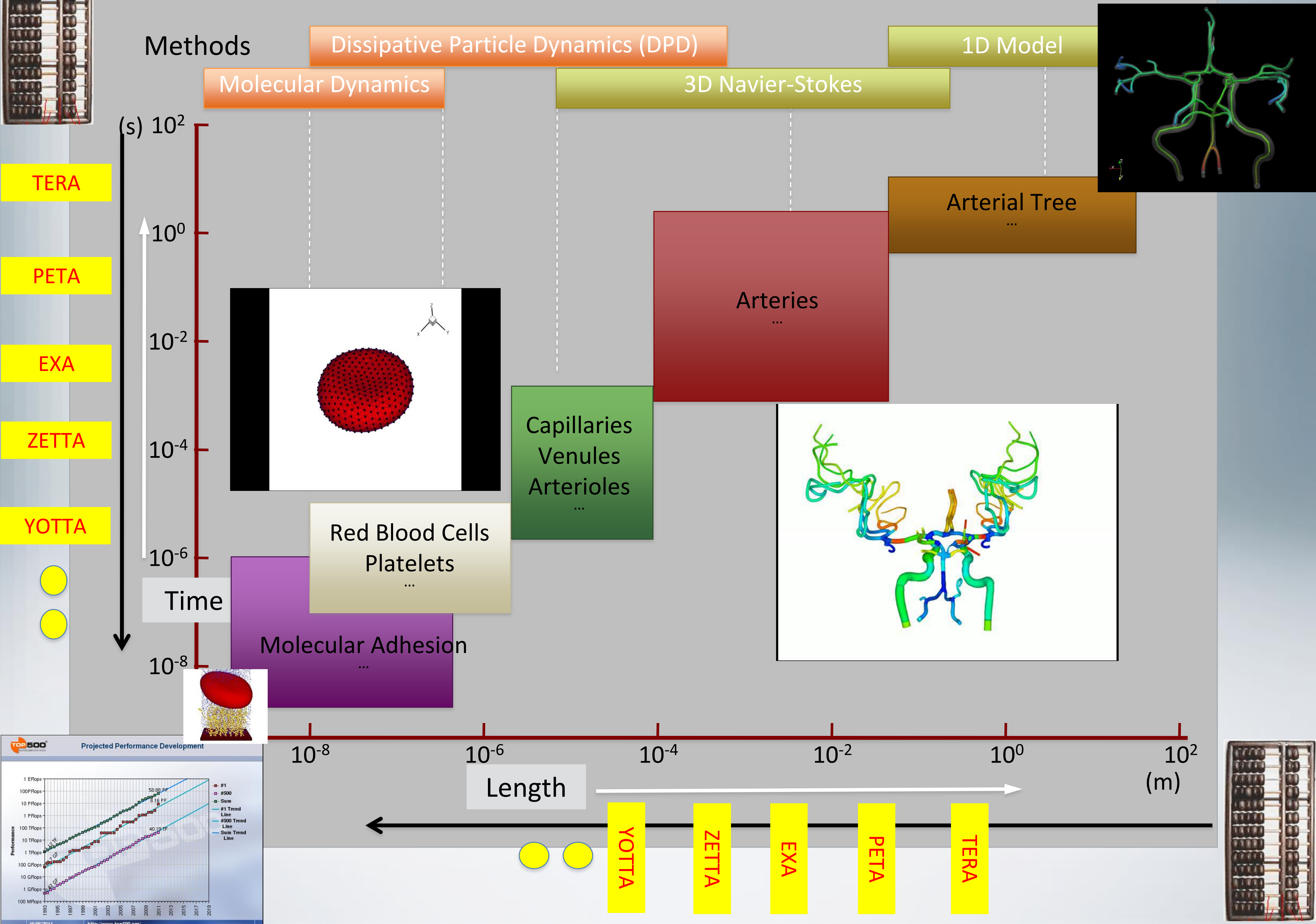


➢**Credit:**
T. Prevost, Subra Suresh & S. Socrate

George Em Karniadakis
Division of Applied Mathematics, Brown University
Also, @MIT & PNNL/DOE

The **CRUNCH** group: **https://www.brown.edu/research/projects/crunch/home**

Multi-scale Modeling of Brain Vasculature



At the 2010 MSM Consortium meeting, a white paper titled "Cell scale to macro-scale integration" was presented by the Cell-to-Macroscale WG to discuss the advantages, limitations and prospects of different numerical methods and modeling strategies to predict integrative physics and physiology that take place at cellular-scale to macro-scale in the human body.

1. **Fluid–structure interaction**
2. **Image-registration driven simulation**
3. **Three-dimensional to one-dimensional model coupling and interface conditions**
4. **Combined continuum-mesoscale-atomistic-level simulation**
5. **Interface and boundary conditions: accuracy and dynamical importance**
6. **Multiscale geometry representation and boundary conditions**
7. **Integration of imaging data with modeling and computer simulation**
8. **Direct versus indirect interactions between processes that operate at disparate scales**
9. **Uncertainty in materials properties, boundary conditions, and geometry**
10. **Sensitivity and uncertainty in multiscale and multi-physics integration**
11. **Cell models**

Special Issue of Journal Of Computational Physics
Vol. 244, 2013

Multi-scale Modeling and Simulation of Biological Systems

Ching-Long Lin, Grace C.Y. Peng and George Karniadakis

- *Cardiovascular systems (8 papers)*
- *Respiratory systems (3 papers)*
- *Cells/proteins (5 papers)*
- *Biochemical processes (2 papers)*
- *Bone mechanics (2 papers)*
- *Predicting surgery outcomes (1 paper)*

"…Because I had worked in the closest possible ways with physicists and engineers, I knew that our data can never be precise…"

Norbert Wiener

Solving Differential Equations from Measurements Only!

"…once we allow that we don't know f(x), but do know some things, it becomes natural to take a Bayesian approach"

Persi Diaconis, Stanford (1988)

✓ **Remove the tyranny of Grids! And of serious Math!**

✓ **Use noisy measurements - Predict with uncertainty!**

✓ **Execute Poincare's will!**

Outline

I. Gaussian Process Regression

II. Noisy Data & Multi-fidelity

III. PDEs via Machine Learning

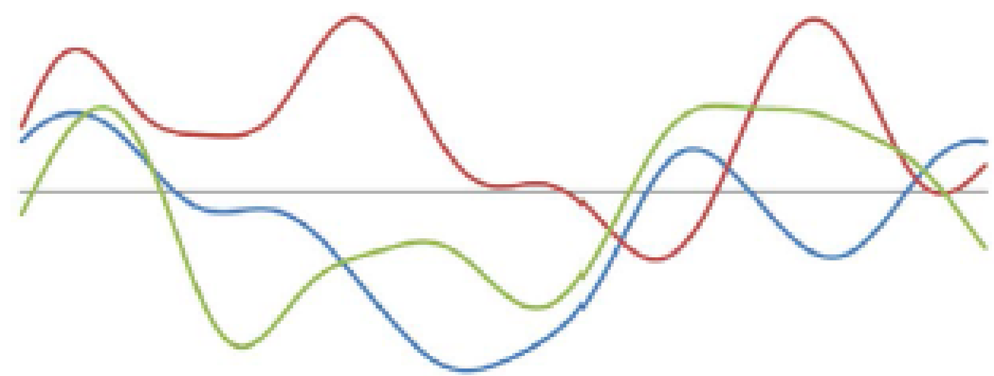
Gaussian processes

Starting point: The multivariate Gaussian distribution

$$p(\underbrace{f_1, f_2, \cdots, f_s}_{\mathbf{f}_A}, \underbrace{f_{s+1}, f_{s+2}, \cdots, f_N}_{\mathbf{f}_B}) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}) \qquad \boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_A \\ \boldsymbol{\mu}_B \end{bmatrix} \text{ and } \mathbf{K} = \begin{bmatrix} \mathbf{K}_{AA} & \mathbf{K}_{AB} \\ \mathbf{K}_{BA} & \mathbf{K}_{BB} \end{bmatrix}$$

Generalization: The Gaussian process

$$\underbrace{\boldsymbol{\mu}_\infty = \begin{bmatrix} \boldsymbol{\mu}_{\mathbf{f}} \\ \cdots \\ \cdots \end{bmatrix}}_{\textit{mean function}} \text{ and } \underbrace{\mathbf{K}_\infty = \begin{bmatrix} \mathbf{K}_{\mathbf{ff}} & \cdots \\ \cdots & \cdots \end{bmatrix} \qquad \mathbf{K}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)}_{\textit{covariance function}}$$



Samples from a GP prior

Priors over functions: $f \sim \mathcal{GP}(\mu(x), K(\mathbf{x}, \mathbf{x}'; \theta))$

Infinite dimensional model, but finitely many observations: The marginalization property

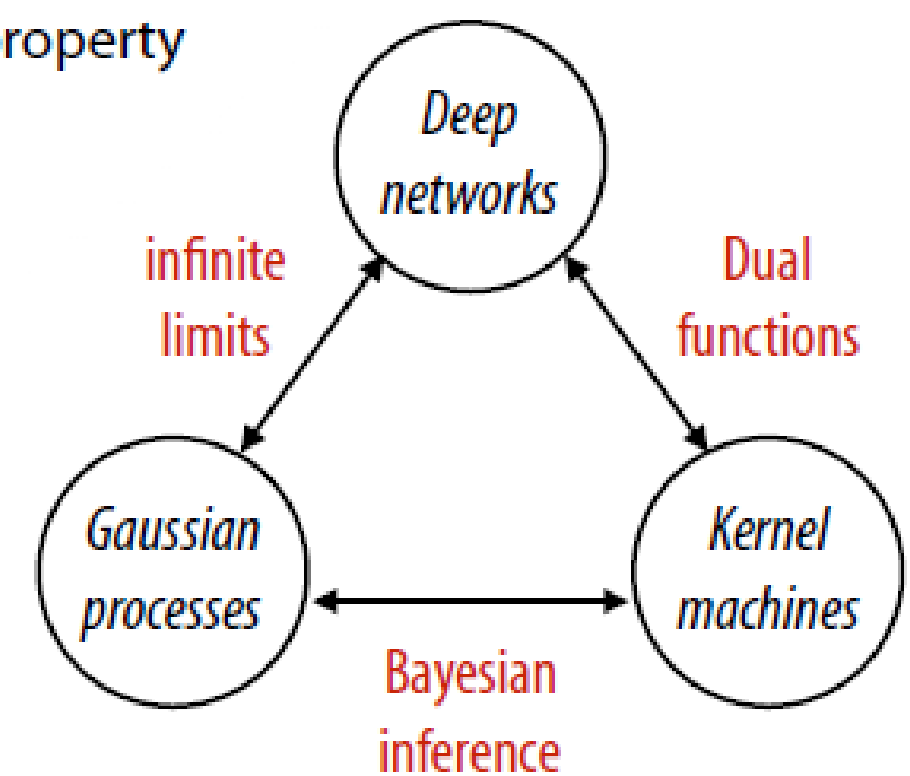
$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$

$$p(\mathbf{f}_A) = \int_{\mathbf{f}_B} p(\mathbf{f}_A, \mathbf{f}_B) d\mathbf{f}_B = \mathcal{N}(\boldsymbol{\mu}_A, \mathbf{K}_{AA})$$

Posterior is also Gaussian:

$$p(\mathbf{f}_A, \mathbf{f}_B) \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K}). \quad \text{Then:}$$

$$p(\mathbf{f}_A | \mathbf{f}_B) = \mathcal{N}(\boldsymbol{\mu}_A + \mathbf{K}_{AB}\mathbf{K}_{BB}^{-1}(\mathbf{f}_B - \boldsymbol{\mu}_B), \mathbf{K}_{AA} - \mathbf{K}_{AB}\mathbf{K}_{BB}^{-1}\mathbf{K}_{BA})$$



Rasmussen, C. E. Gaussian processes for machine learning (2006)

Nonlinear regression with Gaussian processes

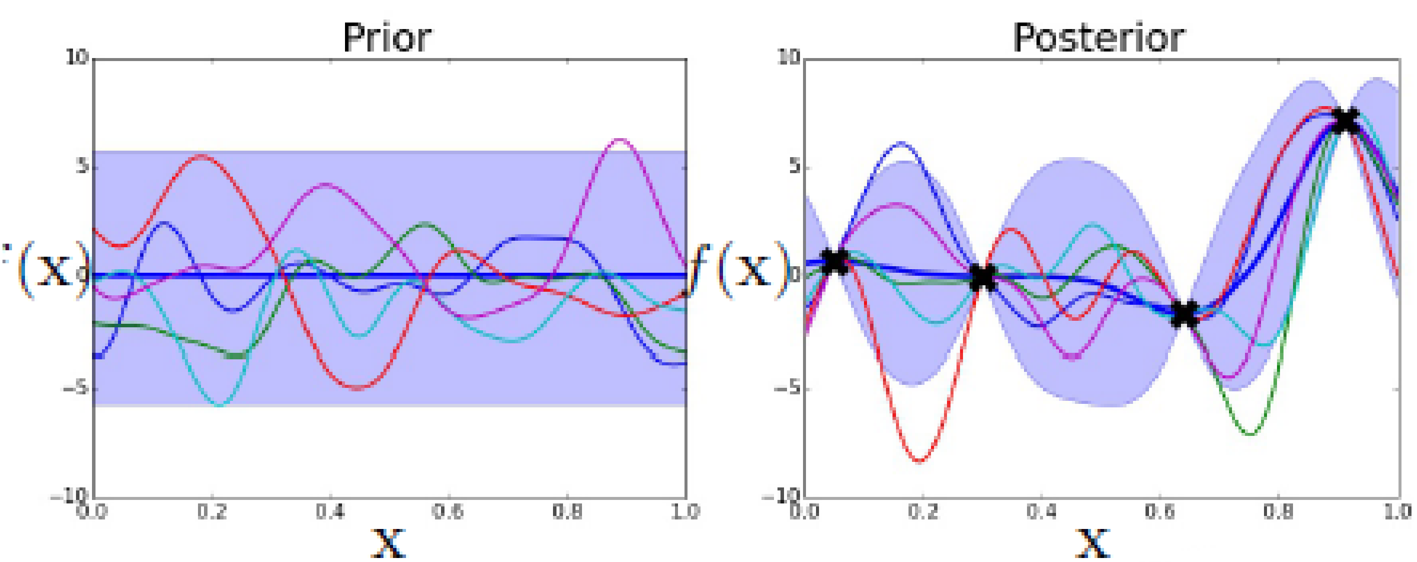
$$\boxed{y = f(\mathbf{x}) + \epsilon, \qquad f \sim \mathcal{GP}(\mu(x), K(\mathbf{x}, \mathbf{x}'; \theta))}$$

History:

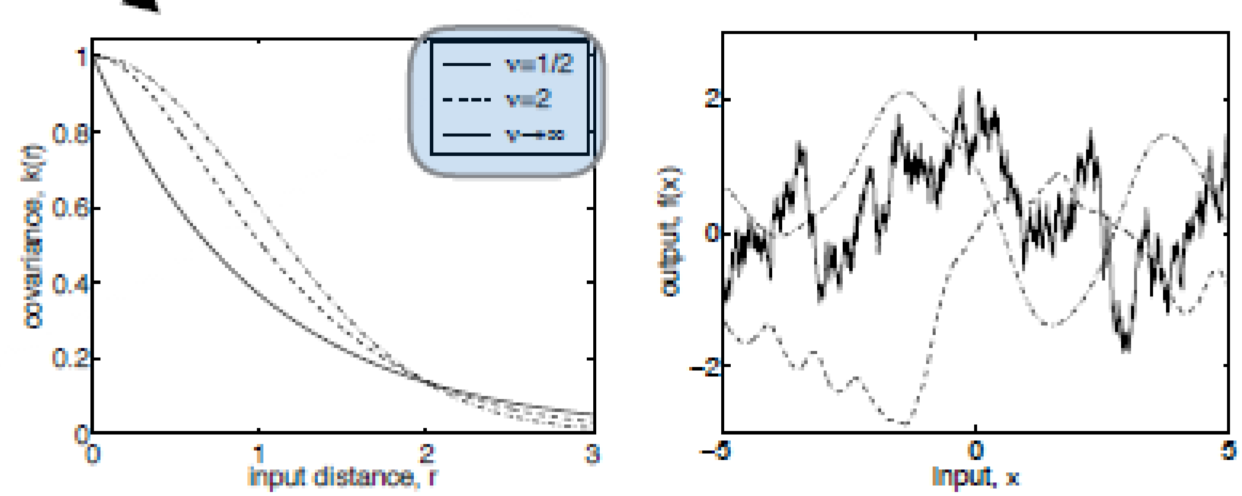
- Wiener–Kolmogorov filtering (1940)
- Kriging (spatial statistics, 1970)
- GP regression (machine learning, 1996)

Workflow:

- Assign a Gaussian process (GP) prior over functions
- Given a training set of observations (x,y) calibrate the GP hyper-parameters
- Use the conditional posterior [f|y] to infer predictions for unobserved x's with quantified uncertainty

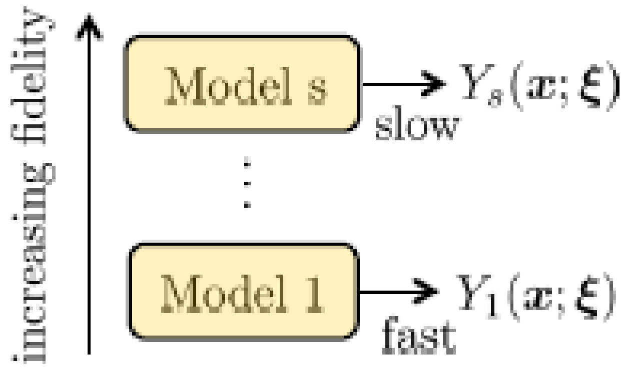


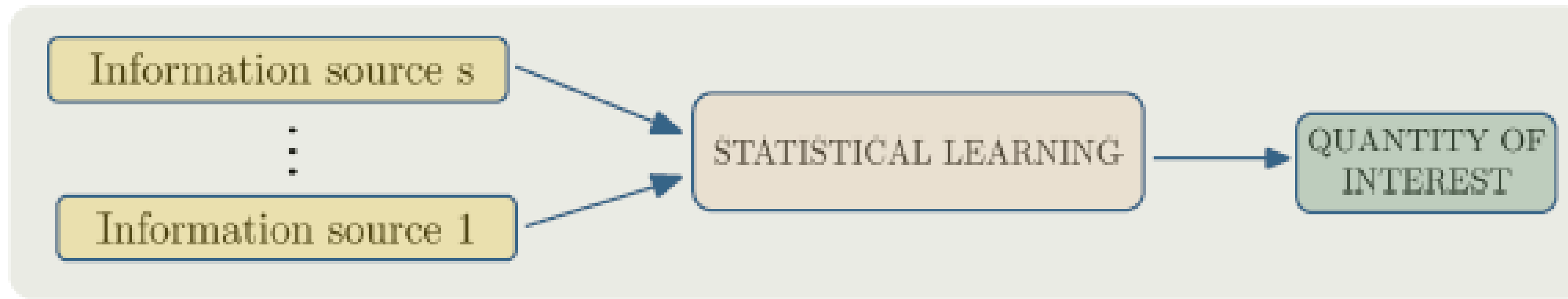
covariance function	expression
constant	σ_0^2
linear	$\sum_{d=1}^{D} \sigma_d^2 x_d x'_d$
polynomial	$(\mathbf{x} \cdot \mathbf{x}' + \sigma_0^2)^p$
squared exponential	$\exp(-\frac{r^2}{2\ell^2})$
Matérn	$\frac{1}{2^{\nu-1}\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{\ell} r\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}}{\ell} r\right)$
exponential	$\exp(-\frac{r}{\ell})$
γ-exponential	$\exp\left(-(\frac{r}{\ell})^{\gamma}\right)$
rational quadratic	$(1 + \frac{r^2}{2\alpha\ell^2})^{-\alpha}$
neural network	$\sin^{-1}\left(\frac{2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}}'}{\sqrt{(1+2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}})(1+2\tilde{\mathbf{x}}'^\top \Sigma \tilde{\mathbf{x}}')}}\right)$



Rasmussen, C. E. Gaussian processes for machine learning (2006)

Multi-fidelity modeling

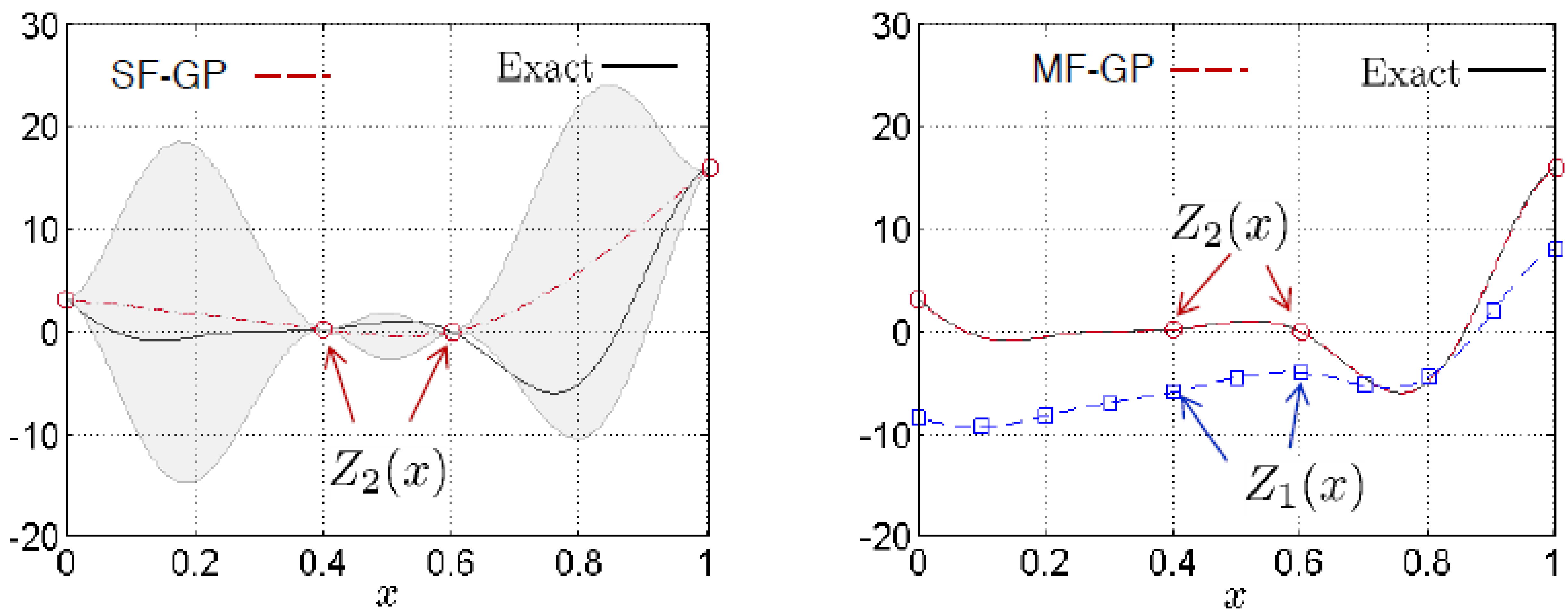




Number of runs is limited by time
and computational resources

We cannot compute at all $(\boldsymbol{x};\boldsymbol{\xi})$

Prediction of $Z_i(\boldsymbol{x}) = \mathbb{E}[f(Y_i(\boldsymbol{x};\boldsymbol{\xi}))]$ is a
problem of statistical inference

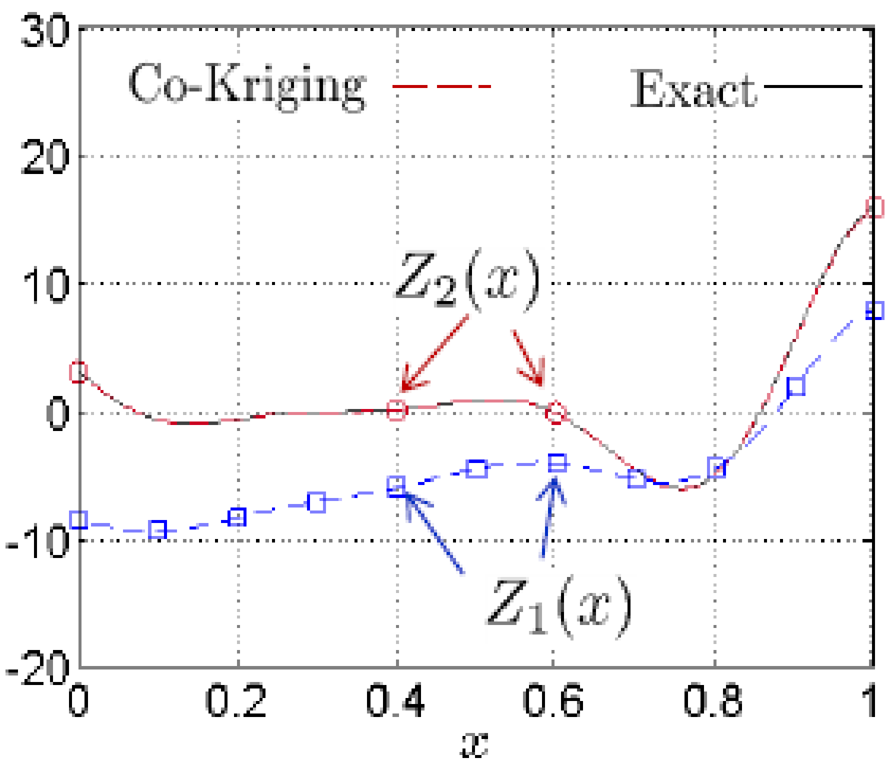


Multi-fidelity modeling with GPs

Auto-regressive model: AR1

$$f_t(\mathrm{x}) = \rho_{t-1}(\mathrm{x}) f_{t-1}(\mathrm{x}) + \delta_t(\mathrm{x})$$

$$t = 1, \ldots, s$$



Predicting the Output from a Complex Computer Code When Fast Approximations Are Available

M. C. Kennedy; A. O'Hagan

Biometrika, Vol. 87, No. 1. (Mar., 2000), pp. 1-13.

Predictive posterior

$$p(f_*|\boldsymbol{y}, \boldsymbol{X}, \boldsymbol{x}_*) = \mathcal{N}(f_*|\mu_*, \sigma_*^2),$$
$$\mu_*(\boldsymbol{x}_*) = \boldsymbol{k}_{*N}(\boldsymbol{K} + \sigma_\epsilon^2 \boldsymbol{I})^{-1}\boldsymbol{y},$$
$$\sigma_*^2(\boldsymbol{x}_*) = k_{**} - \boldsymbol{k}_{*N}(\boldsymbol{K} + \sigma_\epsilon^2 \boldsymbol{I})^{-1}\boldsymbol{k}_{N*},$$

$$\begin{array}{cc} N_1 & \\ N_2 & \end{array}\begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix}$$

Block covariance matrix

Theorem (LeGratiet, 2014):

The predictive posterior of the recursive scheme has exactly the same distribution with the the fully coupled model given a nested experimental design.

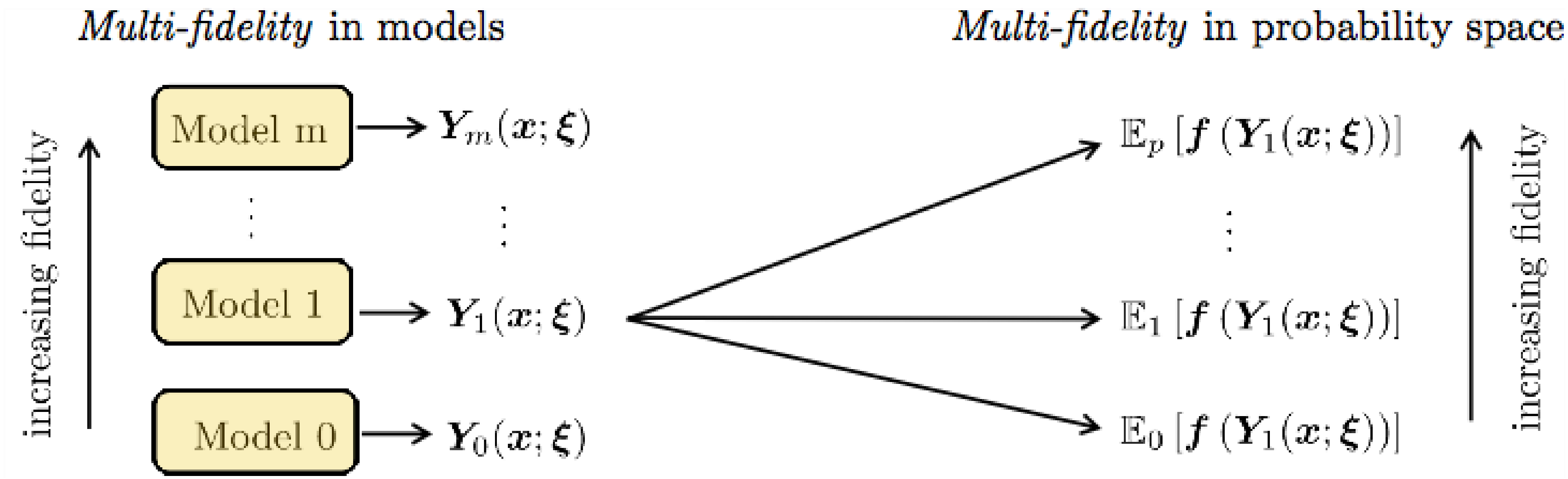
Key idea: Replace f_{t-1} with the GP posterior of the previous level $\tilde{f}_{t-1}$

$$f_t(\mathrm{x}) = \rho_{t-1}(\mathrm{x}) f_{t-1}(\mathrm{x}) + \delta_t(\mathrm{x}) \qquad \tilde{f}_{t-1} \sim f_{t-1}|\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_{t-1}$$

M.C Kennedy, and A. O'Hagan. *Predicting the output from a complex computer code when fast approximations are available*, 2000.

L. Le Gratiet, and J. Garnier, *"Recursive co-kriging model for design of computer experiments with multiple levels of fidelity." International Journal for Uncertainty Quantification*, 2014

A general framework




PROCEEDINGS OF THE ROYAL SOCIETY A

rspa.royalsocietypublishing.org

Research

Article submitted to journal

Multi-fidelity modeling via recursive co-kriging and Gaussian Markov random fields

P. Perdikaris[1], D. Venturi[1], J.O. Royset[2], and G.E. Karniadakis[1]

[1]Division of Applied Mathematics, Brown University, Providence, RI 02912, USA
[2]Operations Research Department, Naval Postgraduate School, Monterey, CA 93943, USA

$$\mathbb{E}_{k+1}[f(\mathbf{Y}_l(\mathbf{x};\xi))] = \rho_{k+1}\mathbb{E}_k[f(\mathbf{Y}_l(\mathbf{x};\xi))] + \delta_{k+1}(\mathbf{x}), \quad k \le p, \quad l \le m$$

$$\begin{pmatrix} \mathbb{E}_1\left[\boldsymbol{f}\left(\boldsymbol{Y}_1\right)\right] & \mathbb{E}_1\left[\boldsymbol{f}\left(\boldsymbol{Y}_2\right)\right] & \cdots & \mathbb{E}_1\left[\boldsymbol{f}\left(\boldsymbol{Y}_m\right)\right] \\ \mathbb{E}_2\left[\boldsymbol{f}\left(\boldsymbol{Y}_1\right)\right] & \mathbb{E}_2\left[\boldsymbol{f}\left(\boldsymbol{Y}_2\right)\right] & \cdots & \mathbb{E}_2\left[\boldsymbol{f}\left(\boldsymbol{Y}_m\right)\right] \\ \vdots & & \ddots & \vdots \\ \mathbb{E}_p\left[\boldsymbol{f}\left(\boldsymbol{Y}_1\right)\right] & \mathbb{E}_p\left[\boldsymbol{f}\left(\boldsymbol{Y}_2\right)\right] & \cdots & \mathbb{E}_p\left[\boldsymbol{f}\left(\boldsymbol{Y}_m\right)\right] \end{pmatrix}$$

Nonlinear multi-fidelity modeling via probabilistic deep learning

We generalize the classical linear scheme of Kennedy and O'Hagan

AR1

$$f_2(\boldsymbol{x}) = \rho f_1(\boldsymbol{x}) + \delta_2(\boldsymbol{x})$$

$$f_1 \sim \mathcal{GP}(f_1|0, K_1(\boldsymbol{x}, \boldsymbol{x}'; \theta_1))$$

$$\delta_2 \sim \mathcal{GP}(\mu_{\delta_2}, K_2(\boldsymbol{x}, \boldsymbol{x}'; \theta_2))$$

to a ***compositional and robust*** model inspired by ***deep learning*** that can learn complex ***nonlinear and space-dependent cross-correlations***

NARGP

$$f_2(\boldsymbol{x}) = g_2(\boldsymbol{x}, f_1(\boldsymbol{x}))$$

$$f_1 \sim \mathcal{GP}(f_1|0, K_1(\boldsymbol{x}, \boldsymbol{x}'; \theta_1))$$

$$g_2 \sim \mathcal{GP}(g_2|0, K_2((\boldsymbol{x}, f_1(\boldsymbol{x})), (\boldsymbol{x}', f_1(\boldsymbol{x}')); \theta_2))$$

$$K_2((\boldsymbol{x}, f_1(\boldsymbol{x})), (\boldsymbol{x}', f_1(\boldsymbol{x}')); \theta_2)) = k_{t_\rho}(\boldsymbol{x}, \boldsymbol{x}'; \theta_{t_\rho}) \times k_{t_f}(f_1(\boldsymbol{x}), f_1(\boldsymbol{x}'); \theta_{t_f}) + k_{t_\delta}(\boldsymbol{x}, \boldsymbol{x}'; \theta_{t_\delta})$$

Space-dependent nonlinear map from the low- to the high-fidelity model

***example for 2 levels of fidelity — extension to more levels and deeper networks is straightforward.**

M.C Kennedy, and A. O'Hagan. *Predicting the output from a complex computer code when fast approximations are available*, 2000.

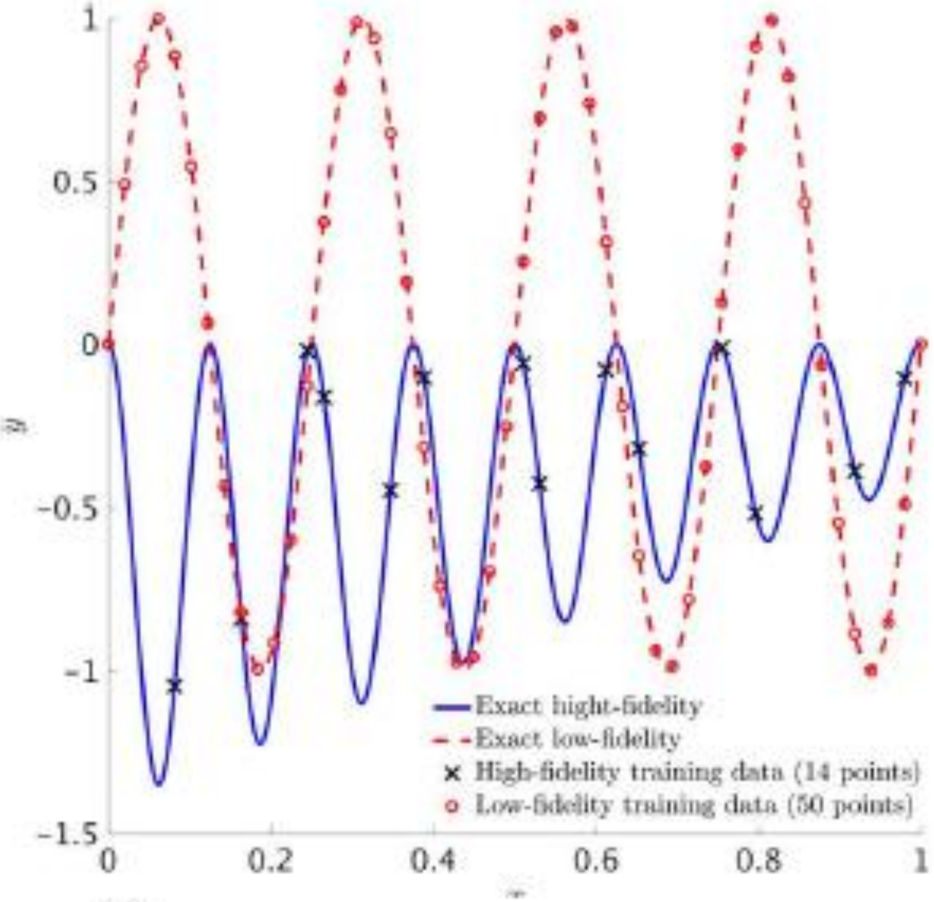
P. Perdikaris, M. Raissi, A. Damianou, N.D Lawrence, G. E Karniadakis, *Nonlinear information fusion algorithms for robust multi-fidelity modeling*, 2016.

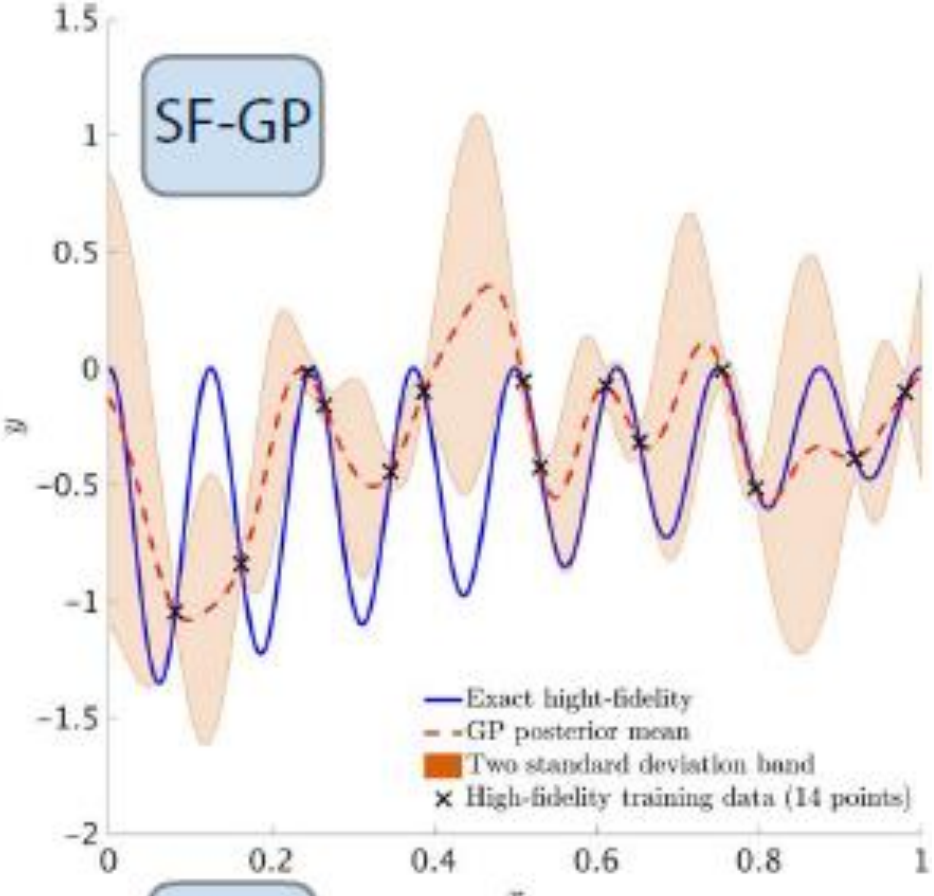
A deceptively simple example

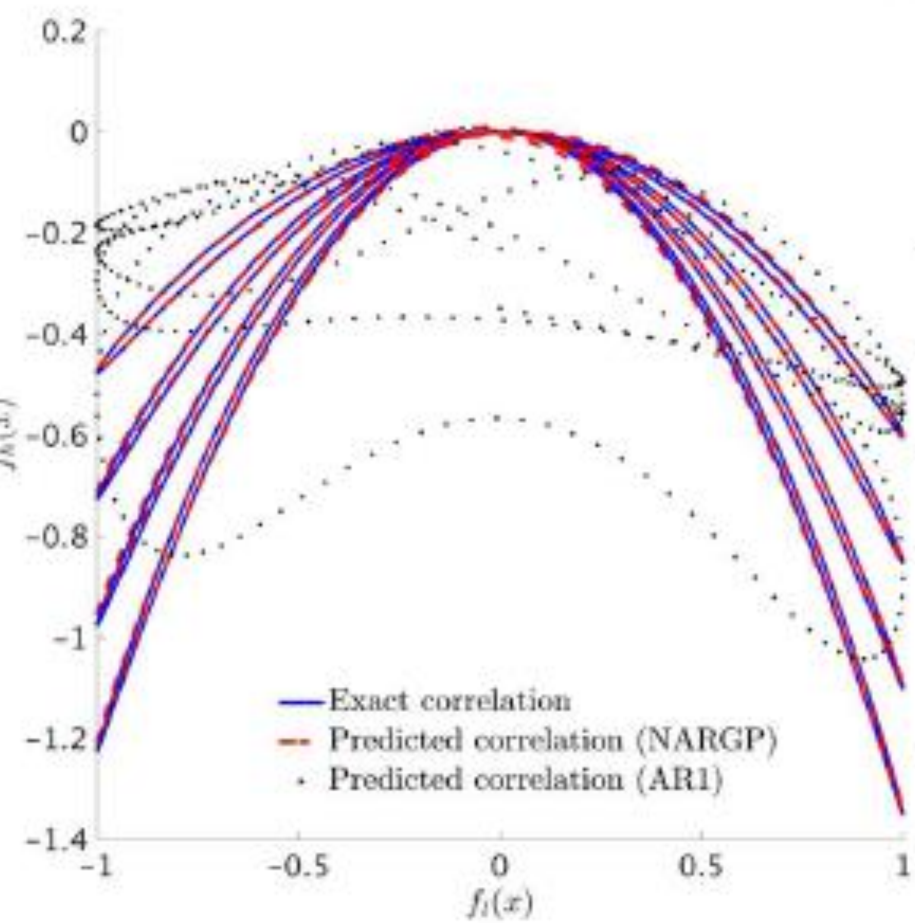
$$f_l(x) = \sin(8\pi x)$$

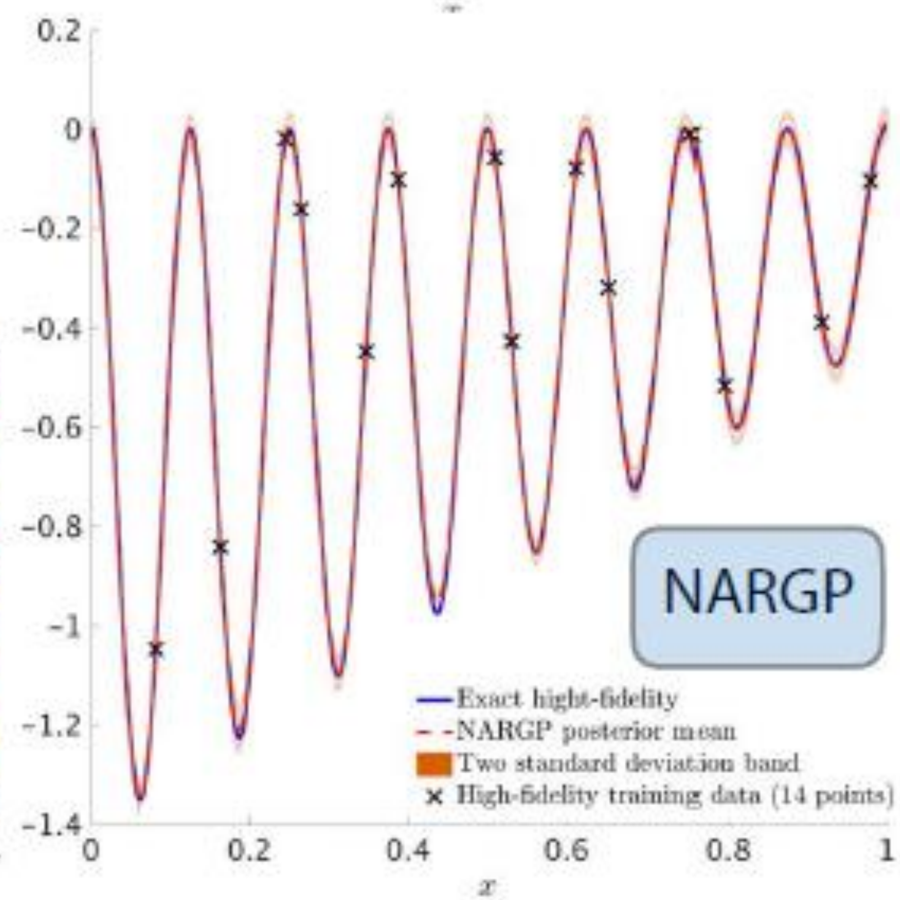
non-uniform scaling and quadratic nonlinearity

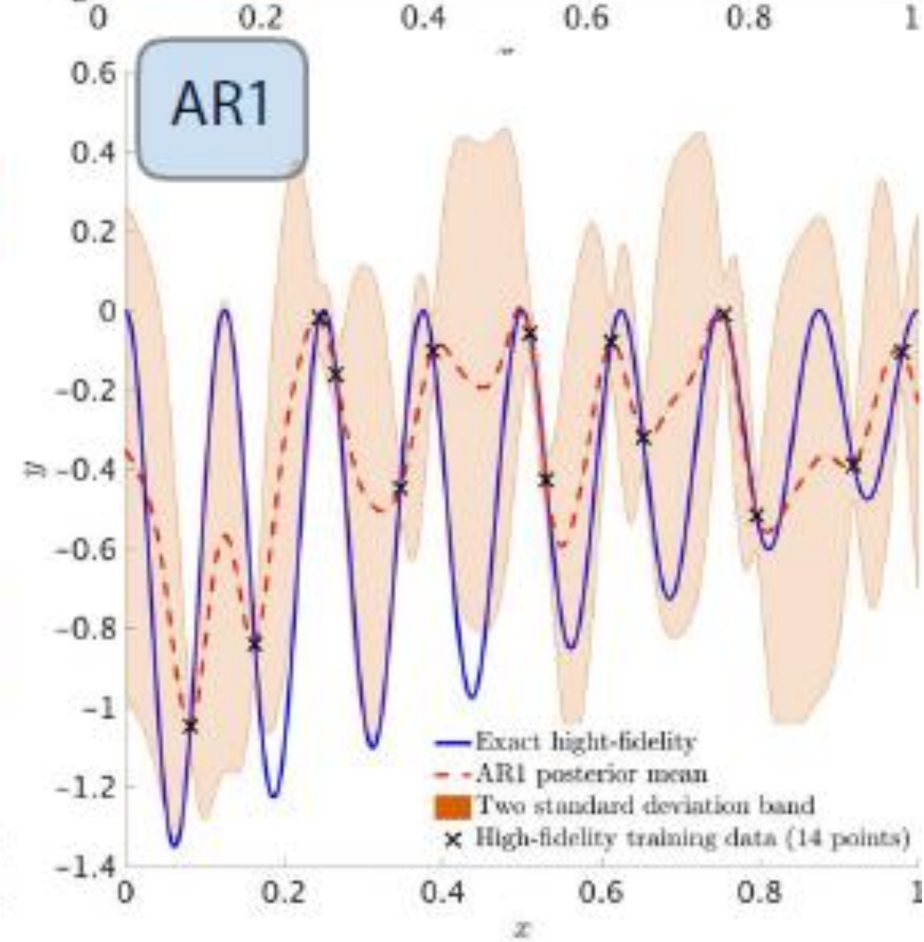
$$f_h(x) = (x - \sqrt{2}) f_L^2(x)$$





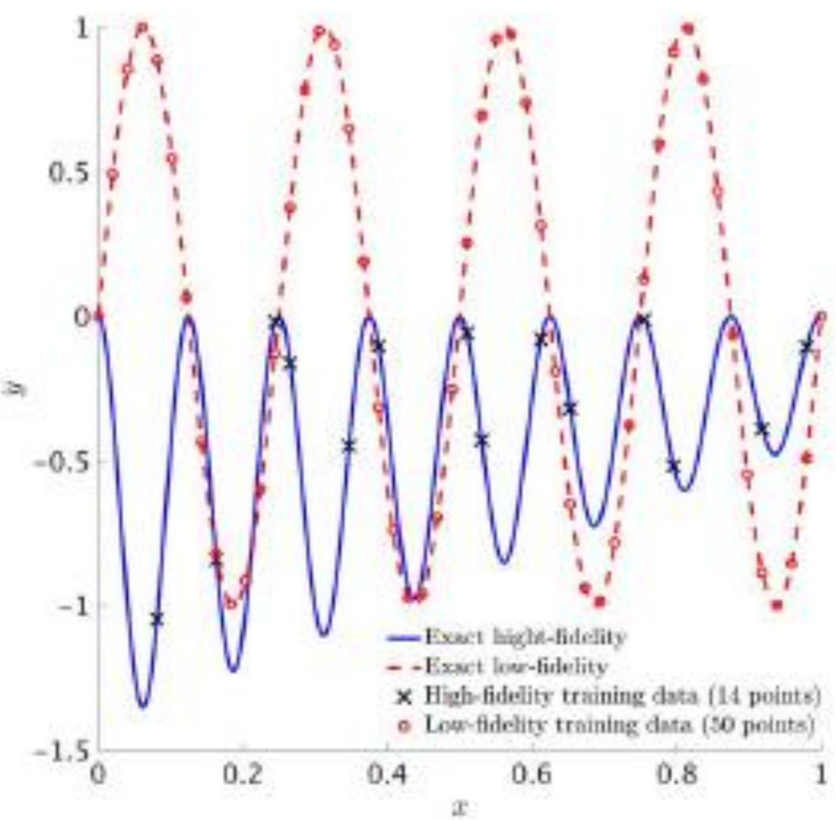


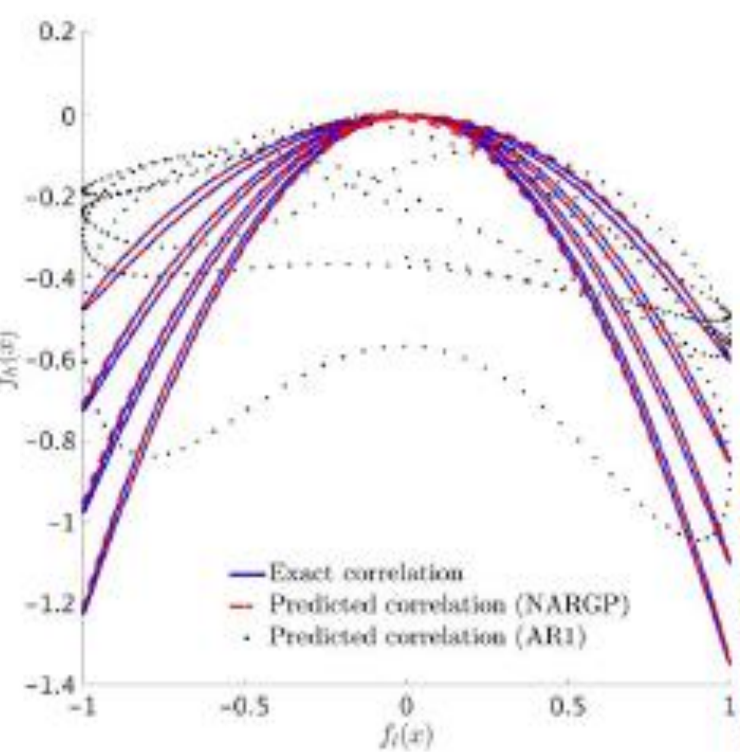




P. Perdikaris, M. Raissi, A. Damianou, N.D Lawrence, G. E Karniadakis, *Nonlinear information fusion algorithms for robust multi-fidelity modeling*, 2016.

Nonlinear multi-fidelity modeling via probabilistic deep learning





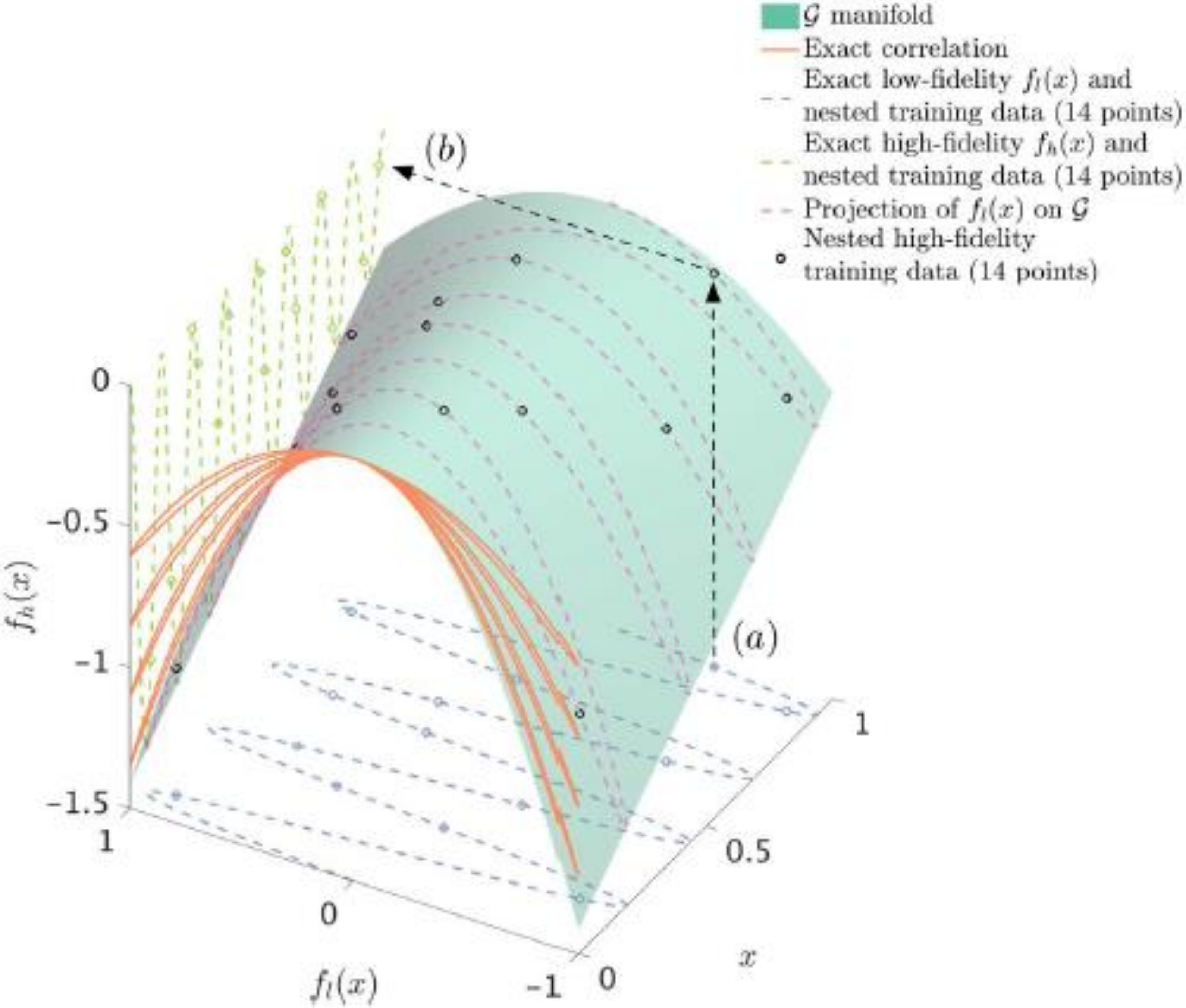


Figure 4: *A pedagogical example*: The NARGP algorithm can capture complex non-linear, non-functional, and space dependent cross-correlations by inferring the nonlinear latent manifold $\mathcal{G}$ that governs the functional relation between the inputs x and the outputs of the low- and high-fidelity models $f_l(x)$ and $f_h(x)$, respectively. (a) The low-fidelity model is projected onto the nonlinear latent manifold $\mathcal{G}$ that is inferred using the deep non-parametric representation of Eq. 2.11. (b) The high-fidelity function $f_h(x)$ is recovered by a smooth mapping from the $\mathcal{G}$ manifold to the high-fidelity data.

P. Perdikaris, M. Raissi, A. Damianou, N.D Lawrence, G. E Karniadakis, *Nonlinear information fusion algorithms for robust multi-fidelity modeling*, 2016.

Multi-fidelity Bayesian optimization

Goal: Identify a set of parameters that generates a response matching a target performance $y^\star$

$$\mathbf{x}^\star = \min_{\mathbf{x}\in\mathbb{R}^d} ||f(\mathbf{x}) - y^\star||$$ (potentially intractable)

Idea: We model the response of a system using deep multi-fidelity surrogates

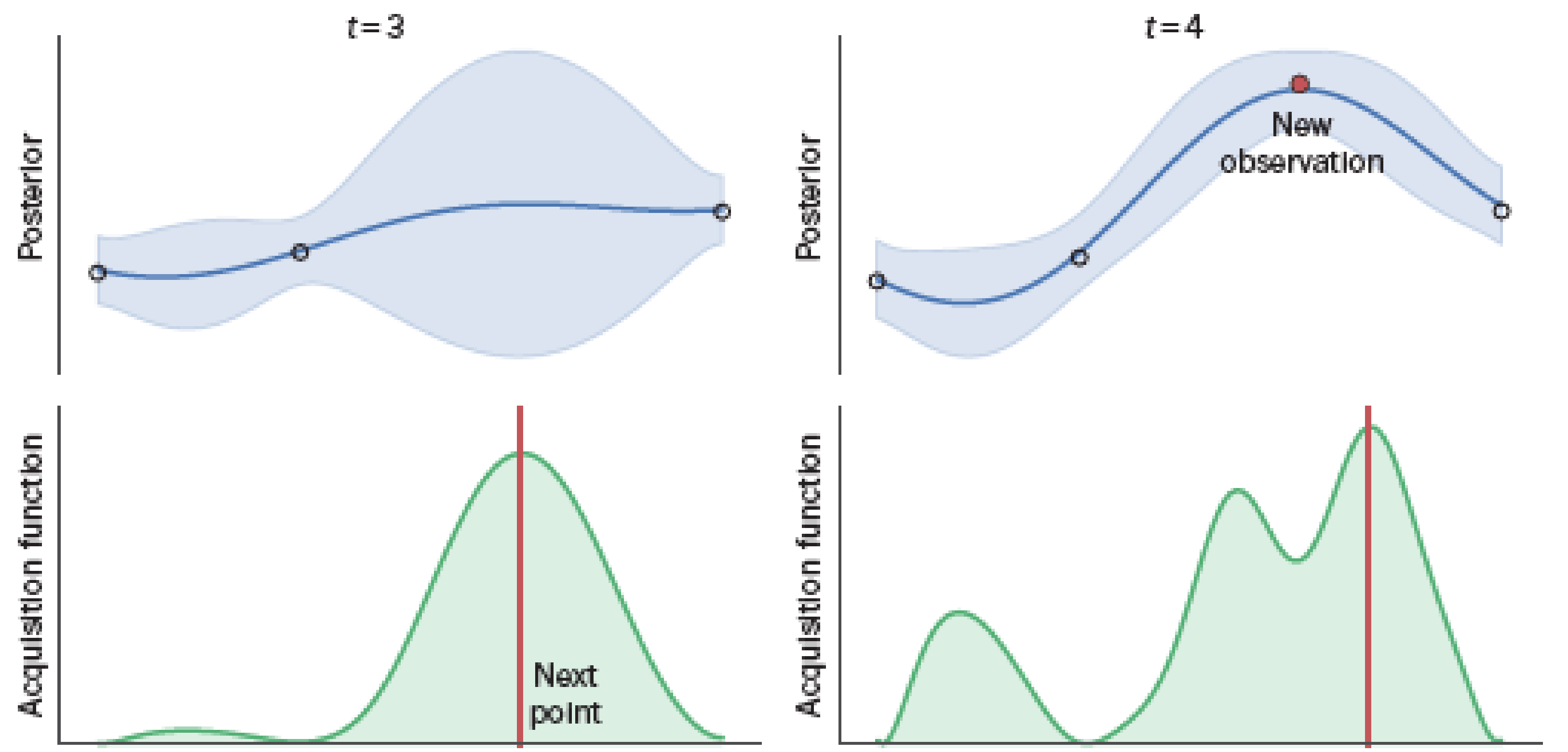
$$y = f_t(f_{t-1}(...(f_1(\mathbf{x})))), \quad f_i \sim \mathcal{GP}(\mu_i(\mathbf{x}), \Sigma_t)$$

Setup: Black-box and expensive to evaluate objective function, noisy observations, no gradients

Then the surrogate posterior distribution along with an acquisition function suggest a sampling plan than balances exploration vs exploitation towards identifying a global optimum

The optimization problem is transformed to:

$$\mathbf{x}_{n+1} = \arg\max_{\mathbf{x}\in\mathbb{R}^d} \alpha(\mathbf{x}; \mathcal{D}_n, \mathcal{M}_n)$$



Example: 1D function maximization

Remark:

Acquisition functions aim to balance the trade-off between exploration and exploitation.

P. Perdikaris, and G.E Karniadakis. "Model inversion via multi-fidelity Bayesian optimization." J. R. Soc. Interface (2016)

B. Shahriari, Swersky, K., Wang, Z., Adams, R. P., & de Freitas, N., "Taking the human out of the loop: A review of bayesian optimization." Proceedings of the IEEE, 104(1), 148-175, (2016)

Calibration of blood flow simulations

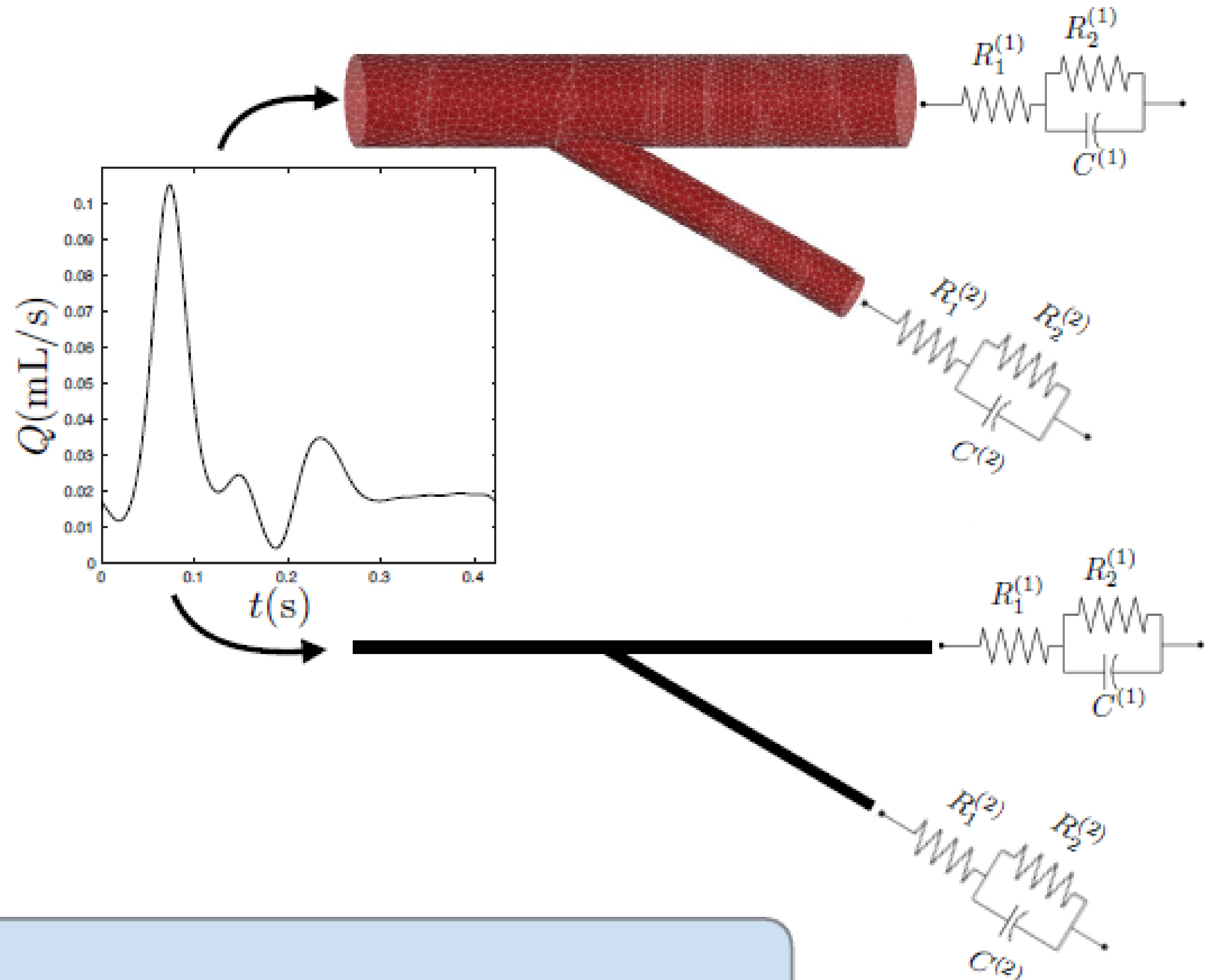
Goal:
Calibrate the outflow boundary condition parameters to match a target inlet systolic pressure, i.e.,

$$\mathbf{x}^\star = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmin}} \, |p_s^\star - p_s(\mathbf{x})|^2,$$

$$\mathbf{x} = [R_T^{(1)}, R_T^{(2)}]$$

$$\mathcal{X} = [10^{10}, 10^{11}] \times [10^{11}, 10^{12}]$$

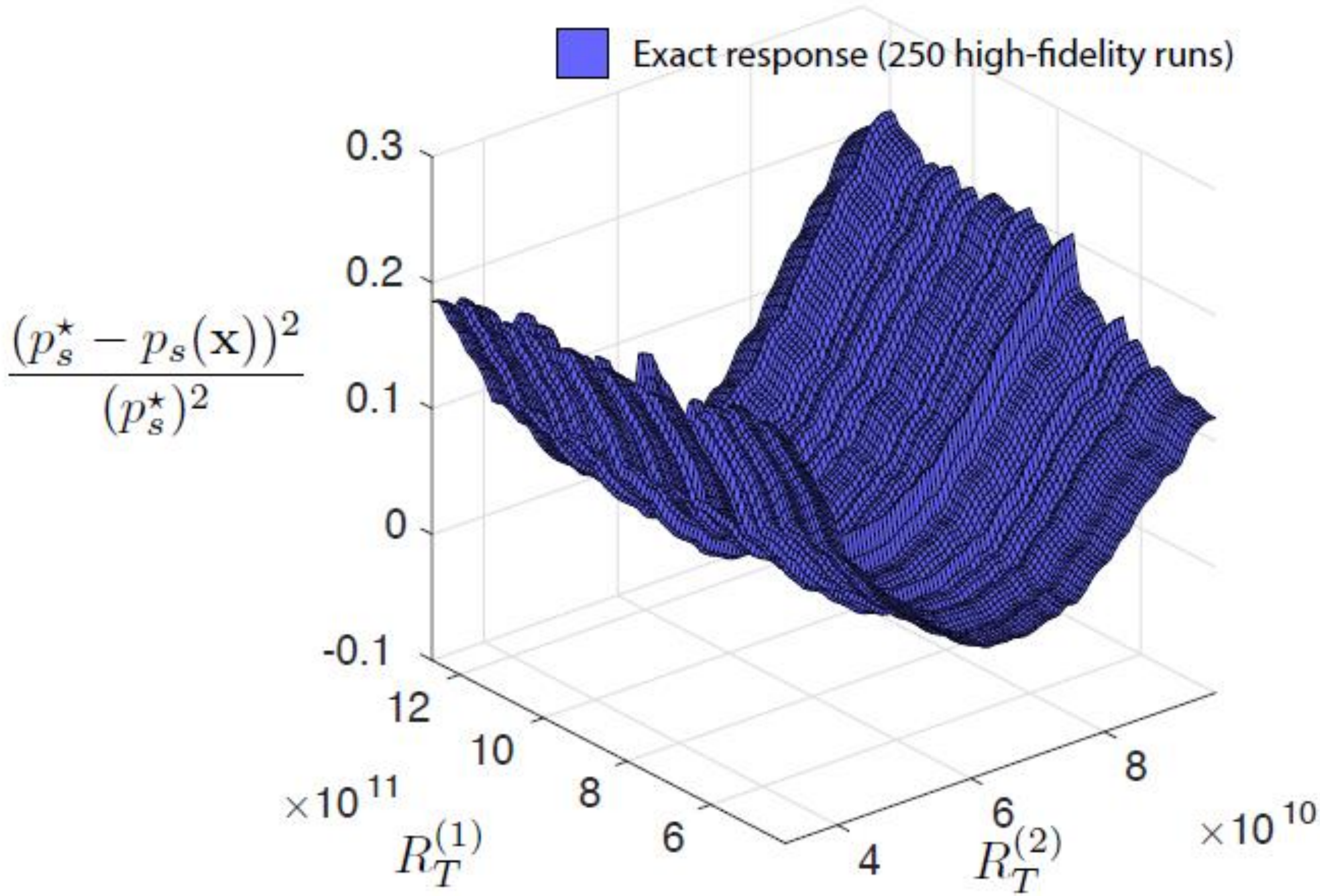
$$p_s^\star = 47\text{mmHg}$$



Multi-fidelity approach:
1.) 3D Navier-Stokes (spectral/hp elements, rigid artery) —> high fidelity O(hrs)
2.) Non-linear 1D-FSI (DG, compliant artery) —> intermediate fidelity O(mins)
3.) Linearized 1D-FSI solver around an inaccurate reference state —> low fidelity O(s)

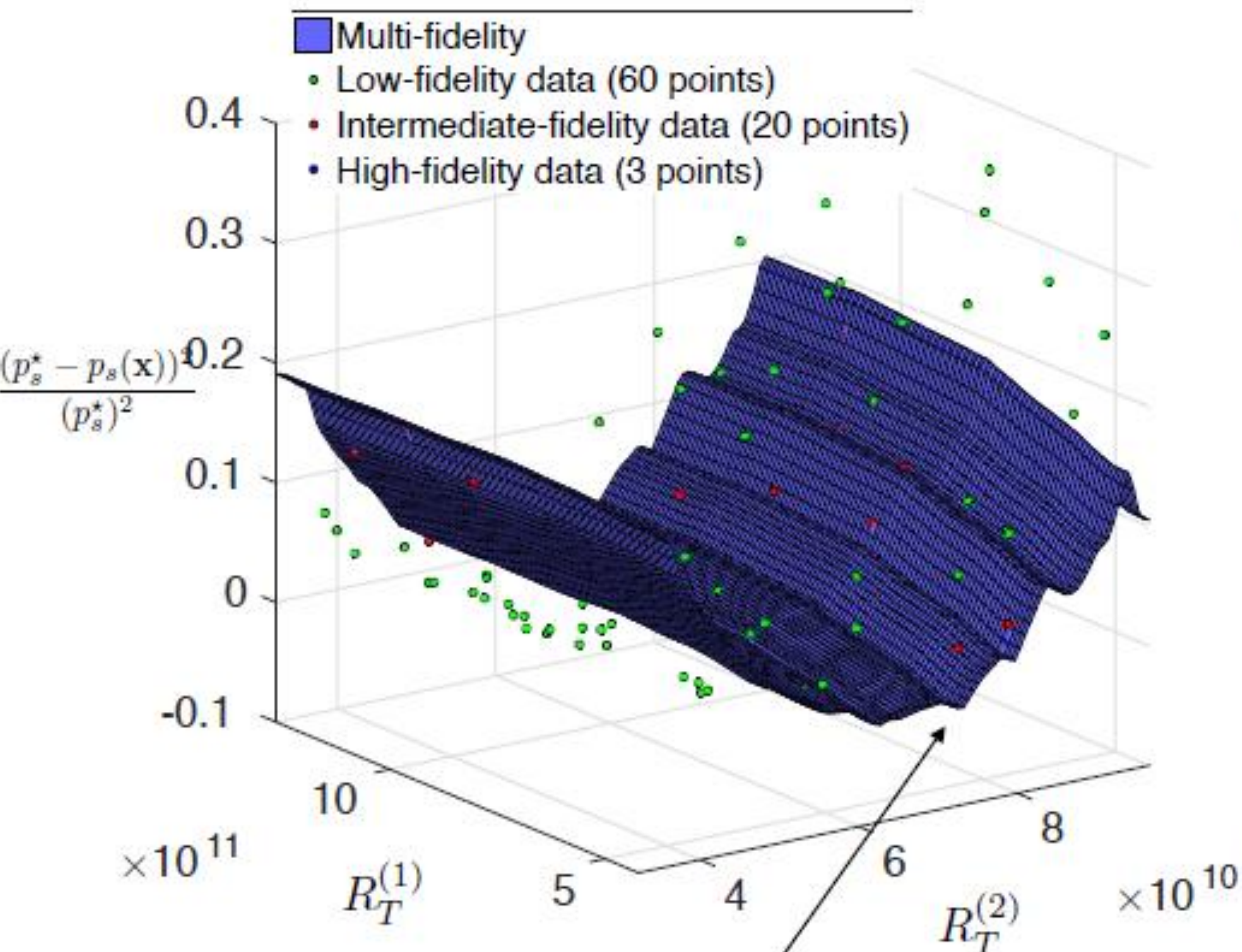
P. Perdikaris, and G.E Karniadakis. "Model inversion via multi-fidelity Bayesian optimization." J. R. Soc. Interface (2016)

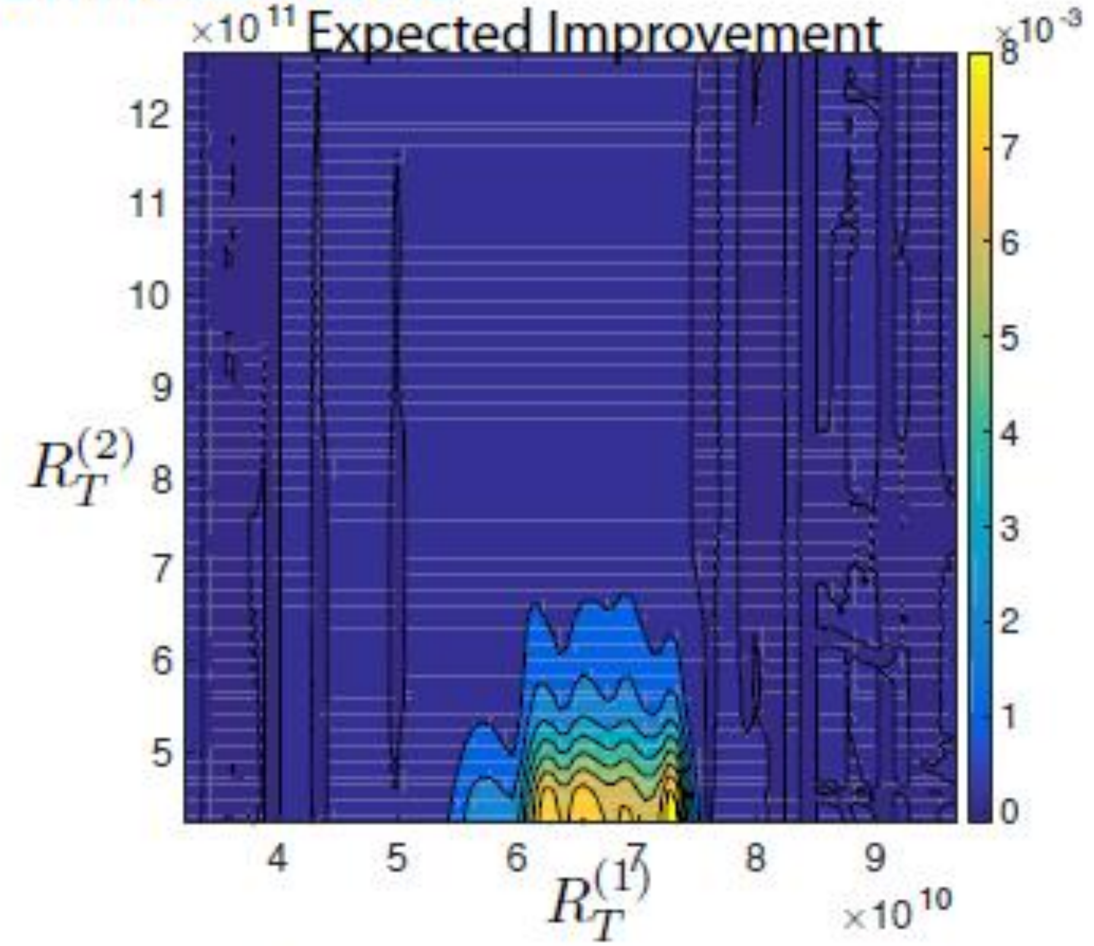
Calibration of blood flow simulations

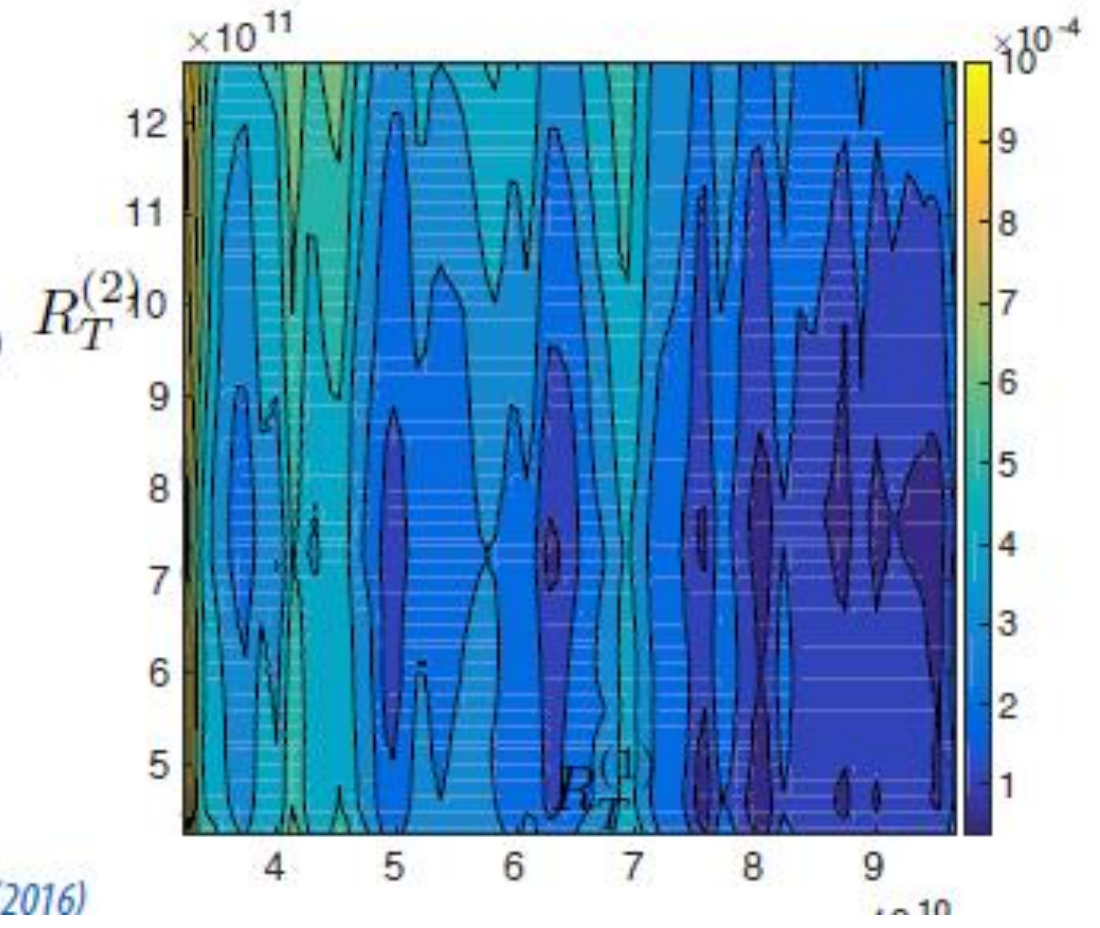


P. Perdikaris, and G.E Karniadakis. "Model inversion via multi-fidelity Bayesian optimization." J. R. Soc. Interface (2016)

Calibration of blood flow simulations







Decreased the relative error to $\mathcal{O}(10^{-3})$ after 3 iterations of BO, mainly sampling the lowest fidelity (cheapest) solver.

P. Perdikaris, and G.E Karniadakis, "Model inversion via multi-fidelity Bayesian optimization," J. R. Soc. Interface (2016)

Solving differential equations from measurements only

$$\mathcal{L}_x u_2(x) = f_2(x)$$

$$\frac{\partial}{\partial x} u_2(x) + \int_0^x u_2(\xi) d\xi = f(x)$$

$$\sum_{d=1}^{10} \frac{\partial^2}{\partial x_d^2} u(x) = f(x)$$

$$u_2(t,x) + \frac{\partial}{\partial x} u_2(t,x) - \frac{\partial^2}{\partial x^2} u_2(t,x) - u_2(t,x) = f(t,x)$$

$${}_{-\infty}D_x^{\alpha} u_2(x) - u_2(x) = f(x)$$

$$u_2(x) \sim \mathcal{GP}(0, g(x,x';\theta)) \xrightarrow{\textit{Linearity}} f_2(x) \sim \mathcal{GP}(0, k(x,x';\theta)) \longrightarrow k(x,x';\theta) = \mathcal{L}_x \mathcal{L}_{x'} g(x,x';\theta)$$

Problem setup:

- $f_2(x)$ is an unknown, black box function
- only scattered, noisy, variable fidelity observations of $f_2(x)$ are available
- we have no data on $u_2(x)$ other than the necessary initial/boundary conditions
- no numerical discretization!

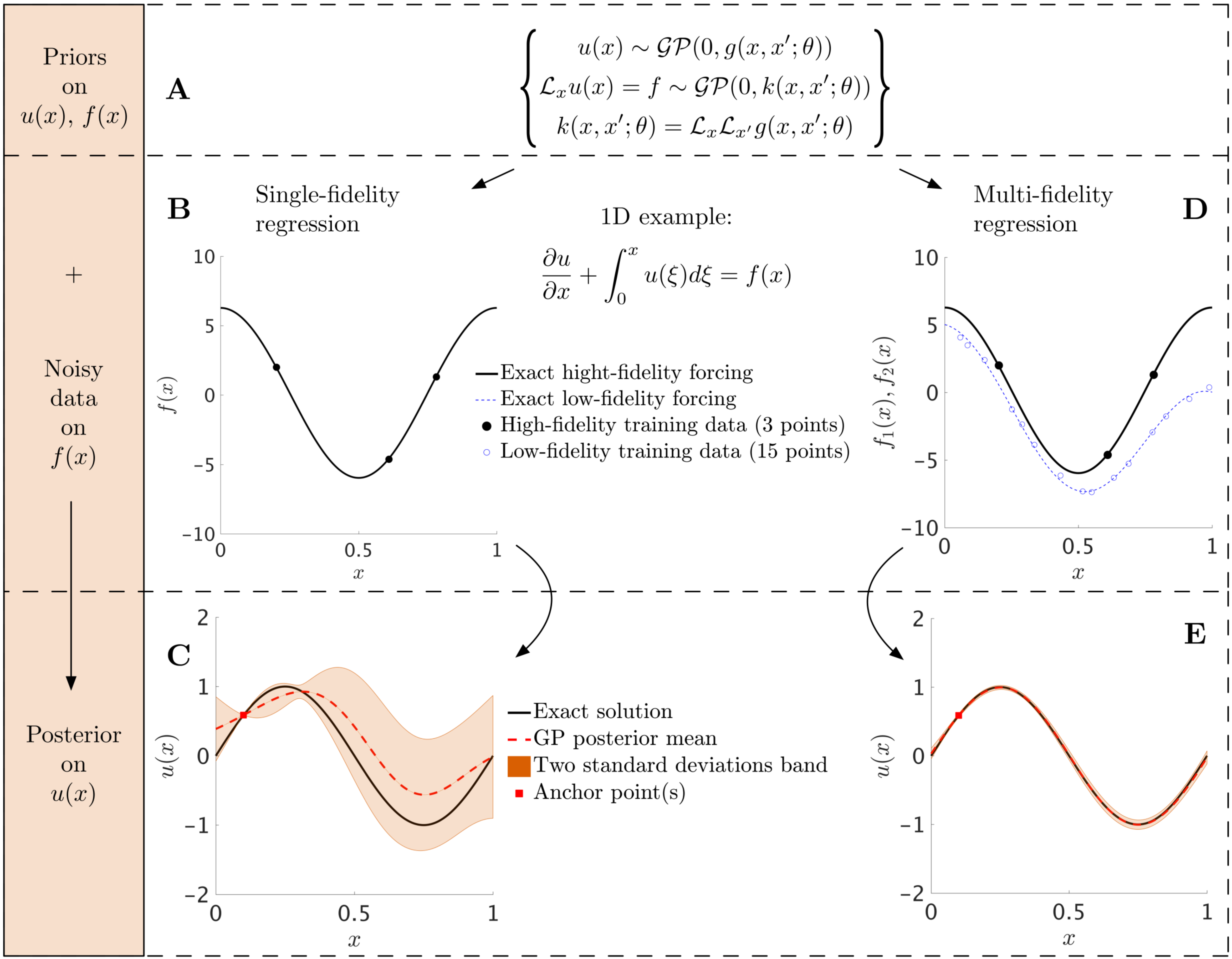
"once we accept that we don't know f, but we do know something, it becomes natural to take a Bayesian approach" P. Diaconis, "Bayesian numerical analysis", 1988

"stochastic methods will transform pure and applied mathematics in the beginning of the third millennium, as probability and statistics will come to be viewed as the natural tools to use in mathematical as well as scientific modeling" D. Mumford, "The dawning age of stochasticity", 2000

Revisiting numerical methods from a statistical inference viewpoint traces all the way back to the Poincar's courses on probability theory!

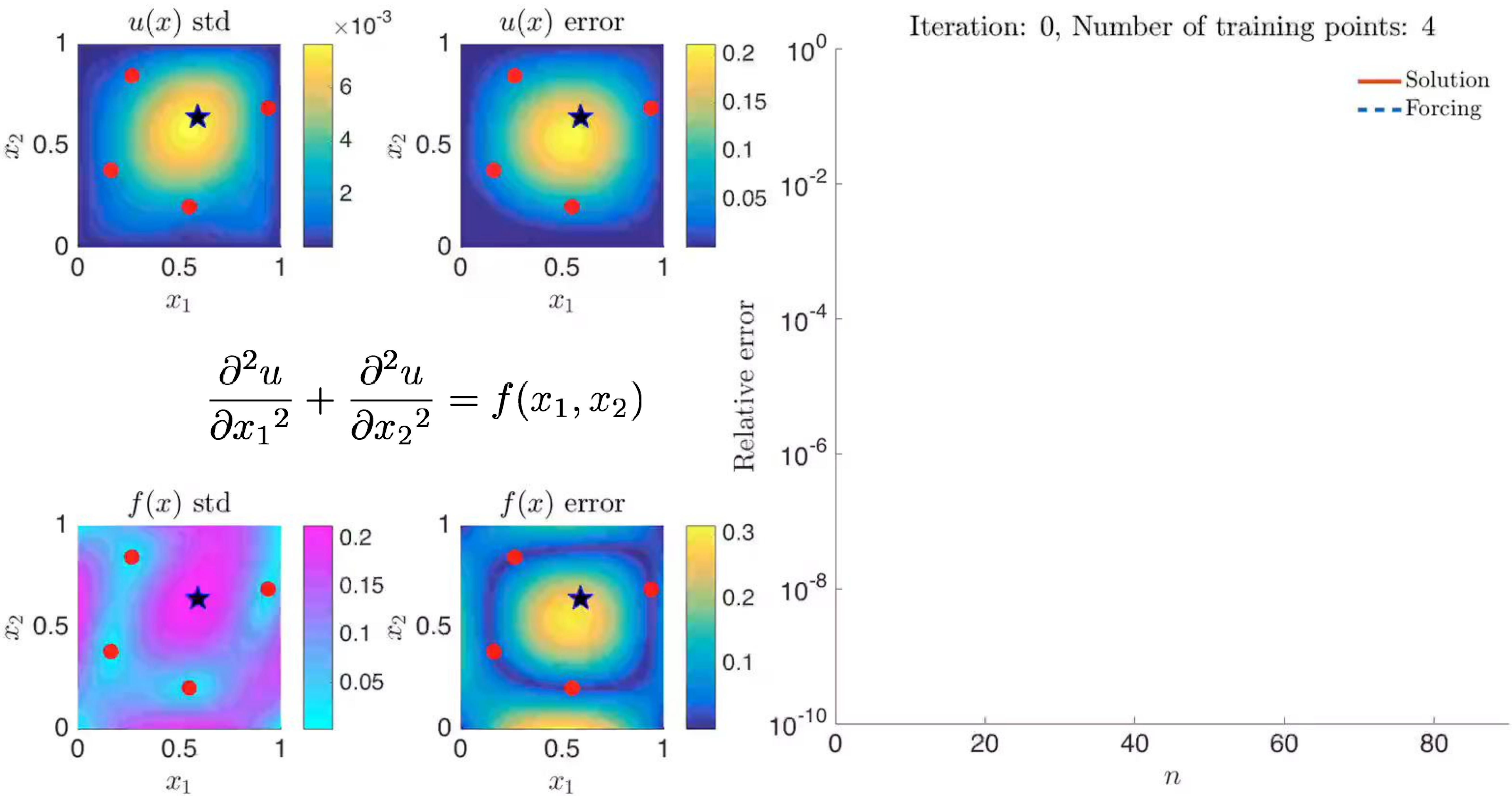
M. Raissi,., P. Perdikaris, and G.E. Karniadakis, Inferring solutions of differential equations using noisy multi-fidelity data, http://128.84.21.199/abs/1607.04805, 2016

Numerical solution of PDEs via machine learning



M. Raissi,., P. Perdikaris, and G.E. Karniadakis, Inferring solutions of differential equations using noisy multi-fidelity data, http:/

Adaptive refinement via active learning



• denotes the training data actively collected by the scheme.

★ denotes the next sampling point suggested by the active learning scheme.

M. Raissi,., P. Perdikaris, and G.E. Karniadakis, Inferring solutions of differential equations using noisy multi-fidelity data, http://128.84.21.199/abs/1607.04805, 20

Nonlinear equations via probabilistic time-stepping

Example: 1D viscous Burgers —> *The equation, along with the choice of a time-stepping scheme define a GP prior!*

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}, \quad x \in [-1,1], \quad t \in [0,1],$$
$$u(0,x) = u^0(x)$$
$$u(t,0) = u(t,1) = 0.$$

$$+ \quad \mathcal{L}_x u^{n+1}(x) := u^{n+1}(x) + \Delta t\ u^n(x)\ \frac{du^{n+1}(x)}{dx} - \Delta t\ \nu\frac{d^2u^{n+1}(x)}{dx^2} = u^n(x)$$

e.g., Backward Euler time-stepping

Let us start by making the assumption that

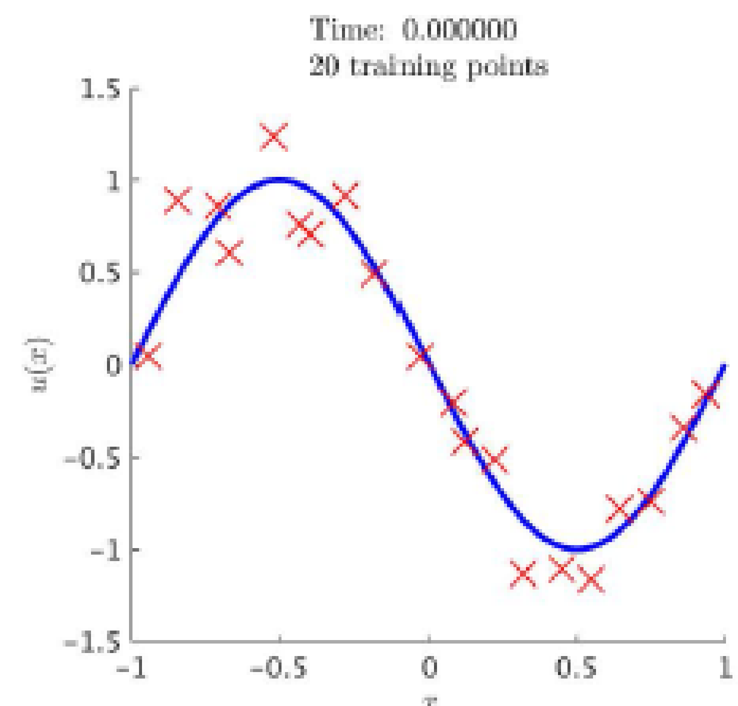
$$u^{n+1,n+1}(x) \sim \mathcal{GP}(0, k^{n+1,n+1}(x,x';\theta)).$$

Consequently,

$$u^n(x) \sim \mathcal{GP}(0, k^{n,n}(x,x';\theta)),$$

with the fundamental relationships

$$k^{n,n}(x,x';\theta) = \mathcal{L}_x\mathcal{L}_{x'}k^{n+1,n+1}(x,x';\theta),$$
$$k^{n+1,n}(x,x';\theta) = \mathcal{L}_{x'}k^{n+1,n+1}(x,x';\theta).$$



We can now train the hyper-parameters θ using the data $\{\boldsymbol{x}^n, \boldsymbol{u}^n\}$, $\{\boldsymbol{x}_b^{n+1}, \boldsymbol{u}_b^{n+1}\}$ and by minimizing the negative log marginal likelihood obtained from

$$\begin{bmatrix} \boldsymbol{u}_b^{n+1} \\ \boldsymbol{u}^n \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} k^{n+1,n+1}(\boldsymbol{x}_b^{n+1}, \boldsymbol{x}_b^{n+1}) & k^{n+1,n}(\boldsymbol{x}_b^{n+1}, \boldsymbol{x}^n) \\ k^{n,n+1}(\boldsymbol{x}^n, \boldsymbol{x}_b^{n+1}) & k^{n,n}(\boldsymbol{x}^n, \boldsymbol{x}^n) \end{bmatrix}\right).$$

Here, $\{\boldsymbol{x}^n, \boldsymbol{u}^n\}$ are *artificially generated* data and $\{\boldsymbol{x}_b^{n+1}, \boldsymbol{u}_b^{n+1}\}$ are data on the boundary.

Remark: Since $\{\boldsymbol{x}^n, \boldsymbol{u}^n\}$ are artificially generated data, in order to consistently propagate uncertainty in time we need to marginalize out $\boldsymbol{u}^n$.

M. Raissi,., P. Perdikaris, and G.E. Karniadakis.. Probabilistic time-stepping and uncertainty propagation for partial differential equations. (in preparation)

Scalability, non-stationarity & high dimensions

Scalability: GPs suffer from a cubic scaling with the data

✓ Low-rank approximations to the covariance
Snelson, E., and Z. Ghahramani. "Sparse Gaussian processes using pseudo-inputs."

✓ Frequency-domain learning algorithms
Perdikaris P., D. Venturi, G.E. Karniadakis "Multi-fidelity information fusion algorithms for high dimensional systems and massive data-sets", SIAM J. Sci. Comput., 2016

✓ Stochastic variational inference
Hensman, J., N. Fusi, and N.D. Lawrence. "Gaussian processes for big data."

Discontinuities and non-stationarity: GPs struggle with discontinuous data

✓ Use warping functions to transform into a jointly stationary input space

$$X \xrightarrow{f_1} H \xrightarrow{f_2} Y$$

- Log, sigmoid, betaCDF —> *"Warped GPs"* *Snelson, E., C.E. Rasmussen, and Z.Ghahramani. "Warped gaussian processes."*
- Neural networks —> *"Manifold GPs"* *Calandra, R., et al. "Manifold Gaussian processes for regression."*
- Gaussian processes —> *"Deep GPs"* *Damianou, A. C., and N.D. Lawrence. "Deep Gaussian processes."*

High-dimensions: Tensor product kernels suffer from the curse of dimensionality, i.e. the require an exponentially increasing amount of training data

✓ Data-driven additive kernels
Perdikaris P., D. Venturi, G.E. Karniadakis "Multi-fidelity information fusion algorithms for high dimensional systems and massive data-sets", SIAM J. Sci. Comput., 2016

✓ Unsupervised dimensionality-reduction (*GPLVM*, *deep auto-encoders*)
Lawrence, N.D. "Gaussian process latent variable models for visualisation of high dimensional data."

Summary & Future vision