Probabilistic Numerics

Khachik Sargsyan

Sandia National Laboratories

Data Science Reading Group
February 9, 2017

Main reference [www.probabilistic-numerics.org]

PROCEEDINGS A

rspa.royalsocietypublishing.org

Research

Article submitted to journal

Subject Areas:
Statistics, Computational Mathematics, Artificial Intelligence

Keywords:
numerical methods, probability, inference, statistics

Probabilistic Numerics and Uncertainty in Computations

Philipp Hennig[1], Michael A Osborne[2] and Mark Girolami[3]

[1]Max Planck Institute for Intelligent Systems, Tübingen, Germany
[2]University of Oxford, United Kingdom
[3]University of Warwick, United Kingdom

We deliver a call to arms for *probabilistic numerical methods*: algorithms for numerical tasks, including linear algebra, integration, optimization and solving differential equations, that return uncertainties in their calculations. Such uncertainties, arising from the loss of precision induced by numerical calculation with limited time or hardware, are important for much contemporary science and industry. Within

- Hennig, Philipp, Michael A. Osborne, and Mark Girolami. "Probabilistic numerics and uncertainty in computations." Proc. R. Soc. A. Vol. 471. No. 2179. The Royal Society, 2015.

Uncertainty in computations

- *Uncertainty quantification:* usually ill-posed problems, epistemic uncertainty set-up of computation
- *Stochastic numerical methods:* aleatoric (inherent) randomness, repeated computations
- ***Probabilistic numerics:*** well-posed deterministic problems, turned into a *learning* problem
  - Estimate uncertainty due to the numerical method itself
    - Integration
    - Optimization
    - Linear solvers
    - PDE/ODE
    - . . .

Probabilistic numerics, handy if...

- one needs to propagate through some computational pipeline. E.g., machine learning methods are simply chains of
  - linear algebra (least-sq.)
  - optimization (fitting)
  - integration (MCMC, nuisance parameters)
  - diff. eq. (control)
- one has noisy function evaluations, or a built-in UQ problem, e.g. uncertain input parameters
- interested in accuracy-vs-time tradeoff... what if we stop short in any computational endeavor?

Background: Bayesian paradigm

Bayesian paradigm:

- Quantify the unknown with probability distribution
- *Data model:* $d \approx f(m)$
- Bayes formula

$$\underbrace{p(m|d)}_{\text{Posterior}} = \frac{\overbrace{p(d|m)}^{\text{Likelihood}} \overbrace{p(m)}^{\text{Prior}}}{\underbrace{p(d)}_{\text{Evidence}}}$$

Ingredients:

- Prior: knowledge of m prior to data
- Likelihood: fit forward model to data; measurement noise
- Posterior: combines information from prior and data
- Evidence: normalizing constant; useful for model selection

Background: Bayesian paradigm

- Bayes formula

$$\underbrace{p(m|d)}_{\text{Posterior}} = \frac{\overbrace{p(d|m)}^{\text{Likelihood}}\ \overbrace{p(m)}^{\text{Prior}}}{\underbrace{p(d)}_{\text{Evidence}}}$$

Allows:

- Flexible way of combining prior knowledge and data
- Dealing with heterogeneous sources of uncertainty
- Sequential setting
- Quantifying lack-of-knowledge (information content)

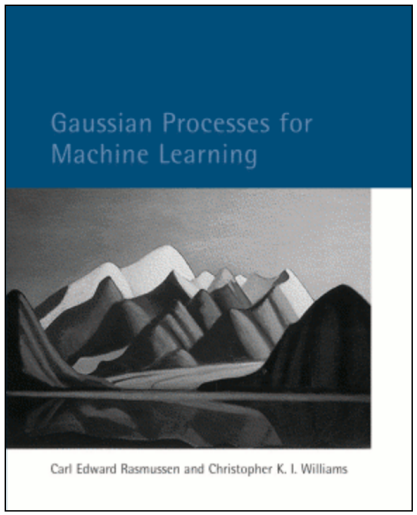
Background: Gaussian Processes (GP)

- Extension of normal r.v. to stochastic processes; Scalar $\Rightarrow$ Function
- $f(x)$ is a GP $\iff$ $f(x_1), f(x_2), \ldots, f(x_n)$ is multivariate normal r.v. for any $\{x_i\}_{i=1}^n$
- A GP is defined by its mean function $\mu(x)$ and covariance function $C(x, x')$
- A good resource is www.gaussianprocess.org/
- ... even better, the Rasmussen & Williams book

www.gaussianprocess.org/gpml/

Gaussian Processes for Machine Learning

Carl Edward Rasmussen and Christopher K. I. Williams
The MIT Press, 2006. ISBN 0-262-18253-X.

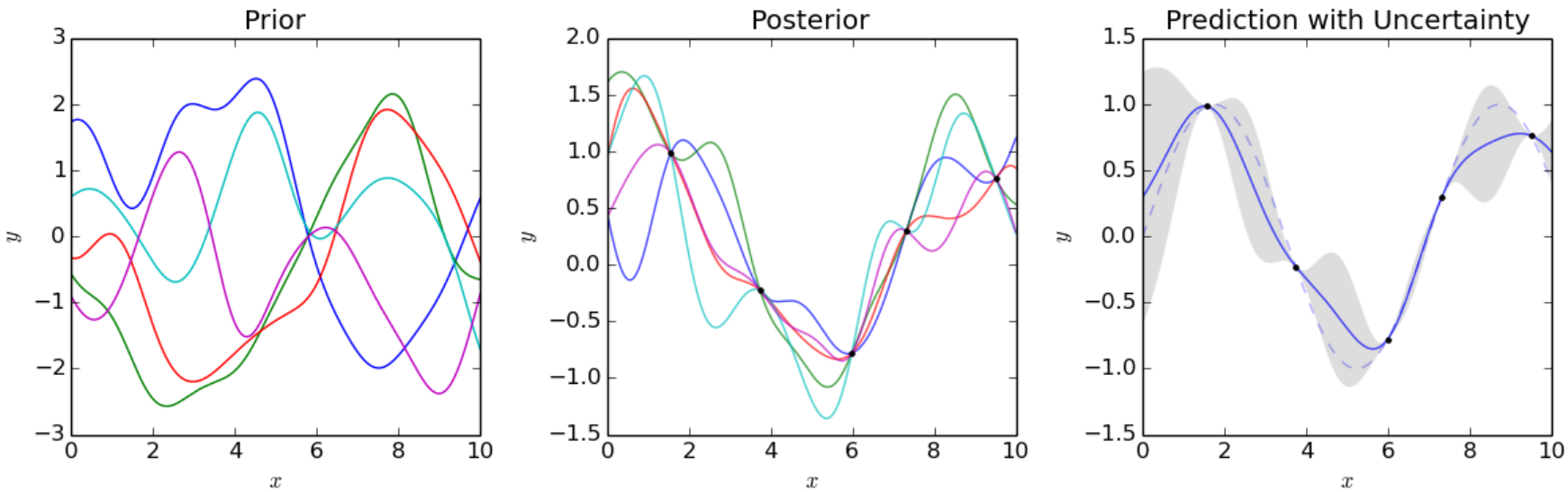


[Contents | Software | Datasets | Errata | Authors | Order]

Background: Gaussian Processes (GP)

- Extension of normal r.v. to stochastic processes; Scalar $\Rightarrow$ Function
- $f(x)$ is a GP $\iff$ $f(x_1), f(x_2), \ldots, f(x_n)$ is multivariate normal r.v. for any $\{x_i\}_{i=1}^n$
- A GP is defined by its mean function $\mu(x)$ and covariance function $C(x, x')$
- GP regression, or kriging, is a handy tool for surrogate model construction
- Can be posed in a Bayesian context, i.e. given $f_i = f(x_i)$ for $i = 1, \ldots, N$, *learn* mean and covariance functions, or point value at any x^*

$$\overbrace{p(f(x^*)|\{f(x_i\}_{i=1}^N)}^{\text{Posterior}} \propto \overbrace{p(\{f(x_i\}_{i=1}^N|f(x^*))}^{\text{Likelihood (Lin. Alg.)}} \overbrace{p(f(x^*))}^{\text{Prior}} \tag{1}$$



Probabilistic numerics

Deep connections

- Gaussian quadrature $\Leftrightarrow$ GP regression
- Conj. gradients $\Leftrightarrow$ Gaussian conditioning
- BFGS $\Leftrightarrow$ autoregressive filtering
- Runge-Kutta $\Leftrightarrow$ Gauss-Markov extrapolation

Problem class	integration	linear opt.	nonlinear opt.	ODE IVPs
inferred z	$z = f; \int f(x)\,dx$	$z = A^{-1}; Ax = b$	$z = B = \nabla\nabla^\intercal f$	$z'(t) = f(z(t), t)$
classic method	Gaussian quad.	conjugate gradients	BFGS	Runge-Kutta
$p(z)$	$\mathcal{GP}(f; \mu, k)$	$\mathcal{N}(A^{-1}; M, V)$	$\mathcal{GP}(z; \mu, k)$	$\mathcal{GP}(z; \mu, k)$
$p(\boldsymbol{y} \mid z)$	$\mathbb{I}(f(x_i) = y_i)$	$\mathbb{I}(y_i = Ax_i)$	$\mathbb{I}(y_i = Bx_i)$	$\mathbb{I}(y_i = z'(t))$
decision rule	minimize post. variance	gradient at est. solution	gradient under est. Hessian	evaluate at est. solution

Table 1. Probabilistic description of several basic numerical problems (shortened notation for brevity). In quadrature, (symmetric positive definite) linear optimization, non-linear optimization, and the solution of ordinary differential equation initial value problems, classic methods can be cast as maximum a-posteriori estimation under Gaussian priors. In each case, the likelihood function is a strict conditioning, because observations are assumed to be noise-free. Because numerical methods are active (they decide which computations to perform), they require a decision rule. This is often "greedy": evaluation under the posterior mean estimate. The exception is integration, which is the only area where the estimated solution of the numerical task is not required to construct the next evaluation.

General Recipe

(a) General Recipe for Probabilistic Numerical Algorithms

These recent results, identifying probabilistic formulations for classic numerical methods, highlight a general structure. Consider the problem of approximating the intractable variable z, if the algorithm has the ability to choose 'inputs' $\boldsymbol{x} = \{x_i\}_{i=1,\dots}$ for computations that result in numbers $\boldsymbol{y}(\boldsymbol{x}) = \{y_i(x_i)\}_{i=1,\dots}$. A blueprint for the definition of probabilistic numerical methods requires two main ingredients:

(i) A *generative model* $p(z, \boldsymbol{y}(\boldsymbol{x}))$ for all variables involved—that is, a joint probability measure over the intractable quantity to be computed, and the tractable numerical quantities computed in the process of the algorithm. Like all (sufficiently structured) probability measures, this joint measure can be written as

$$p\big(z, \boldsymbol{y}(\boldsymbol{x})\big) = p(z)\, p\big(\boldsymbol{y}(\boldsymbol{x})\big|z\big), \tag{3.1}$$

i.e. separated into a *prior* $p(z)$ and a *likelihood* $p\big(\boldsymbol{y}(\boldsymbol{x})\big|z\big)$. The prior encodes a hypothesis class over solutions, and assigns a typically non-uniform measure over this class. The likelihood explains how the collected tractable numbers $\boldsymbol{y}$ relate to z. It has the basic role of describing the numerical task. Often, in classic numerical problems, the likelihood is a deterministic conditioning rule, a point measure.

(ii) A *design*, *action* rule, or *policy* r, such that

$$x_{i+1} = r\Big(p\big(z, \boldsymbol{y}(\boldsymbol{x})\big), \boldsymbol{x}_{1:i}, \boldsymbol{y}_{1:i}\Big), \tag{3.2}$$

encoding how the algorithm should collect numbers. (Here $\boldsymbol{x}_{1:i}$ should be understood as the actions taken in the preceding steps 1 to i, and similarly for $\boldsymbol{y}_{1:i}$). This rule can be simple, for example it could be independent of collected data (regular grids for integration). Or it might have a Markov-type property that the decision at i only depends on $k < i$ previous decisions (for example in ODE solvers). Sometimes, these rules can be shown to be associated with the minimization of some empirical loss function, and thus be given a decision-theoretic motivation. This is for example the case for regular grids in quadrature rules [29].

Integration

- One of the first areas in probabilistic numerics
  - O'Hagan, Anthony. "Bayes-Hermite Quadrature" Journal of statistical planning and inference 29.3 (1991): 245-260.
  - O'Hagan, Anthony. "Monte Carlo is fundamentally unsound." The Statistician (1987): 247-249.
  - Rasmussen, Carl Edward, and Zoubin Ghahramani. "Bayesian Monte Carlo." Advances in neural information processing systems (2003): 505-512.
- Basically, rewrite $I[f] = \int f(x)dx$ as a Bayesian problem
  $p(I|\{f_i = f(x_i)\}_{i=1}^n) \propto p(f_i|I)p(f_i)$
- Not many theoretical guarantees; posterior shrinkage estimates in
  - Briol, Francois-Xavier, et al. "Probabilistic Integration: A Role for Statisticians in Numerical Analysis?." arXiv preprint arXiv:1512.00933 (2015).
  - Briol, Francois-Xavier, et al. "Frank-Wolfe Bayesian quadrature: Probabilistic integration with theoretical guarantees." Advances in Neural Information Processing Systems. 2015.
- Can help with adaptive sample selection
  - Shaw, J. E. H. "A quasirandom approach to integration in Bayesian statistics." The Annals of Statistics (1988): 895-914.

Integration: a great read

The Statistician (1987) 36, pp. 247–249

Monte Carlo is fundamentally unsound

A. O'HAGAN

Department of Statistics, University of Warwick, Coventry CV4 7AL, U.K.

Abstract. We present some fundamental objections to the Monte Carlo method of numerical integration.

Two major objections:

- Same sample set can lead to different integral values
- No use of information of the sample set

Integration: visual

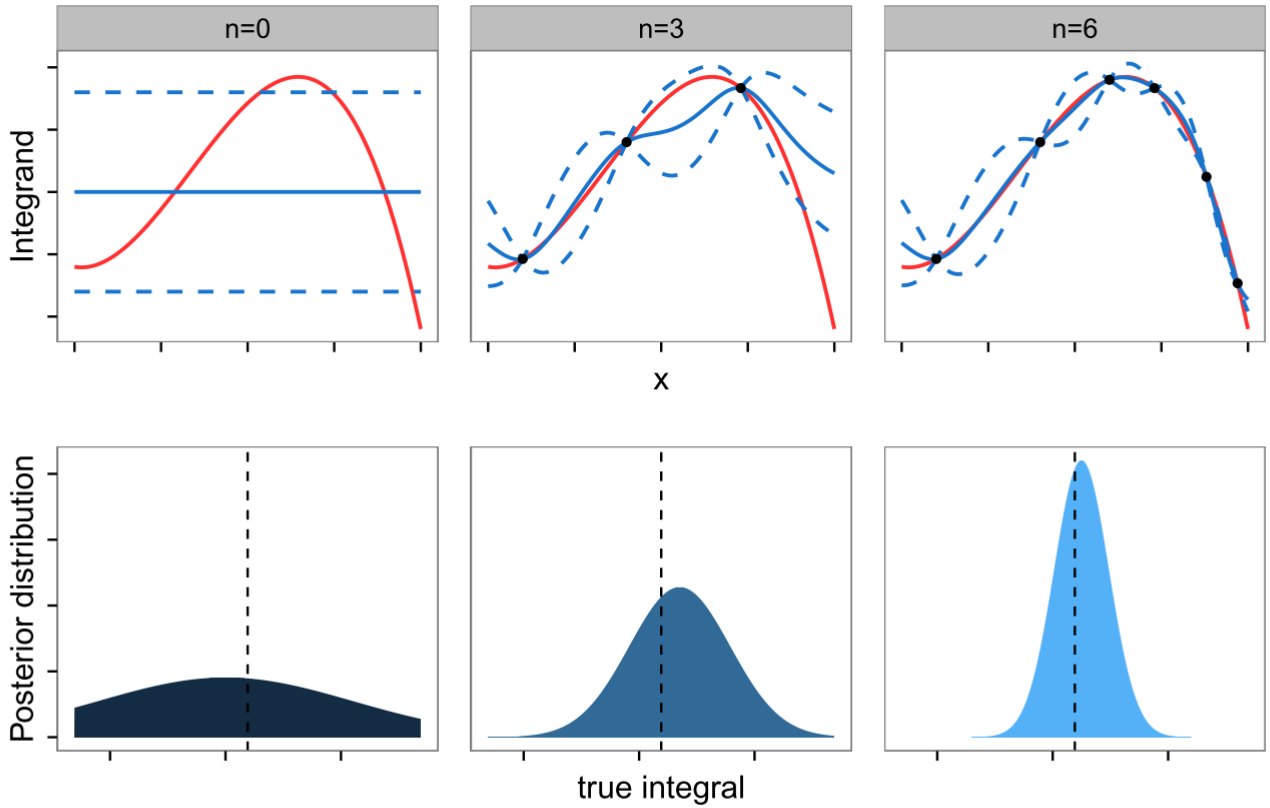


Figure 1: Sketch of Bayesian Quadrature. The top row shows the approximation of the integrand f (in red) by the GP posterior mean m_1 (in blue) as the number n of function evaluations is increased. The dashed lines represent 95% credible intervals. The bottom row shows the Gaussian distribution with mean $\mathbb{E}[\Pi[f]|\mathcal{D}]$ and variance $\mathbb{V}[\Pi[f]|\mathcal{D}]$ that models our uncertainty over the solution of the integral as n increases (the dashed black line gives the true value of the integral). When $n = 0$, the approximation of the integral is fully specified by the GP prior. As the number of states n increases, the approximation of f becomes more precise and the Gaussian posterior distribution contracts onto the true value of the integral.

Integration: $I = \int_0^1 f(x)dx$

- Put a prior on $f(x)$
- Compute $f(x_i)$ for $i = 1, \ldots, n$
- Compute the posterior via Bayes rule (first f then I)

- Brownian motion prior $\Rightarrow$ Posterior mean is piecewise linear interpolation $\Rightarrow$ Trapezoidal rule
- Prior is k-th integral of Brownian motion $\Rightarrow$ Posterior mean is spline of order $2k+1$ $\Rightarrow$ Higher order integration rules

Linear Alg./Optimization

$Ax = b$

- The goal is to replace the point estimates returned by existing methods with a Gaussian posterior belief over the elements of the inverse of A, which can be used to estimate errors.
- Put a Gaussian prior on $H = A^{-1}$, then estimate the posterior action rule $x_{i+1} = x_i - \alpha H_i(Ax_i - b)$ induced by posterior of H_i.
- Hennig, Philipp. "Probabilistic interpretation of linear solvers." SIAM Journal on Optimization 25.1 (2015): 234-260.
- Relation to Bayesian optimization
  - This is *not* simply max posterior
  - To find the next candidate point, a posterior of utility function is evaluated.

Differential equations

... e.g. IVP ODEs $\frac{dx}{dt} = f(x,t)$

- Runge-Kutta (RK): linear extrapolation rule
- Repeatedly construct "estimates" of $\hat{x}_i \approx x(t_i)$ which is then used to collect an "observation" $y_i = f(\hat{x}_i, t_i)$, s.t. $\hat{x}_i = x_0 + \sum_{j<i} w_{ij} y_j$
- Relation to Bayesian integration

- Barber, David. "On solving Ordinary Differential Equations using Gaussian Processes." arXiv preprint arXiv:1408.3807 (2014).

Differential equations

PDE mesh discretization error

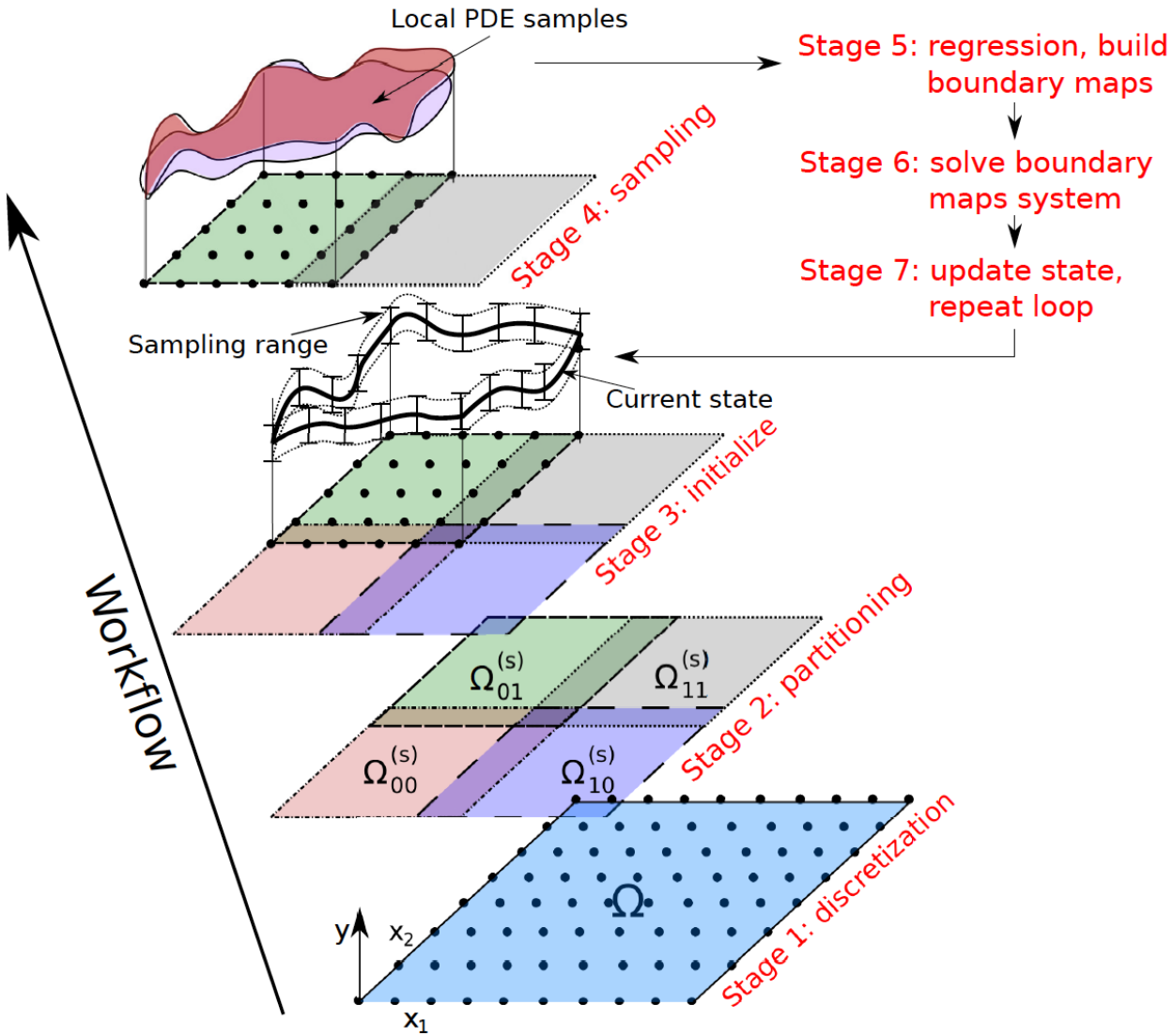
- Conrad, Patrick R., et al. ”Probability measures for numerical solutions of differential equations.” arXiv preprint arXiv:1506.04592 (2015).
- Chkrebtii, Oksana A., et al. ”Bayesian solution uncertainty quantification for differential equations.” Bayesian Analysis 11.4 (2016): 1239-1267.

Resilient solver [Shameless plug]

- K. Sargsyan, et al. ”Fault resilient domain decomposition preconditioner for PDEs.” SIAM Journal on Scientific Computing 37.5 (2015): A2317-A2345.
- F. Rizzi, et al. ”Partial differential equations preconditioner resilient to soft and hard faults.” Cluster Computing (CLUSTER), 2015 IEEE International Conference on. IEEE, 2015.

- Probabilistic preconditioner, or a domain decomposition solver
- PDE solution is taken as state-of-knowledge
- Learn boundary-to-boundary maps to update the solution state
- Targeting resilience: Bayesian inference allows optimal regression in presence of outliers

Rexsss: Resilient EXtreme Scale Scientific Simulations



Literature

- Kadane, Joseph B. ”Parallel and sequential computation: a statistician’s view.” Journal of Complexity 1.2 (1985): 256-263.
- Diaconis, Persi. “Bayesian numerical analysis.” Statistical decision theory and related topics IV 1 (1988): 163-175.
- O’Hagan, Anthony. “Some Bayesian numerical analysis.” Bayesian statistics 4.345-363 (1992): 4-2.
- Hennig, Philipp, and Martin Kiefel. ”Quasi-Newton methods: a new direction.” arXiv preprint arXiv:1206.4602 (2012).
- Kac, Mark. ”On distributions of certain Wiener functionals.” Transactions of the American Mathematical Society 65.1 (1949): 1-13.
- Bui-Thanh, Tan, and Mark Girolami. ”Solving large-scale PDE-constrained Bayesian inverse problems with Riemann manifold Hamiltonian Monte Carlo.” Inverse Problems 30.11 (2014): 114014.
- Minka, Thomas P. ”Deriving quadrature rules from Gaussian processes.” Statistics Department, Carnegie Mellon University, Tech. Rep (2000).
- Tan, Zhiqiang. ”On a likelihood approach for Monte Carlo integration.” Journal of the American Statistical Association 99.468 (2004): 1027-1036.
- Oates, Chris, FranÃ§ois-Xavier Briol, and Mark Girolami. ”Probabilistic Integration and Intractable Distributions.” arXiv preprint arXiv:1606.06841 (2016).
- Bartels, Simon, Philipp Hennig, and TÃ $\frac{1}{4}$ bingen Germany. ”Probabilistic Approximate Least-Squares.” Proceedings of the 19th International Conference on Artificial Intelligence and Statistics. 2016.