NON-LINEAR APPROXIMATION OF BAYESIAN UPDATE

Alexander Litvinenko\textsuperscript{1}, Hermann G. Matthies\textsuperscript{2}, Elmar Zander\textsuperscript{2}

\textsuperscript{1}Extreme Computing Research Center, KAUST,
\textsuperscript{2}Institute of Scientific Computing, TU Braunschweig, Brunswick, Germany

http://sri-uq.kaust.edu.sa/
Figure: KAUST campus, 7 years old, approx. 7000 people (include 1700 kids), 100 nations, 36 km².
Stochastic Numerics Group at KAUST
Advances in Uncertainty Quantification Methods, Algorithms and Applications (UQAW 2015)
January 6 – 9, 2015
9:00 a.m. – 5:00 p.m.
Level 0 auditorium, between Al-Jazri and Al-Kindi (buildings 4 and 5)

WORKSHOP TOPICS
1. Uncertainty Quantification Methods and Algorithms
2. Verification and Validation
3. Experimental Design
4. Applications to Problems in Computational Science, Engineering, Networks and the Environment

ORGANIZERS
RAUL TEMPORE
Director, Center for Uncertainty Quantification in Computational Science & Engineering, Associate Professor, Computer, Electrical, and Mathematical Sciences and Engineering Division, KAUST

OMAR KNIO
Deputy Director, Center for Uncertainty Quantification in Computational Science & Engineering, Professor, Computer, Electrical and Mathematical Sciences and Engineering Division, KAUST

For more information contact:
http://sri-qa.kaust.edu.sa
Raul.Tempone@kaust.edu.sa; Omar.Knio@kaust.edu.sa
Et. Alexander.Livinenco@kaust.edu.sa
1. Computing the full Bayesian update is very expensive (MCMC is expensive)
2. Look for a cheap surrogate (linear, quadratic, cubic,... approx.)
3. Kalman filter is a particular case
4. Do Bayesian update of Polynomial Chaos Coefficients! (not probability densities!)
5. Consider non-Gaussian cases


General idea:

We observe / measure a system, whose structure we know in principle.

The system behaviour depends on some quantities (parameters),

which we do not know ⇒ uncertainty.

We model (uncertainty in) our knowledge in a Bayesian setting:

as a probability distribution on the parameters.

We start with what we know a priori, then perform a measurement.

This gives new information, to update our knowledge (identification).

Update in probabilistic setting works with conditional probabilities

⇒ Bayes’s theorem.

Repeated measurements lead to better identification.
Consider
\[ A(u; q) = f \implies u = S(f; q), \]
where \( S \) is solution operator.
Operator depends on parameters \( q \in Q \),
hence state \( u \in \mathcal{U} \) is also function of \( q \):
Measurement operator \( Y \) with values in \( \mathcal{Y} \):
\[ y = Y(q; u) = Y(q, S(f; q)). \]
Examples of measurements:
\[ y(\omega) = \int_{D_0} u(\omega, x) \, dx, \text{ or } u \text{ in few points} \]
For given $f$, measurement $y$ is just a function of $q$. This function is usually not invertible $\Rightarrow$ ill-posed problem, measurement $y$ does not contain enough information.

In Bayesian framework state of knowledge modelled in a probabilistic way, parameters $q$ are uncertain, and assumed as random. Bayesian setting allows updating / sharpening of information about $q$ when measurement is performed. The problem of updating distribution—state of knowledge of $q$ becomes well-posed.

Can be applied successively, each new measurement $y$ and forcing $f$ —may also be uncertain—will provide new information.
Conditional probability and expectation

With state $u$ a RV, the quantity to be measured

$$y(\omega) = Y(q(\omega), u(\omega))$$

is also uncertain, a random variable. 

Noisy data: $\hat{y} + \epsilon(\omega)$, where $\hat{y}$ is the “true” value and a random error $\epsilon$.

Forecast of the measurement: $z(\omega) = y(\omega) + \epsilon(\omega)$.

Classically, Bayes’s theorem gives conditional probability

$$P(I_q|M_z) = \frac{P(M_z|I_q)}{P(M_z)} P(I_q) \quad \text{(or } \pi_q(q|z) = \frac{p(z|q)}{Z_s} p_q(q))$$

expectation with this posterior measure is conditional expectation. Kolmogorov starts from conditional expectation

$$\mathbb{E}(\cdot | M_z),$$

from this conditional probability via $P(I_q|M_z) = \mathbb{E} (\chi_{I_q} | M_z)$. 

The conditional expectation is defined as orthogonal projection onto the closed subspace $L_2(\Omega, \mathbb{P}, \sigma(z))$:

$$\mathbb{E}(q|\sigma(z)) := P_{\mathcal{Z}_\infty} q = \arg\min_{\tilde{q} \in L_2(\Omega, \mathbb{P}, \sigma(z))} \| q - \tilde{q} \|_{L_2}^2$$

The subspace $\mathcal{Z}_\infty := L_2(\Omega, \mathbb{P}, \sigma(z))$ represents the available information.

The update, also called the assimilated value $q_a(\omega) := P_{\mathcal{Z}_\infty} q = \mathbb{E}(q|\sigma(z))$, and represents new state of knowledge after the measurement. 

Doob-Dynkin: $\mathcal{Z}_\infty = \{ \varphi \in \mathcal{Z} : \varphi = \phi \circ z, \phi \text{ measurable} \}$. 
Multivariate Hermite polynomials were used to approximate random fields/stochastic processes with Gaussian random variables. According to Cameron and Martin theorem PCE expansion converges in the $L_2$ sense.

Let $Y(x, \theta)$, $\theta = (\theta_1, \ldots, \theta_M, \ldots)$, is approximated:

$$Y(x, \theta) = \sum_{\beta \in J_{m,p}} H_\beta(\theta) Y_\beta(x), \quad |J_{m,p}| = \frac{(m + p)!}{m!p!},$$

$$H_\beta(\theta) = \prod_{k=1}^{M} h_{\beta_k}(\theta_k),$$

$$Y_\beta(x) = \frac{1}{\beta!} \int_{\Theta} H_\beta(\theta) Y(x, \theta) \mathbb{P}(d\theta).$$

$$Y_\beta(x) \approx \frac{1}{\beta!} \sum_{i=1}^{N_q} H_\beta(\theta_i) Y(x, \theta_i) w_i.$$
Look for $\varphi$ such that $q(\xi) = \varphi(z(\xi))$, $z(\xi) = y(\xi) + \varepsilon(\omega)$:

$$\varphi \approx \tilde{\varphi} = \sum_{\alpha \in J_p} \varphi_\alpha \Phi_\alpha(z(\xi)) \quad (1)$$

and minimize $\|q(\xi) - \tilde{\varphi}(z(\xi))\|^2$, where $\Phi_\alpha$ are polynomials (e.g. Hermite, Laguerre, Chebyshev or something else). Taking derivatives with respect to $\varphi_\alpha$:

$$\frac{\partial}{\partial \varphi_\alpha} \langle q(\xi) - \tilde{\varphi}(z(\xi)), q(\xi) - \tilde{\varphi}(z(\xi)) \rangle = 0 \quad \forall \alpha \in J_p \quad (2)$$

Inserting representation for $\tilde{\varphi}$, obtain
Numerical computation of NLBU

\[
\frac{\partial}{\partial \varphi_\alpha} \mathbb{E} \left( q^2(\xi) - 2 \sum_{\beta \in J} q \varphi_\beta \Phi_\beta(z) + \sum_{\beta, \gamma \in J} \varphi_\beta \varphi_\gamma \Phi_\beta(z) \Phi_\gamma(z) \right)
\]

\[
= 2 \mathbb{E} \left( -q \Phi_\alpha(z) + \sum_{\beta \in J} \varphi_\beta \Phi_\beta(z) \Phi_\alpha(z) \right)
\]

\[
= 2 \left( \sum_{\beta \in J} \mathbb{E} \left[ \Phi_\beta(z) \Phi_\alpha(z) \right] \varphi_\beta - \mathbb{E} \left[ q \Phi_\alpha(z) \right] \right) = 0 \quad \forall \alpha \in J
\]

\[
\mathbb{E} \left[ \Phi_\beta(z) \Phi_\alpha(z) \right] \varphi_\beta = \mathbb{E} \left[ q \Phi_\alpha(z) \right]
\]
Now, rewriting the last sum in a matrix form, obtain the linear system of equations ($= A$) to compute coefficients $\varphi_\beta$:

$$
\begin{pmatrix}
\vdots & \ddots & \ddots & \ddots \\
\vdots & \mathbb{E} [\Phi_\alpha(z(\xi))\Phi_\beta(z(\xi))] & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots \\
\mathbb{E} [q(\xi)\Phi_\alpha(z(\xi))] & \ddots & \ddots & \vdots
\end{pmatrix}
\begin{pmatrix}
\varphi_\beta \\
\vdots
\end{pmatrix}
= 
\begin{pmatrix}
\mathbb{E} [q(\xi)\Phi_\alpha(z(\xi))]
\end{pmatrix},
$$

where $\alpha, \beta \in \mathcal{J}$, $A$ is of size $|\mathcal{J}| \times |\mathcal{J}|$. 

Numerical computation of NLBU

Using the same quadrature rule of order $q$ for each element of $A$, we can write

$$A = \mathbb{E} \left[ \Phi_{J_\alpha}(z(\xi))\Phi_{J_\beta}(z(\xi))^T \right] \approx \sum_{i=1}^{N^A} w_i^A \Phi_{J_\alpha}(z_i)\Phi_{J_\beta}(z_i)^T,$$  

(3)

where $(w_i^A, \xi_i)$ are weights and quadrature points, $z_i := z(\xi_i)$ and $\Phi_{J_\alpha}(z_i) := (...) \Phi_{\alpha}(z(\xi_i))...(\cdot)^T$ is a vector of length $|J_\alpha|$.

$$b = \mathbb{E} [q(\xi)\Phi_{J_\alpha}(z(\xi))] \approx \sum_{i=1}^{N^b} w_i^b q(\xi_i) \Phi_{J_\alpha}(z_i),$$  

(4)

where $\Phi_{J_\alpha}(z(\xi_i)) := (...) \Phi_{\alpha}(z(\xi_i)), ...)$, $\alpha \in J_\alpha$. 
We can write the Eq. 15 with the right-hand side in Eq. 4 in the compact form:

\[
[\Phi_A]\begin{bmatrix} \text{diag}(...w_i^A...)
\end{bmatrix} [\Phi_A]^T \begin{pmatrix} \vdots \\
\varphi_\beta \\
\vdots 
\end{pmatrix} = [\Phi_b] \begin{pmatrix} w_0^b q(\xi_0) \\
\vdots \\
w_{Nb}^b q(\xi_{Nb})
\end{pmatrix}
\tag{5}
\]

\[ [\Phi_A] \in \mathbb{R}^{J_\alpha \times N^A}, \begin{bmatrix} \text{diag}(...w_i^A...)
\end{bmatrix} \in \mathbb{R}^{N^A \times N^A}, [\Phi_b] \in \mathbb{R}^{J_\alpha \times N^b}, [w_0^b q(\xi_0)...w_{Nb}^b q(\xi_{Nb})] \in \mathbb{R}^{N^b}.
\]

Solving Eq. 5, obtain vector of coefficients \((...\varphi_\beta...)^T\) for all \(\beta\). Finally, the assimilated parameter \(q_a\) will be

\[
q_a = q_f + \bar{\varphi}(\hat{y}) - \bar{\varphi}(z),
\tag{6}
\]

\[
z(\xi) = y(\xi) + \varepsilon(\omega), \quad \bar{\varphi} = \sum_{\beta \in J_p} \varphi_\beta \Phi_\beta(z(\xi))
\]
Example 1: \( \varphi \) does not exist in the Hermite basis

Assume \( z(\xi) = \xi^2 \) and \( q(\xi) = \xi^3 \). The normalized PCE coefficients are \((1, 0, 1, 0)\)
\[
(\xi^2 = 1 \cdot H_0(\xi) + 0 \cdot H_1(\xi) + 1 \cdot H_2(\xi) + 0 \cdot H_3(\xi))
\]
and \((0, 3, 0, 1)\)
\[
(\xi^3 = 0 \cdot H_0(\xi) + 3 \cdot H_1(\xi) + 0 \cdot H_2(\xi) + 1 \cdot H_3(\xi)).
\]
For such data the mapping \( \varphi \) does not exist. The matrix \( A \) is close to singular.
Support of Hermite polynomials (used for Gaussian RVs) is \((-\infty, \infty)\).
Example 2: \( \varphi \) does exist in the Laguerre basis

Assume \( z(\xi) = \xi^2 \) and \( q(\xi) = \xi^3 \).
The normalized gPCE coefficients are \((2, -4, 2, 0)\) and (6, −18, 18, −6).
For such data the mapping mapping \( \varphi \) of order 8 and higher produces a very accurate result.
Support of Laguerre polynomials (used for Gamma RVs) is \([0, \infty)\).
Is a system of ODEs. Has chaotic solutions for certain parameter values and initial conditions.

\[
\begin{align*}
\dot{x} &= \sigma(\omega)(y - x) \\
\dot{y} &= x(\rho(\omega) - z) - y \\
\dot{z} &= xy - \beta(\omega)z
\end{align*}
\]

Initial state \(q_0(\omega) = (x_0(\omega), y_0(\omega), z_0(\omega))\) are uncertain.

Solving in \(t_0, t_1, ..., t_{10}\), Noisy Measur. \(\rightarrow\) UPDATE, solving in \(t_{11}, t_{12}, ..., t_{20}\), Noisy Measur. \(\rightarrow\) UPDATE,...
Trajectories of $x, y$ and $z$ in time. After each update (new information coming) the uncertainty drops. (O. Pajonk)
Figure: Partial state trajectory with uncertainty and three updates
Lorenz-84 Problem

Figure: NLBU: Linear measurement \((x(t), y(t), z(t))\): prior and posterior after one update
Figure: Linear measurement: Comparison posterior for LBU and NLBU after second update
Figure: Quadratic measurement \((x(t)^2, y(t)^2, z(t)^2)\): Comparison of a priori and a posterior for NLBU
Example 4: 1D elliptic PDE with uncertain coeffs

Taken from Stochastic Galerkin Library (sglib), by Elmar Zander (TU Braunschweig)

\[-\nabla \cdot (\kappa(x, \xi) \nabla u(x, \xi)) = f(x, \xi), \quad x \in [0, 1]\]

Measurements are taken at \(x_1 = 0.2\), and \(x_2 = 0.8\). The means are \(\overline{y}(x_1) = 10\), \(\overline{y}(x_2) = 5\) and the variances are 0.5 and 1.5 correspondingly.
Example 4: updating of the solution $u$

**Figure:** Original and updated solutions, mean value plus/minus 1,2,3 standard deviations

See more in sglib by Elmar Zander
Example 4: Updating of the parameter

Figure: Original and updated parameter $q$.

See more in sglib by Elmar Zander
Conclusion about NLBU

- Step 1. Introduced a way to derive MMSE $\varphi$ (as a linear, quadratic, cubic etc approximation, i.e. compute conditional expectation of $q$, given measurement $Y$.
- Step 2. Apply $\varphi$ to identify parameter $q$
- All ingredients can be given as gPC.
- we apply it to solve inverse problems (ODEs and PDEs).
- Stochastic dimension grows up very fast.
I used a Matlab toolbox for stochastic Galerkin methods (sglib)
https://github.com/ezander/sglib
Alexander Litvinenko and his research work was supported by the King Abdullah University of Science and Technology (KAUST), SRI-UQ and ECRC centers.


