Multi-fidelity modeling for time-series output

Baptiste Kerleguer$^1,2$, Claire Cannamela$^1$, Josselin Garnier$^2$

"École thématique sur les incertitudes en calcul scientifique " | October 2020

$^1$ CEA,DAM,DIF, F-91297, Arpajon, France,  
$^2$ Centre de Mathématiques Appliquées, Ecole Polytechnique, 91128 Palaiseau Cedex, France
Outline

1. Surrogate model and multi-fidelity
   - Surrogate model
   - Multi-fidelity
   - Time-series

2. Multi-fidelity time-series outputs model
   - The model
   - The resolution

3. Steps
   - Dimension Reduction
   - Multi-fidelity Gaussian process regression
   - Tensorisation of the covariance

4. Illustration
Model

Input parameters $x$

Physical system

$g_{\text{exp}}(x)$

Computer code

$g_{\text{num}}(x)$

Surrogate model

$\hat{g}(x)$
Multi-fidelity surrogate model

Low Fidelity Code

High Fidelity Code

Enriched

Surrogate Model
Time dependent outputs, definition and use

- Functional output of dimension 1
- Each realization of the code gives a 1D curve
- $Z(x, t_u)$ is known for $t_u$ with $u = 1, \ldots, N_t$ but only for some $x \in \mathbb{R}^d$. 
Objective: model a multi-fidelity and time-series outputs system

Idea: Combine the tensorized covariance method and Multifidelity CoKriging.

Decomposition of the problem into two parts:

\[ Z_L(x, t_u) = \sum_{i=1}^{N} A_{i,L}(x) \Gamma_i(t_u) + Z_L^\perp(x, t_u) \]

\[ Z_H(x, t_u) = \sum_{i=1}^{N} \left[ A_{i,H}(x) \Gamma_i(t_u) + Z_H^\perp(x, t_u) \right] \]

- dimension reduction
- orthogonal part

\( Z_L(x, t_u) \) and \( Z_H(x, t_u) \) refer to the low-fidelity and high-fidelity models, respectively.
Method in 3 steps

1. Dimension reduction, definition of the basis $\Gamma$ expression onto $\Gamma$.

2. Multi-fidelity CoKriging for $A_{i,H}(x)$ and $A_{i,L}(x)$.

3. Gaussian process regression with tensorized covariance for $Z_H^{1}(x, t_u)$.

- The value of $N$ is chosen by cross-validation.
We try to reduce the size of the outputs so we express:

\[ Z(x, t) = \sum_{i=1}^{N} A_i(x) \Gamma_i(t) + Z^\perp(x, t) \]

So we have \( A_i(x) \) that we can use to do our regression. We are back to the previous regression problem.
We propose different basis distributions for $\Gamma$:

**Dirac**

- We can give ourselves a basis a priori thanks to an expert judgement.

- The other solution is to carry out the SVD on the low fidelity data set.

**Empirical**

- The method is inspired by K-fold cross-validation:
  - First define a learning set out of the available low fidelity output set.
  - Second generate the basis with the SVD on the learning set.
  - Finally compute the first two moments of the basis elements.

→ allows to compute the predictive mean and variance for $Z_H(x, t_u)$. 
Definition of the empirical basis

Illustration of the construction of the bases taking into account the subset data used:
The quantities of interest: \( a_H(x) \in \mathbb{R} \), respectively \( a_L(x) \) with \( x \in \mathbb{R}^d \).

Hypothesis: \((a_H(x), a_L(x))\) realization of GP \((A_H(x), A_L(x))\).

Autoregressive CoKriging model from Kennedy et O’Hagan 2000; Le Gratiet et Garnier 2014:

\[
A_H(x) = \rho(x)A_L(x) + \delta(x),
\]

where \( \delta(x) \) GP independent of \( A_L(x) \) and \( \rho(x) \) adjustment linear form.

Prediction: when the hyperparameters of the model are known,

\[
[A_H(x)|\text{data}] \sim \mathcal{GP}(m_{A_H}(x), \sigma^2_{A_H}(x)),
\]

the quantities \( m_{A_H}(x) \) and \( \sigma^2_{A_H}(x) \) have **analytical expressions**.
Assumption: $z_H(x,t)$ realization of a Gaussian process $Z_H(x,t)$ with values in \(\text{Span}\{\Gamma_{N+1}(t_u)\cdots\Gamma_N(t_u)\}\).

Proposed approach PERRIN 2019: the covariance function is separable in time and space. The covariance function $R_x(x,x')$ in $x$ is Matèrn kernel. The covariance matrix in $t_u$ is estimated by maximum likelihood.

We keep the stationarity for $x$, but the non-stationarity in $t$ is used.

The process $Z_H(x,t_u) : [Z_H(x,t_u)|\text{data, hyperparameters}] \sim \mathcal{GP}(\mu_\star(x), R_\star(x,x',t_u))$ have analytical expressions PERRIN 2019.

Estimation of $R_x$ hyperparameters: Leave One Out-Cross validation:

\[
l_{c,\text{LOO}} = \arg\min_{l \in \mathbb{R}_+^{nx}} \sum_{k=1}^{nx} \| \mu_\star(-k) \left(x^{(k)}, l_c\right) - z\perp(k) \|^2,
\]
The double pendulum

We use the outputs of a calculation code that models the system with two levels of fidelity:

- **inputs**
  - $k$: spring stiffness
  - $M$: mobile mass
  - $l$: pendulum length
  - $y$: mobile position
  - $\theta$: pendulum angle
  - $m$: pendulum mass

- **output**
  - $z$: pendulum position

The difference between high and low fidelity is that for low fidelity we assume that the angle $\theta$ is small.
Examples of Outputs

Table 1.5 – Range of variation of the different system parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>$[3; 8]$</td>
</tr>
<tr>
<td>$k$</td>
<td>$[0.1; 2.1]$</td>
</tr>
<tr>
<td>$\theta_0$</td>
<td>$[\pi/4; \pi/2]$</td>
</tr>
<tr>
<td>$y_0$</td>
<td>$[0; 0.2]$</td>
</tr>
<tr>
<td>$\dot{y}_0$</td>
<td>0</td>
</tr>
<tr>
<td>$l$</td>
<td>2</td>
</tr>
<tr>
<td>$m$</td>
<td>0.5</td>
</tr>
<tr>
<td>$g$</td>
<td>9.81</td>
</tr>
</tbody>
</table>

For $N_H = 10$ (red), $N_L = 100$ (blue) and $N_t = 101$
Basis and orthogonal part of the example

First 4 elements of the basis

Orthogonal realisations for 5 different values of $x$
The result in $Q^2$:

Optimise in $\mathcal{N}$ different for each method and each data set
Example of prediction with and without orthogonal part

with

without
About this method:

- Uncertainty is fully estimated even for the orthogonal part.

- Our method is always better than the simple fidelity. The advantages of coKriging and tensorisation approaches are exploited.

- We are able to reduce the dimension and to exploit the orthogonal part.

Perspectives:

- The sequential learning

- Image output

- Non linear multi-fidelity approach


Questions ?