A Bayesian neural network approach to multi-fidelity surrogate modeling

UQSay # 70

7 march 2024

Baptiste Kerleguer (baptiste.kerleguer@cea.fr)[1], Claire Cannamela[1], Josselin Garnier[2]

[1]CEA, DAM, DIF, F-91297 Arpajon, France [2]CMAP, Ecole Polytechnique, Institut Polytechnique de Paris, 91128 Palaiseau Cedex, France

Presentation of the article

International Journal for Uncertainty Quantification, 14(1):43–60 (2024)

A BAYESIAN NEURAL NETWORK APPROACH TO MULTI-FIDELITY SURROGATE MODELING

Baptiste Kerleguer,[1,2,] Claire Cannamela,[1] & Josselin Garnier[2]*

[1]*Commissariat à l'Énergie Atomique et aus Energies Alternatives (CEA), DAM, DIF, Arpajon, France*

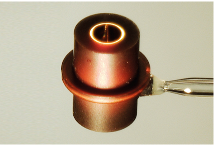
[2]*Centre de Mathématiques Appliquées, Ecole Polytechnique, Institut Polytechnique de Paris, 91128 Palaiseau Cedex, France*

*Address all correspondence to: Baptiste Kerleguer, CEA, DAM, DIF, F-91297, Arpajon, France,

Motivation of surrogate modeling

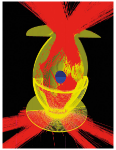
Megajoule laser experiment [CEA DAM, 2021]

- **Physical system** (Inertial confinement fusion)
  - Manufacture of targets $\sim$ 1 year
  - Set up of the installation $\sim$ 2 months
  - Performing an experiment $\sim$ 1 day



- **Computer code**
  - Computation of a numerical experiment $\sim$ 1 week

  $\rightarrow$ Uncertainty quantification requires "many" simulations, intractable with this computer code.

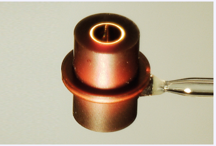


- **Surrogate model**
  - Computing a surrogate experiment $\sim$ 1 second

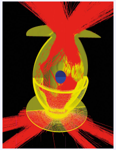
Motivation of surrogate modeling

Input parameters x

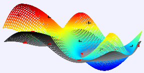
Physical system



Computer code



Surrogate model



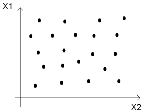
$z_{\mathrm{exp}}(x)$ $\approx$ $z(x)$ $\approx$ $\tilde{z}(x)$

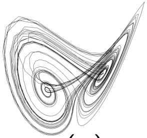
Experimental | Simulate | Predict

Multi-fidelity surrogate model

- **Two versions** of the code are available:

Low Fidelity Code (Cheap)



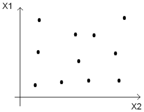


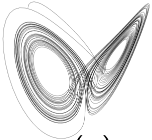
$\longrightarrow$ Low Fidelity

x $\qquad$ $z_L(x)$

High Fidelity Code (Expensive)

Enriched





$\longrightarrow$ Surrogate Model

x $\qquad$ $z_H(x)$

Goal: Construct a surrogate from $\left(z_L(x^{(i)})\right)_{i=1}^{N_L}$ and $\left(z_H(x^{(i)})\right)_{i=1}^{N_H}$ $N_H < N_L$

Hierarchical 2 levels multi-fidelity

2 versions of the same code

- **Low-fidelity** : cheap and approximation
- **High-fidelity** : expensive and very accurate

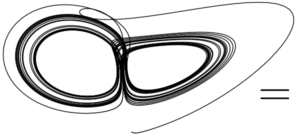
$$Z_H(x) \quad = \quad \rho\left(x,\ Z_L(x) \quad\right) + \quad \delta(x)$$

- The interaction between code depends on the model.
- Low-fidelity is learn with an independant surrogate model.

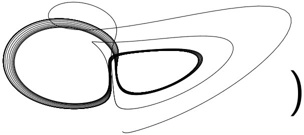
Hierarchical 2 levels multi-fidelity

2 versions of the same code

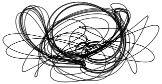
- **Low-fidelity** : cheap and approximation
- **High-fidelity** : expensive and very accurate



$= \rho(x,$



$) +$



- The interaction between code depends on the model.
- Low-fidelity is learn with an independant surrogate model.

Goal

Method for surrogate modeling in a multi-fidelity framework

Stat-of-the-art methods:

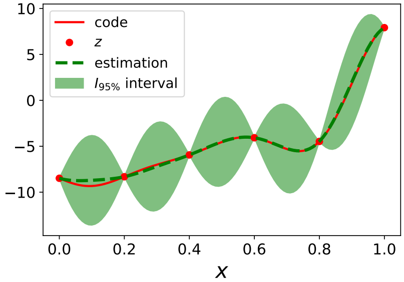
- Multi-fidelity surrogate modeling with simple interactions and uncertainties quantification,
- Multi-fidelity surrogate modeling with complexe interactions and without uncertainties quantification.

Challenges:

- Take into-account non-linear interactions and non-given interactions between fidelities.
- Quantify the prediction uncertainties associated with the surrogate model.

Simple fidelity Gaussian process regression

- Hypothesis: $z(x)$ is a realization of a Gaussian process (GP) $Z(x)$
- We have N observations $z(x^{(i)}) = y^{(i)}$, $i = 1, \cdots, N$.
- The conditional GP gives a prediction of $z(x)$, with analytical expressions for mean and variance.



This framework is presented in [Williams and Rasmussen, 2006].

Multi-fidelity with scalar outputs

Problem: We want to predict a costly code outputs $a_H(x) \in \mathbb{R}$, with $x \in \mathbb{R}^d$. We also have access to a cheaper code $a_L(x)$ with more observations available.

Multi-fidelity with scalar outputs

Problem: We want to predict a costly code outputs $a_H(x) \in \mathbb{R}$, with $x \in \mathbb{R}^d$. We also have access to a cheaper code $a_L(x)$ with more observations available.

State-of-the-art GP-based methods:

- Multi-fidelity AR(1) Gaussian process regression [Kennedy and O'Hagan, 2000].
- Multi-fidelity with Deep Gaussian processes [Perdikaris et al., 2017].
- Neural network for multi-fidelity [Meng et al., 2020].

Multi-fidelity AR(1) Gaussian process regression

- Hypothesis: The emulator is a Gaussian process $(A_H(x), A_L(x))$.

- Autoregressive CoKriging model from [Kennedy and O'Hagan, 2000] :

$$A_H(x) = \rho(x)A_L(x) + \delta(x),$$

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

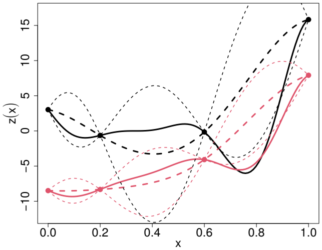
- Estimation of hyperparameters: Maximum likelihood [Le Gratiet and Garnier, 2014], [Ma, 2020].

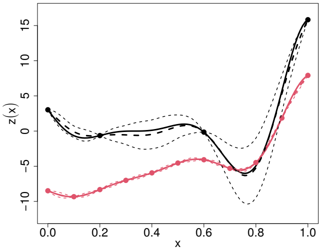
- Prediction: We have

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

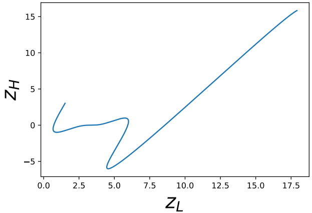
the quantities $m_{A_H}(x)$ and $\sigma^2_{A_H}(x)$ have analytical expressions.

Illustration AR(1) surrogate model





Linear interactions between low- and high-fidelity:



AR(1) multi-fidelity $\longrightarrow$ for linear interactions

If the interaction between fidelities is non-linear ?

Deep Gaussian process for multi-fidelity [Perdikaris et al., 2017]

- Hypotheses :
  - There is a known relation between fidelity
  - The output of the code is a realization of a Gaussian process

$$f_L(\mathbf{x}) = h_L(\mathbf{x}),$$

$$f_L(\mathbf{x}) = h_H(\mathbf{x}, f_L(\mathbf{x})) + \delta(\mathbf{x}),$$

with $h_{L,H}$ two GPs and δ a GP.
The GP prior f_L with the GP posterior from the previous inference level $f_L^\star(\mathbf{x})$. Then, using the additive structure , along with the independence assumption between the GPs f_L and δ, we can summarize the autoregressive scheme as

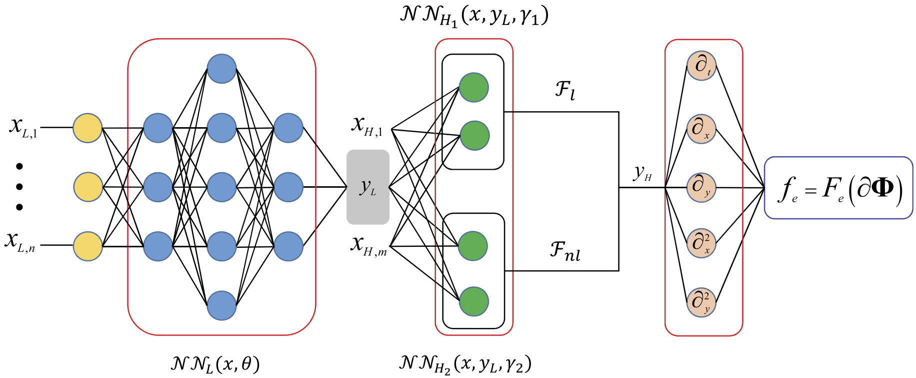
$$f_L(\mathbf{x}) = g_L(\mathbf{x}, f_L^\star(\mathbf{x})),$$

where $g_L \sim \mathcal{GP}\left(f_L|\mathbf{0}, k_2((\mathbf{x}, f_1^\star(\mathbf{x})), (\mathbf{x}, f_1^\star(\mathbf{x})), \theta)\right)$. θ is the hyperparameters of the model.

The machine learning option

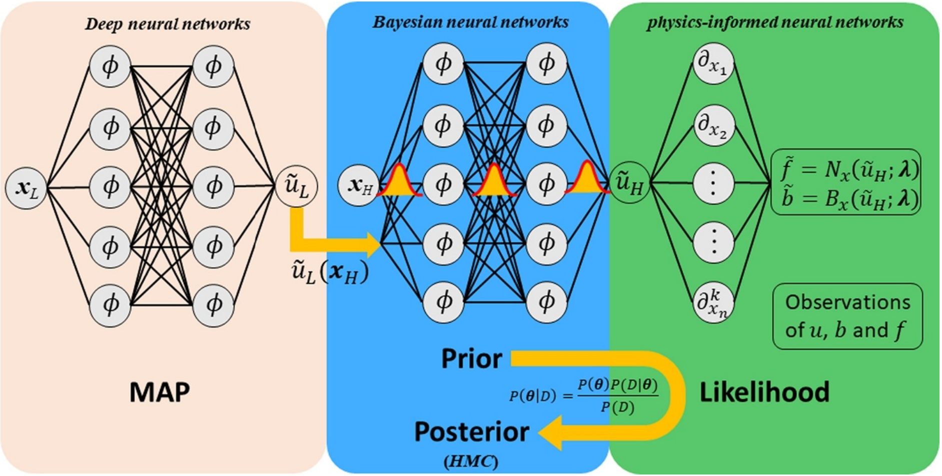
Multi-fidelity neural networks [Meng et al., 2020]

X. Meng, G.E. Karniadakis / Journal of Computational Physics 401 (2020) 109020



The machine learning option

Multi-fidelity Bayesian neural networks [Meng et al., 2020]



Multi-fidelity with scalar outputs

Problem: We want to predict a costly code outputs $a_H(x) \in \mathbb{R}$, with $x \in \mathbb{R}^d$. We also have access to a cheaper code $a_L(x)$ with more observations available.

State-of-the-art GP-based methods:

- Multi-fidelity AR(1) Gaussian process regression [Kennedy and O'Hagan, 2000].
- Multi-fidelity with Deep Gaussian processes [Perdikaris et al., 2017].
- Neural network for multi-fidelity [Meng et al., 2020].

Proposed approach:

- Gaussian process and Bayesian Neural network combined (GP-BNN) [Kerleguer, et al., 2024].

Bayesian Neural Network (BNN)

- Bayesian neural network (BNN) for regression.
- We start from the same formalism as a neural network:

$$y = g(w_1 x + \beta_1)$$

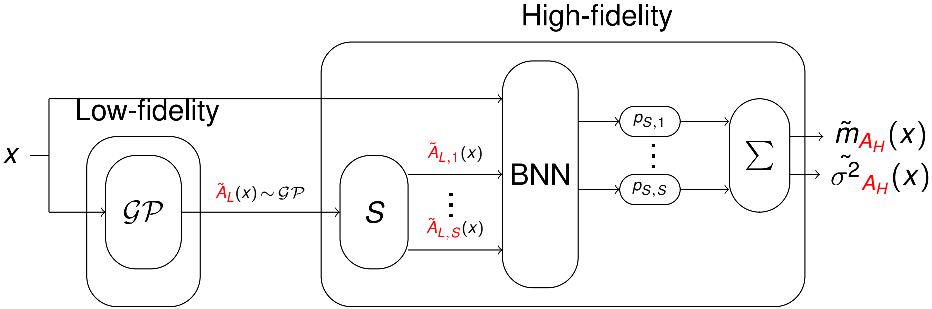
$$BNN(x) = w_2 y + \beta_2$$

$w_1 \in \mathbb{R}^{q\times d}$, $\beta_1 \in \mathbb{R}^q$, $w_2 \in \mathbb{R}^q$, $\beta_2 \in \mathbb{R}$ and g activation function.

- The parameters w_i and β_i follow the Bayesian formalism.
- To predict we use Markov-Chain Monte-Carlo methods like Hamiltonian Monte Carlo.

GP-BNN [Kerleguer, et al., 2024]

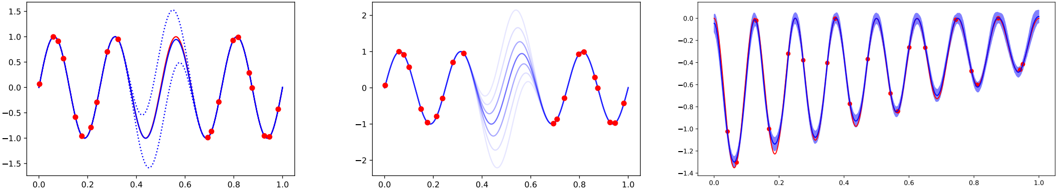
- The low-fidelity code is modeled by a Gaussian process.
- The High-fidelity and high-low interaction is modeled by a Bayesian Neural Network.
- The predictive distribution of the low-fidelity is transfered to the BNN using Gauss-Hermite quadrature.



GP-BNN estimators

The Gauss-Hermite sampling in the GP-BNN method.

Low-fidelity Sampling High-fidelity



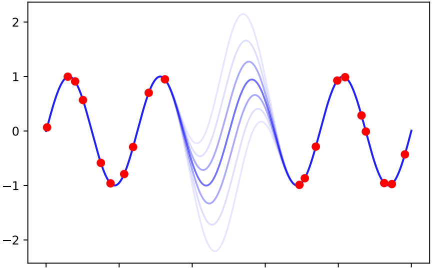
- GP-BNN $\longrightarrow$ for non-linear interactions

Sampling of $\tilde{A}_L(\mathbf{x})$

- Let $z_{S,i}$ be the roots of the Hermite polynomials $H_S(x) = (-1)^S e^{x^2} \partial_x^S e^{-x^2}$, $S \in \mathbb{N}$.
- For each input $\mathbf{x}$ the GP posterior law has mean $\mu_L(\mathbf{x})$ and covariance $C_L(\mathbf{x}, \mathbf{x})$.
- Therefore, the ith realization in the Gauss-Hermite quadrature formula is:

$$\tilde{f}_{L,i}(\mathbf{x}) = \mu_L(\mathbf{x}) + z_{S,i}\sqrt{C_L(\mathbf{x}, \mathbf{x})},$$

- the associated weight is $p_{S,i} = \frac{2^{S-1} S! \sqrt{\pi}}{S^2 H_{S-1}^2(z_{S,i})}$, for $i = 1, \ldots, S$.



The High-fidelity BNN

- The inputs: $\mathbf{x}$ and $\tilde{f}_{L,i}(\mathbf{x})$
- Output: $BNN_{\theta}(\mathbf{x}, \tilde{f}_{L,i}(\mathbf{x}))$
- The estimator of the predictive mean of the output of the high-fidelity model is:

$$\tilde{f}_H(\mathbf{x}) = \frac{1}{N_v} \sum_{j=1}^{N_v} \sum_{i=1}^{S} p_{S,i} BNN_{\theta_j}(\mathbf{x}, \tilde{f}_{L,i}(\mathbf{x})),$$

and the estimator of the predictive variance is:

$$\tilde{C}_H(\mathbf{x}) = \frac{1}{N_v} \sum_{j=1}^{N_v} \left(\sum_{i=1}^{S} p_{S,i} BNN_{\theta_j}(\mathbf{x}, \tilde{f}_{L,i}(\mathbf{x})) \right)^2 - \tilde{f}_H^2(\mathbf{x}) + \frac{1}{N_v} \sum_{j=1}^{N_v} \left(\sum_{i=1}^{S} p_{S,i}^2 \right) \sigma_j^2 .$$

$\rightarrow$ Also available for Mean-Standard Deviation Method, Quantiles Method

Hyperparameters of the model

Low fidelity model

- Prior mean and parameters (null mean and classical DICEkriging priors)
- Kernel function (Matèrn $\frac{5}{2}$)
- (Low fidelity Sampler)

High fidelity

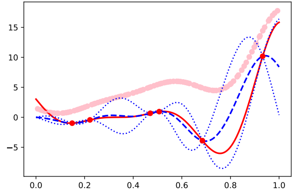
- Dimension of the neural network (Set at 100)
- Number of samples in the MCMC estimator ($N_v = 500$ explaine in [Kerleguer, et al., 2024])
- Number of samples of the low-fidelity Gaussian process S (Explain in the following)

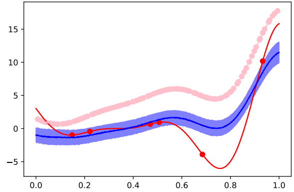
Multi-Fidelity scalar illustration

Interaction	AR(1)	GP-BNN

Linear

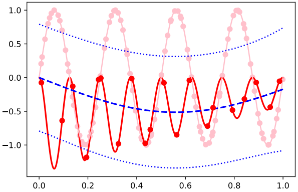
[Forrester et al., 2007]

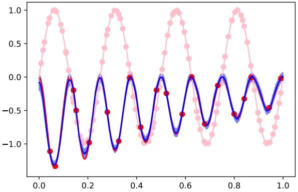




Non-linear

sin-sin^2





Evaluation of surrogate models

Q^2

$$Q^2_{\mathrm{T}} = 1 - \frac{\sum_{i=1}^{N_{\mathrm{T}}} \left[\tilde{\mu}_H(\mathbf{x}_{\mathrm{T}}^{(i)}) - f_H(\mathbf{x}_{\mathrm{T}}^{(i)})\right]^2}{N_{\mathrm{T}} \mathbb{V}_{\mathrm{T}}\left(f_H\right)},$$

with $\mathbb{V}_{\mathrm{T}}(f_H) = \frac{1}{N_{\mathrm{T}}} \sum_{i=1}^{N_{\mathrm{T}}} \left[f_H(\mathbf{x}_{\mathrm{T}}^{(i)}) - \frac{1}{N_{\mathrm{T}}} \sum_{j=1}^{N_{\mathrm{T}}} f_H(\mathbf{x}_{\mathrm{T}}^{(j)})\right]^2$.

- A highly predictive model gives a Q^2_{T} close to 1 while a less predictive model has a smaller Q^2_{T}.

Evaluation of the uncertainty interval

- The uncertainty prediction interval is not taken into account with Q^2. Two metrics are therefore introduced: the coverage probability and the mean predictive interval width. Both were studied in [Acharki et al., 2023] and [Kerleguer, et al., 2024].

Evaluation of surrogate models

Coverage probability (CP)

- CP_α: probability for $f_H(\mathbf{x}_\mathrm{T})$ to be within the prediction interval with confidence level α :

$$\mathrm{CP}_\alpha = \frac{1}{N_\mathrm{T}} \sum_{i=1}^{N_\mathrm{T}} \mathbf{1}_{f_H(\mathbf{x}_\mathrm{T}^{(i)}) \in \mathcal{I}_\alpha(\mathbf{x}_\mathrm{T}^{(i)})},$$

  with $\mathbf{1}$ the indicator function and $\mathcal{I}_\alpha(\mathbf{x})$ the prediction interval at point $\mathbf{x}$ with confidence level α.

- The prediction uncertainty of the surrogate model is well characterized when CP_α is close to α.

Mean Predictive Interval Width

- The mean predictive interval width MPIW_α is the average width of the prediction intervals:

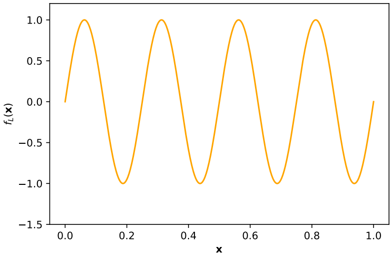
$$\mathrm{MPIW}_\alpha = \frac{1}{N_\mathrm{T}} \sum_{i=1}^{N_\mathrm{T}} \left|\mathcal{I}_\alpha(\mathbf{x}_\mathrm{T}^{(i)})\right|,$$

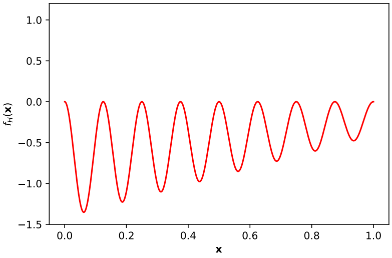
  where $|\mathcal{I}_\alpha(\mathbf{x})|$ the length of the prediction interval $\mathcal{I}_\alpha(\mathbf{x})$.

- The prediction uncertainty of the surrogate model is small when MPIW_α is small.

Parameters of the surrogate model

A test multi-fidelity function [Perdikaris et al., 2017]

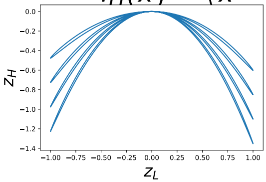




$$f_L(x) = \sin 8\pi x$$

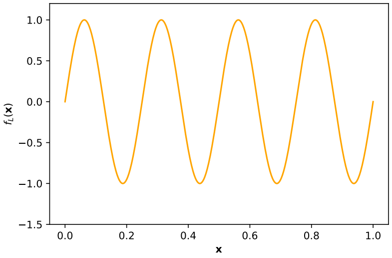
$$f_H(x) = (x - \sqrt{2})\, f_L^2(x)$$

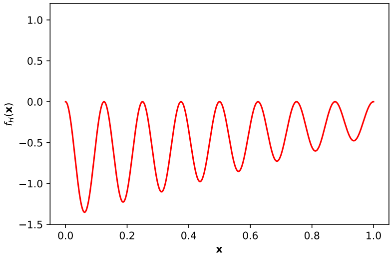
Interactions :



Parameters of the surrogate model

A test multi-fidelity function [Perdikaris et al., 2017]





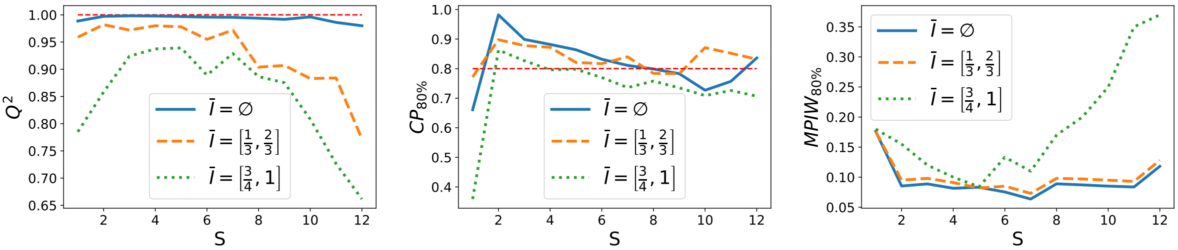
Low fidelity learning intervals

The low-fidelity learning interval is $I = [0, 1] \backslash \bar{I}$

$\bar{I}$	$\emptyset$	$[\frac{1}{3}, \frac{2}{3}]$	$[\frac{3}{4}, 1]$
Q^2_{L}	0.99	0.98	0.84

Optimal S for this example

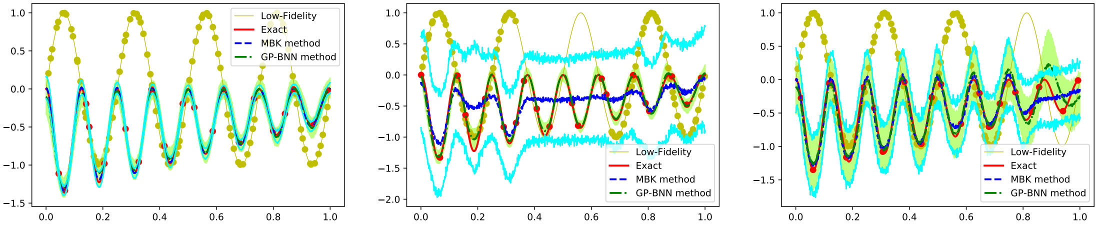
- GPBNN performance with different values of S



$\rightarrow$ $S = 5$, number of neurons $N_n = 100$ and number of samples $N_v = 500$

Performance of the surrogate model

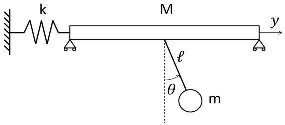
MBK method is proposed in [Meng et al., 2020]



→ Prediction performance is equivalent, but in areas where low fidelity is poorly reconstructed, GPBNN performs better.

→ The uncertainty interval is more relevant in all cases for the GPBNN method

Double pendulum



	GP 1F	AR(1)	[Perdikaris et al., 2017]	[Meng et al., 2020]	[Kerleguer, et al., 2024]
Q^2_{T}	0.93	0.94	**0.95**	0.54	**0.95**
$\mathrm{CP}_{80\%}$	**0.81**	**0.78**	0.62	0.88	**0.80**
$\mathrm{MPIW}_{80\%}$	0.154	0.146	0.069	0.859	**0.101**

Q^2_{T}: represent the quadratic error.
$\mathrm{CP}_{80\%}$: coverage probabilité of the Uncertainty Quantification.
$\mathrm{MPIW}_{80\%}$ size of the Uncertainty interval.

Perspectives

Gaussian-Process Bayesian Neural Network:

- Modele available on [Kerleguer, et al., 2024]
- **Uncertainty prediction** for both high- and low-fidelity
- Interactions between high- and low-fidelity **linear and non-linear**

Adaptations needed:

- Growing the **dimension of output**
- Image processing
- **Non-hierarchical** Multi-fidelity surrogate modeling

Hierarchical Multi-fidelity is more than

MF : Multi-Fidelity, GP : Gaussien Process, NN : Neural Network

	Scalar	Time-series
Linear	Co-Kriging MF x x autorecursive MF x x	MF time series x x
Non-linear	Deep GP x x **GPBNN** x x Deep MF Transfert learning	Deep MF (MF wavelet GP x x) Transfert learning

Today's methods
(work in progress)

x Avec quantification d'incertitudes
x Small Data

References

Acharki Naoufal, Antoine Bertoncello, and Josselin Garnier (2023)
Robust prediction interval estimation for Gaussian processes by cross-validation method
Computational Statistics & Data Analysis, 178 (2023): 107597.

CEA DAM (2021)
Première expérience de fusion nucléaire au laser mégajoule
www-dam.cea.fr

Alexander Forrester, Andras Sóbester and Andy Keane (2007)
Multi-fidelity optimization via surrogate modelling
Proceedings of the royal society A: mathematical, physical and engineering sciences, 463(2088), 3251-3269.

M. C. Kennedy and A. O'Hagan (2000)
Predicting the output from a complex computer code when fast approximations are available
Biometrika,(2000), 87, 1, pp. 1–13.

Baptiste Kerleguer, Claire Cannamela, and Josselin Garnier (2024)
A Bayesian neural network approach to Multi-fidelity surrogate modelling
In Review in *International Journal for Uncertainty Quantification*.
International Journal for Uncertainty Quantification 14(1):43–60.

Loic Le Gratiet and Josselin Garnier (2014)
Recursive co-kriging model for design of computer experiments with multiple levels of fidelity
International Journal for Uncertainty Quantification 4(5).

References

Edward Lorenz (1963)
Deterministic Nonperiodic Flow
Journal of the Atmospheric Sciences, 20(2), 130-141.

Pulong Ma (2020)
Objective Bayesian analysis of a cokriging model for hierarchical multifidelity codes.
SIAM/ASA Journal on Uncertainty Quantification, vol. 8, no 4, p. 1358-1382.

Xuhui Meng, Hessam Babaee and George Em Karniadakis (2021)
Multi-fidelity Bayesian neural networks: Algorithms ans applications
Journal of Computational Physics,401 (2021): 110361.

Paris Perdikaris, Maziar Raissi, Andreas Damianou, Neil Lawrence, & George Karniadakis (2017)
Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling
Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences,473.2198 (2017): 20160751.

Williams, Christopher and Carl Edward Rasmussen (2006)
Gaussian processes for machine learning
Cambridge, MA : MIT press,(2006).