A Bayesian methodology for hybrid degradation prognostics

E. Jaber[1,2,3],
E. Remy[1], V. Chabridon[1], M. Mougeot[2], D. Lucor[3],

1. EDF R&D / PRISME Department
2. Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli
3. Laboratoire Interdisciplinaire Des Sciences Du Numérique, LISN-CNRS

Workshop on Mathematical and Computational Foundations of Digital Twins - CIRM, Marseille

08/15/2025

Table of contents

1. Introduction
2. The BMU algorithm
3. Ensemble Kalman smoothing
4. Application to clogging of SGs in NPWRs
5. Summary
6. Appendices

Clogging of steam generators (SGs)

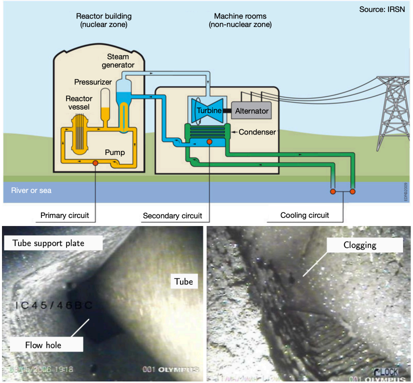
- Clogging of SGs is a complex multiphysics phenomenon that occurs following long operational periods in pressurized-water reactors (PWR) of the French nuclear fleet $\rightarrow$ undermines performance $\&$ weakens the structures $\rightarrow$ *may require chemical cleanings*
- 

Figure: PWR scheme, and example of video examination during a PWR outage (© IRSN, EDF)

Clogging of SGs

- No state-of-the-art model allowing for ground insights on diagnosis and prognosis of clogging rate τ_c $\rightarrow$ very hard to model & challenging to create reproducible lab experiment for model validation + not a lot of literature [Srikantiah and Chappidi, 2000; Prusek et al., 2013; Girard, 2014; Yang et al., 2017]
- Available scarce video field data as well as indirect measurements $\rightarrow$ allow to construct data-driven regression algorithms [Pinciroli et al., 2021] $\approx$ not enough data to have robust predictive models
- Another tool is the physical clogging model developed by [Prusek et al., 2013] $\rightarrow$ subsequent numerical model THYC-Puffer-DEPO [Feng et al., 2023] $\approx$ lack of enough trustworthy field data for precise V&V
- Necessary decision-making on chemical cleaning planning under uncertainty $\rightarrow$ ***how to make use of the available knowledge and models for achieving reliable predictions?***

DTs for NPP components

- Growing interest of creating digital twins (DTs) for the nuclear industry $\rightarrow$ still many industrial challenges to address

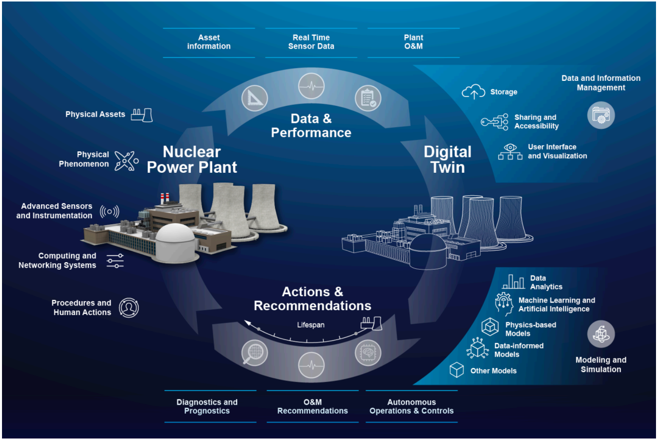


Figure: DT methodologies for nuclear reactors [Vaibhav and al., 2023]

DTs for NPP components

Excerpts from the report [Vaibhav and al., 2023]:

- *For producing a DT, especially one for an NPP, two broad technological needs must be met are: (1) modeling and simulation and (2) data and information management.*
- *The objective of implementing a DT system is to provide actions and recommendations in support of safe, reliable, and efficient system operations. To this end, DT actions and recommendations have been classified into the following categories: diagnostics and prognostics, operations & management recommendations, and autonomous operations and controls*
- *Diagnostics and prognostics [...] can be enabled in real time by DTs, thus providing plant staff with real time notification and recommendations on emergent or future conditions. Predictive algorithms in DTs can even go beyond diagnostics and prognostics to generate recommendations for efficient O&M practices*

Notions of prognostics

- Degradation level of the system $(t \mapsto \delta(t)) \rightarrow$ predict the remaining useful life (RUL) [Biggio and Kastanis, 2020] for a fixed threshold $D \in \mathbb{R}_+$:

$$\mathrm{RUL}(D) = \operatorname*{argmin}_{t_1 < t \leq t_N} \{\delta(t) \geq D\} \tag{1}$$

- Relies on physics-based simulation codes, and/or data driven methods $\rightarrow$ typical DT methodologies
- RUL prediction with each individual approach could lack robustness $\rightarrow$ high level of uncertainty $\&$ individual predictions not fully reliable

Our available tools

- *Physics-based computer simulation model* $g : \mathcal{X} \subset \mathbb{R}^d \to \mathbb{R}^N$ with prior uncertainty on input variables $\boldsymbol{X} = (X_1, \ldots, X_d) \sim \mu_{\boldsymbol{X}}$, one input $\boldsymbol{x}_0$ gives a full trajectory:

$$g(\boldsymbol{x}_0) = (g(t_1, \boldsymbol{x}_0), \ldots, g(t_N, \boldsymbol{x}_0)) \tag{2}$$

  and $\mathrm{pr}_\ell \circ g(\boldsymbol{X}) := g(t_\ell, \boldsymbol{X})$ models $\delta(t_\ell) \to$ grey-box, physics known but code cannot be modified

- q *heterogeneous degradation data groups* (from different sensors, statistical models,...) $\mathcal{D} = (\boldsymbol{y}^1, \ldots, \boldsymbol{y}^q)$ with different sizes $\boldsymbol{y}^i \in \mathbb{R}^{m_i} \to$ corresponding to different time indices in $\mathcal{J}_i$ so that $\mathcal{J} = \cup_{i=1}^q \mathcal{J}_i$ and $|\mathcal{J}| = m_1 + \ldots + m_q$, such that:

$$y^i(t_\ell) = \delta(t_\ell) + \eta_\ell^i, \tag{3}$$

  with $\eta_\ell^i \sim \mathcal{N}(0, R^i) \to$ homoskedastic noise for each data group

- ***How to fuse these tools for hybrid RUL estimation of the system?***

Offline data assimilation

- Perform offline sequential data assimilation over time windows [Geir Evensen, 2022] using ensemble methods
- Let $\boldsymbol{Z} = (\boldsymbol{X}, \boldsymbol{Y})$ where $\boldsymbol{Y} = g(\boldsymbol{X})$ are the computer model outputs on a prescribed time window, $\boldsymbol{X}$ are the uncertain parameters that do not vary over time
- Denote the data-groups as $\mathcal{D}$, the goal of assimilation is to estimate the distribution $p(\boldsymbol{Z} \mid \mathcal{D})$
- Idea is *modular* approach on each time window:
  1. Tailored Bayesian model updating (BMU) to obtain a posterior distribution h such that $(\boldsymbol{X} \mid \mathcal{D}) \sim h$,
  2. perform smoothing to obtain the density $p(\boldsymbol{Y} \mid \mathcal{D}, \boldsymbol{X} \sim h)$, so that the resulting assimilated posterior is approximated as:

$$p(\boldsymbol{Z} \mid \mathcal{D}) \simeq q(\boldsymbol{X}) p(\boldsymbol{Y} \mid \mathcal{D}, \boldsymbol{X} \sim h) \tag{4}$$

Illustration of our methodology

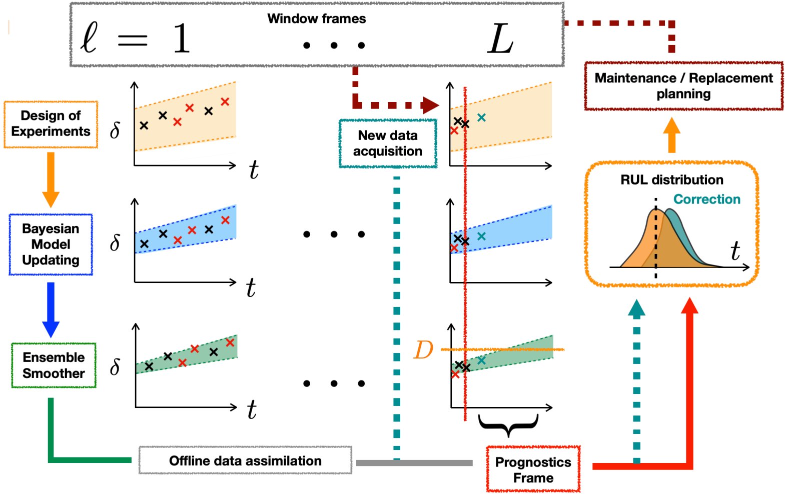


Figure: A sketch of the offline data assimilation methodology.

On the prognostics window

- Assimilation is performed offline until the prognostics window is reached $\ell = L$, then the RUL distribution can be computed:

$$\mathbb{P}(\mathrm{RUL}(D) \leq t_j | \mathcal{D}) = \int_{\mathbb{R}} \mathbf{1}\{\mathrm{pr}_{j+1}(\boldsymbol{y}) \geq D\} f_{\boldsymbol{Y} \mid \mathcal{D}, \boldsymbol{X} \sim h}(\boldsymbol{y}) d\boldsymbol{y} \qquad (5)$$

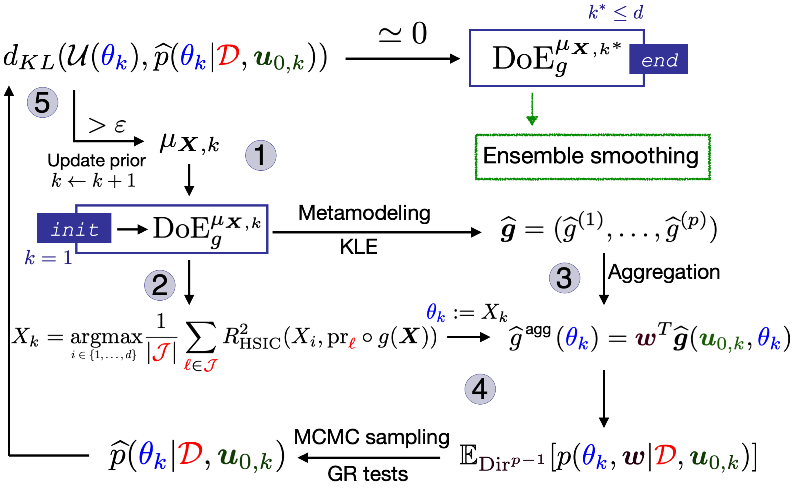
- Where in practice this probability is estimated using a Monte Carlo ensemble $\{(\boldsymbol{X}^{(i)}, g(\boldsymbol{X}^{(i)}))\}_{i=1}^{n} \sim h \otimes g \# h$:

$$\mathbb{P}(\mathrm{RUL}(D) \leq t_j \mid \mathcal{D}) \approx \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{\mathrm{pr}_{j+1} \circ g(\boldsymbol{X}^{(i)}) \geq D\} \qquad (6)$$

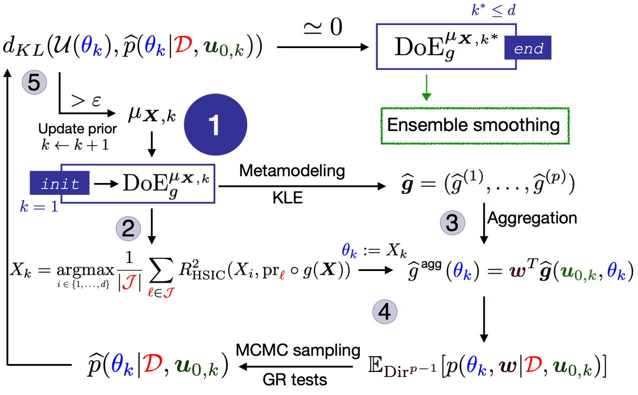
Table of contents

1. Introduction
2. The BMU algorithm
3. Ensemble Kalman smoothing
4. Application to clogging of SGs in NPWRs
5. Summary
6. Appendices

The BMU algorithm



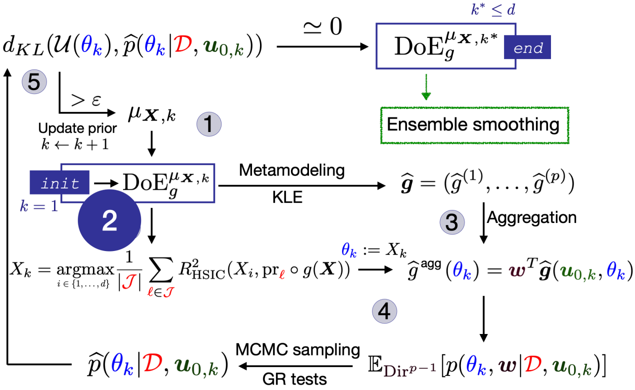
Step 1



Perform k iterations where $1 \leq k \leq d$:

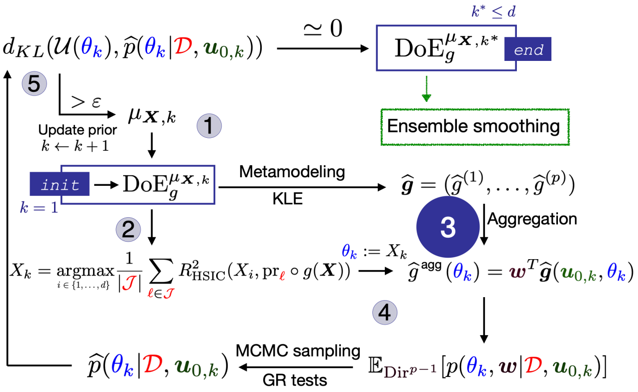
1. If $k = 0$, assume uniform independent priors $\mu_{\boldsymbol{X},k} \simeq \mathcal{U}[-1,1]^{\otimes d} \rightarrow$ generate a design of experiments $\mathrm{DoE}_g^{\mu_{\boldsymbol{X}},k} = \{(\boldsymbol{X}^{(j)}, g(\boldsymbol{X}^{(j)}))\}_{1 \leq j \leq n}$

Step 2



2. Compute **HSIC indices** [Gretton et al., 2005] between input variables and outputs at data time instances $\rightarrow$ **given data** sensitivity analysis method to assess *individual* input variable influence on the output

Step 3



3. If g is time-costly, build and validate p metamodels $\widehat{\boldsymbol{g}} = (\widehat{g}^{(1)}, \dots, \widehat{g}^{(p)})$ with chosen strategy $\rightarrow$ avoid metamodeling bias with *convex aggregation* on the unit-simplex chosing $\boldsymbol{w} \in \Delta^{p-1} := \{\boldsymbol{w} \in [0,1]^p, \ \|\boldsymbol{w}\|_1 = 1\}$, fix nominal value of $\boldsymbol{U}_{0,k} = \boldsymbol{u}_{0,k}$ by taking the mean

Metamodeling step 3

The metamodeling process involves:

- *Data generation:* Using the DoE of g at n input samples $\{\boldsymbol{X}^{(i)}\}_{i=1}^{n} \sim \mu_{\boldsymbol{X}}$, assemble the data matrix:

$$\boldsymbol{Y} = \left[g(\boldsymbol{X}^{(1)}), \ldots, g(\boldsymbol{X}^{(n)})\right] \in \mathbb{R}^{N \times n} \tag{7}$$

- *Dimensionality reduction:* Apply a Karhunen–Loève (KL) decomposition [Sullivan, 2015] using the empirical covariance matrix $\widehat{C} = \frac{1}{n}\boldsymbol{Y}\boldsymbol{Y}^{\top}$. Perform singular value decomposition (SVD):

$$\boldsymbol{Y} = \boldsymbol{V}\Sigma\boldsymbol{W}^{\top}, \tag{8}$$

  where $\boldsymbol{V}$ contains the KL modes $\{\Phi_k\}_{k=1}^{m}$, and Σ holds the singular values.

- *Mode selection:* Retain m modes to capture a prescribed variance (e.g., 99%). Project trajectories onto the retained modes:

$$\xi_k(\boldsymbol{X}^{(i)}) = g(\boldsymbol{X}^{(i)})^{\top}\Phi_k, \quad k = 1, \ldots, m \tag{9}$$

Metamodeling step 3

- *Surrogate modeling:* For each mode k, construct a surrogate model $\widehat{\xi}_k(\boldsymbol{X})$ using a Gaussian process [Rasmussen and Williams, 2006] with identical prior mean and kernel for all modes
- *Reconstruction:* Reconstruct the full trajectory using the surrogate models:

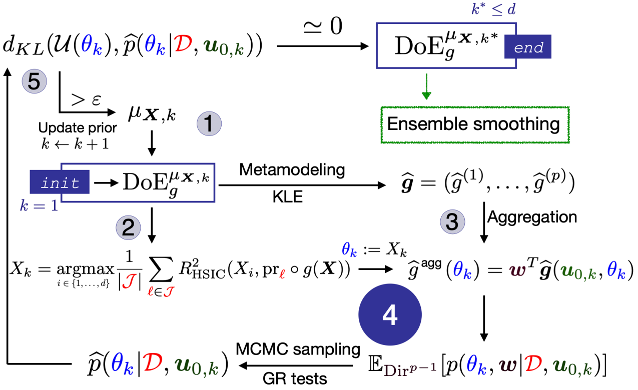
$$\widehat{g}(\boldsymbol{X}) = \sum_{k=1}^{m} \widehat{\xi}_k(\boldsymbol{X}) \boldsymbol{\Phi}_k \tag{10}$$

- *Aggregation:* Combine multiple surrogate models $\{\widehat{g}^{(i)}\}_{i=1}^{p}$ using convex aggregation weights $\boldsymbol{w} \in \Delta^{p-1}$ to form the aggregated surrogate model:

$$\widehat{g}^{\text{agg}}(\boldsymbol{X}) = \sum_{i=1}^{p} w_i \widehat{g}^{(i)}(\boldsymbol{X}) \tag{11}$$

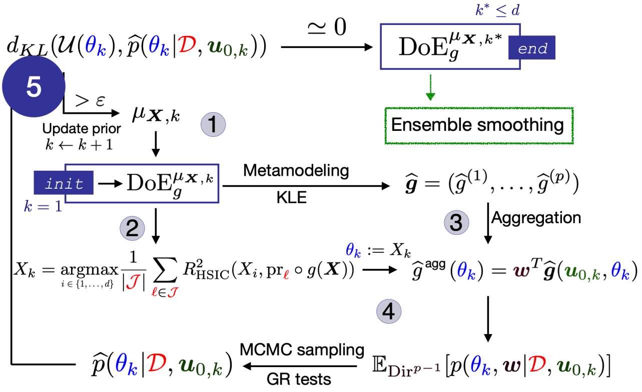
This ensures robustness by leveraging multiple models while minimizing bias

Step 4



4. Estimate the posterior distribution $\widehat{p}(\theta_k|\boldsymbol{y}^1, \ldots, \boldsymbol{y}^q, \boldsymbol{u}_{0,k})$ with an MCMC sampling procedure

Step 5



5. Compute the Kullback-Leibler divergence d_{KL} between prior distribution $\mathcal{U}(\theta_k)$ and the estimated density:
   - If $d_{KL} > \varepsilon$, update the prior $\mu_{\boldsymbol{X},k}$ by replacing marginal $\mathcal{U}(\theta_k)$ with $\widehat{p}(\theta_k|\boldsymbol{y}^1, \ldots, \boldsymbol{y}^q, \boldsymbol{u}_{0,k})$ and continue $k \leftarrow k+1$
   - Otherwise, stop and obtain an *updated* RUL prediction by computing $g\#\mu_{\boldsymbol{X},k^*}$

Bayesian updating step 4

Proposition

Assume $\lambda := 1/\sigma_{\boldsymbol{\eta}}^2 \sim \mathcal{G}(\frac{m}{2}, \frac{1}{2}\|\mathbf{y} - f(\theta)\|^2)$ *(Gamma distribution), where* m *is the number of data points in* $\mathbf{y}$*;* $\theta \sim \mathcal{U}(\theta)$*, and* $p(\theta, \lambda) \propto \lambda^{-1}$*. Then:*

$$p(\theta|\mathbf{y}) \propto \|\mathbf{y} - f(\theta)\|^{-m} \tag{12}$$

Moreover, if multiple groups of data at different time-instances are considered, $\mathbf{y}^1, \ldots, \mathbf{y}^q$*, with respective priors on the inverse of their standard deviations* $\lambda_i \sim \mathcal{G}(\frac{m_i}{2}, \frac{1}{2}\|\mathbf{y}^i - f(\theta)\|^2)$*, then the generalization is:*

$$p(\theta|\mathbf{y}^1, \ldots, \mathbf{y}^q) \propto \prod_{i=1}^{q} \|\mathbf{y}^i - f(\theta)\|^{-m_i} \tag{13}$$

Proof. Bayes' theorem and simplifications.

Bayesian updating step (4)

$$p(\theta_k|\boldsymbol{y}^1, \ldots, \boldsymbol{y}^q, \boldsymbol{u}_{0,k}) \propto \frac{1}{M}\sum_{r=1}^{M}\prod_{i=1}^{q}\|\boldsymbol{y}^i - \langle \boldsymbol{w}^{(r)}, \widehat{\boldsymbol{g}}(\boldsymbol{u}_{0,k}, \theta_k)\rangle\|^{-m_i} \qquad (14)$$

- Use Random Walk Metropolis-Hastings (RWMH) MCMC algorithm [Sullivan, 2015] to sample from (14)
- Monte-Carlo integration using sample $\{\boldsymbol{w}^{(r)}\}_{r=1}^{M}$ from the Dirichlet-1_p distribution on the simplex $\rightarrow$ integrate hyperparameter
- Test convergence of RWMH chains with Gelman-Rubin test [Gelman and Rubin, 1992]
- Updated densities are conditioned on nominal values $\boldsymbol{u}_{0,k}$ of *other $d-1$ input variables* $\rightarrow$ future work on how to integrate uncertainty
- Use log-sum-exp trick for numerical computation

Table of contents

1. Introduction
2. The BMU algorithm
3. Ensemble Kalman smoothing
4. Application to clogging of SGs in NPWRs
5. Summary
6. Appendices

Ensemble Kalman smoothing

- We don't need sequential updating at each time-step $p(g(t_k, \boldsymbol{X})|\mathcal{D})$ (which is the goal of filtering), but to assimilate the data on the entire window frame
- Indeed the data is *unavailable* at each time-step, so we need to account of all the information on a window
- The good paradigm for this is smoothing, i.e getting estimates from the full-posterior distribution $p(g(t_1, \boldsymbol{X}), \ldots, g(t_k, \boldsymbol{X})|\mathcal{D}) \rightarrow$ Ensemble Kalman smoothing (EnKS) [Evensen and van Leeuwen, 2000]

Ensemble Kalman smoothing

- Build an ensemble $\{g(\boldsymbol{X}^{(k)})\}_{k=1}^{n} \sim g\#h$, suppose for each data-group $i = 1, \ldots, q$:

$$y^{i}(t_\ell^i) = g(t_\ell^i, \boldsymbol{X}) + \eta_i, \tag{15}$$

where the variance R^i of the noise *is known*

- Define the ensemble mean and anomalies:

$$\bar{g}(t) = \frac{1}{n}\sum_{k=1}^{n} g(t, \boldsymbol{X}^{(k)}), \quad A^{(k)}(t) = g(t, \boldsymbol{X}^{(k)}) - \bar{g}(t) \tag{16}$$

- The cross-covariance between ensemble states at any time t and the observation times t_ℓ^i are approximated empirically:

$$\widehat{C}(t, t_\ell^i) = \frac{1}{n-1}\sum_{k=1}^{n} A^{(k)}(t) A^{(k)}(t_\ell^i) \tag{17}$$

Ensemble Kalman smoothing

- We define the Kalman gain for each t in the time-window:

$$K(t) = \frac{\widehat{C}(t, t_\ell^i)}{\widehat{C}(t, t_\ell^i) + R^i} \tag{18}$$

- And defining the innovation term as $d_\ell^{(k)} = y^i(t_\ell^i) - g(t_\ell^i, \boldsymbol{X}^{(k)})$, we update all the ensembles following:

$$g(\boldsymbol{X}^{(k)}) \leftarrow g(\boldsymbol{X}^{(k)}) - K_t \cdot d_\ell^{(k)} \tag{19}$$

- This process is repeated sequentially across all observation times t_ℓ^i, for all data groups i

Table of contents

1. Introduction
2. The BMU algorithm
3. Ensemble Kalman smoothing
4. Application to clogging of SGs in NPWRs
5. Summary
6. Appendices

Results

- Computer code THYC-Puffer-DEPO, complex multiphysics [Jaber et al., 2024b] , chaining of 3 codes $\rightarrow$ allows to simulate SG clogging on entire lifespan of the asset integrating past chemical cleanings and predicting future τ_c levels
- Two data groups $q = 2$, corresponding to field data and regression data

Input variable	Distribution
α	$\mathcal{U}(100, 103)$
β	$\mathcal{U}(0.02, 0.025)$
ϵ_e	$\mathcal{U}(0.2, 0.5)$
ϵ_c	$\mathcal{U}(0.01, 0.3)$
d_p	$\mathcal{U}(0.5, 10.0) \times 10^{-6}$
$\Gamma_p(0)$	$\mathcal{U}(1.0, 8.0) \times 10^{-9}$
a_v	$\mathcal{U}(0, 15) \times 10^{-4}$

Table: Probabilistic modeling of uncertain input variables

Results: posterior distributions

- $L = 3$ time windows, 5 MCMC chains are launched for GR convergence test, use uniform proposal distributions
- Computing time around 40 min $\rightarrow 5/7$ distributions are informed by the data, distinct modes for a_v

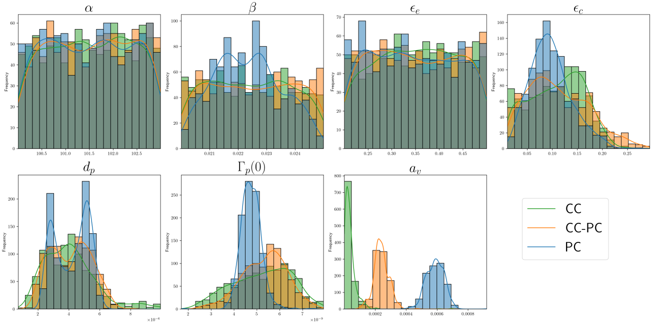


Figure: Posterior distributions of TPD clogging simulation code

Results: posterior trajectories

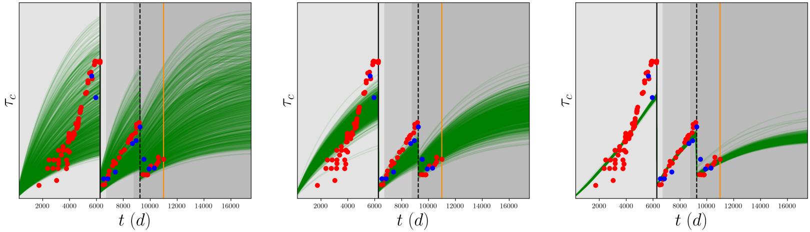
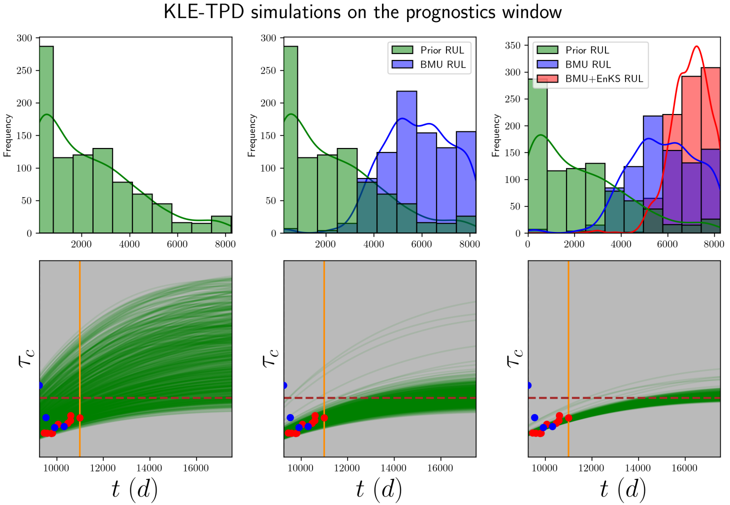


Figure: Prior/posterior and smoothed TPD emulations with Karhunen-Loève expansion surrogate

Results: posterior RUL



- RUL prediction uncertainty substantially reduced and mean of the distribution is shifted compared to the prior $\rightarrow$ positive impact for maintenance planning

Table of contents

1. Introduction
2. The BMU algorithm
3. Ensemble Kalman smoothing
4. Application to clogging of SGs in NPWRs
5. Summary
6. Appendices

Summary

- Presented an iterative algorithm leveraging kernel sensitivity analysis (HSIC) to identify individual influential variables and update priors $\rightarrow$ acting sequentially on each marginal to keep independence assumption for all dimensions $\rightarrow$ avoids cross-correlation in MCMC and curse of dimensionality
- Methodology works with a metamodeling step, enriched at each iteration by an optimization + aggregation of the metamodels on the refined DoEs to avoid bias in the posteriors
- The method integrates the noise uncertainty and works with heteroskedastic groups of data points
- Demonstrated the approach on industrial steam generator clogging, showing improved posterior inference and reduced RUL uncertainty
- Methodology is general and can be adapted to other industrial prognostics problems with scarce and heterogeneous data $\rightarrow$ GitHub repository

Some extensions and future work

- How to integrate uncertainty in nominal parameters $\boldsymbol{u}_k$?
- Prior work on adaptive conformal prediction for GP surrogate models validation [Jaber et al., 2024a] $\rightarrow$ to appear in Journal of Machine Learning for Modeling and Computing
- We define the cross-conformal estimator at a new point $\boldsymbol{X}_{n+1}$ using the posterior mean of the GP $\widetilde{g}$ and the posterior variance $\widetilde{\gamma}$:

$$\widehat{C}_{n,\alpha}^{J+GP}(\boldsymbol{X}_{n+1}) = \left[\widehat{q}_{n,\alpha}^{\pm}\left\{\widetilde{g}_{-i}(\boldsymbol{X}_{n+1}) \pm R_i^{LOO\gamma} \times \widetilde{\gamma}_{-i}(\boldsymbol{X}_{n+1})\right\}\right] \quad (20)$$

  where $R_i^{LOO\gamma}$ is the Leave-One-Out error normalized by $\widetilde{\gamma}$
- Since the intervals are adaptive, one can use it as a proxy for metamodel accuracy $\rightarrow$ perform active learning during MCMC evaluations to refine surrogates
- Available GitHub repository

Thank you for your attention!
Any questions?

Figure: Bayesian fusion

Figure: GitHub repository

Figure: CP+GP

reach me out at `edgar.jaber@ens-paris-saclay.fr`

Table of contents

1. Introduction
2. The BMU algorithm
3. Ensemble Kalman smoothing
4. Application to clogging of SGs in NPWRs
5. Summary
6. Appendices

Sensitivity analysis: HSIC

- Hilbert-Schmidt Independence Criterion (HSIC) [Gretton et al., 2005], kernel method $\rightarrow$ evaluates sensitivity of a single-input in a given-data context, no need for surrogate models
- Theoretical result for all $i \in \{1, \ldots, d\}, k \in \{1, \ldots, N\}$:

$$\text{HSIC}(X_i, g(\boldsymbol{X}, t_k)) = 0 \iff X_i \perp g(\boldsymbol{X}, t_k) \tag{21}$$

- The index disposes of U-stat and V-stat estimators $+$ hypothesis testing with corresponding p-value $\rightarrow$ implemented in the OpenTURNS
- The normalized R^2_{HSIC} index is better suited for interpretation:

$$R^2_{\text{HSIC}}(X_i, g(\boldsymbol{X}, t_k)) = \frac{\text{HSIC}(X_i, g(\boldsymbol{X}, t_k))}{\sqrt{(\text{HSIC}(X_i, X_i)\text{HSIC}(g(\boldsymbol{X}, t_k), g(\boldsymbol{X}, t_k)))}} \in [0, 1]$$

- Empirical evidence suggests that R^2_{HSIC} can be used confidently for variable ranking $\rightarrow$ HSIC-ANOVA decompositions also exist *but* only pathological cases create stark differences (see [Sarazin et al., 2022])

References I

Biggio, L. and Kastanis, B. (2020). Prognostics and health management of industrial assets: Current progress and road ahead. *Frontiers in Artificial Intelligence*, 3:578613.

Evensen, G. and van Leeuwen, P. J. (2000). An ensemble Kalman smoother for nonlinear dynamics. *Monthly Weather Review*, 128:1852–1867.

Feng, Q., Nebes, J., Bachet, M., Pujet, S., You, D., and Deri, E. (2023). Tube support plates blockage of PWR steam generators: thermalhydraulics and chemical modeling. Proceedings of the International Conference on Water Chemistry in Nuclear Reactor Systems (NPC 2023), Juan-les-Pins.

Geir Evensen, Femke C. Vossepoel, P. J. v. L. (2022). *Data assimilation fundamentals*. Springer Cham.

Gelman, A. and Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457 – 472.

References II

Girard, S. (2014). *Clogging of recirculating nuclear steam generators*. Springer International Publishing.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. (2005). Measuring statistical dependence with Hilbert-Schmidt norms. In *Proceedings Algorithmic Learning Theory*, pages 63–77. Springer-Verlag.

Jaber, E., Blot, V., Brunel, N., Chabridon, V., Remy, E., Iooss, B., Lucor, D., Mougeot, M., and Leite, A. (2024a). Conformal Approach To Gaussian Process Surrogate Evaluation With Coverage Guarantees.

Jaber, E., Chabridon, V., Remy, E., Baudin, M., Lucor, D., Mougeot, M., and Iooss, B. (2024b). Sensitivity analyses of a multi-physics long-term clogging model for steam generators. *International Journal for Uncertainty Quantification*.

Pinciroli, L., Baraldi, P., Shokry, A., Zio, E., Seraoui, R., and Mai, C. (2021). A semi-supervised method for the characterization of degradation of nuclear power plants steam generators. *Progress in Nuclear Energy*, 131:103580.

References III

Prusek, T., Moleiro, E., Oukacine, F., Adobes, A., Jaeger, M., and Grandotto, M. (2013). Deposit models for tube support plate flow blockage in steam generators. *Nuclear Engineering and Design*, 262:418–428.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press, Massachusetts.

Sarazin, G., Marrel, A., Da Veiga, S., and Chabridon, V. (2022). Test d'indépendance basé sur les indices HSIC-ANOVA d'ordre total. Proceedings des 53èmes Journées de Statistique de la SFdS.

Srikantiah, G. and Chappidi, P. (2000). Particle deposition and fouling in PWR steam generators. *Nuclear Engineering and Design*, 200(1-2):285–294.

Sullivan, T. J. (2015). *Introduction to uncertainty quantification*. Texts in Applied Mathematics, 63. Springer-Verlag.

References IV

Vaibhav, Y. and al. (2023). Technical challenges and gaps in integration of advanced sensors, instrumentation, and communication technologies with digital twins for nuclear application. Oak Ridge National Laboratory (ORNL), Oak Ridge, TN (United States).

Yang, G., Pointeau, V., Tevissen, E., and Chagnes, A. (2017). A review on clogging of recirculating steam generators in pressurized-water reactors. *Progress in Nuclear Energy*, 97:182–196.