Hybrid prognostics using simulation codes and statistical models – Application to the study of steam generators clogging

Edgar JABER - PhD Defense - 09/02/2026

Examinators:
J. GARNIER
P. M. CONGEDO
C. SCHILLINGS
E. VAZQUEZ

Reviewers:
E. CHATZI
B. LAURENT-BONNEAU

Supervisors:
M. MOUGEOT
D. LUCOR
E. REMY
V. CHABRIDON

Outline

Clogging of steam generators (SGs)

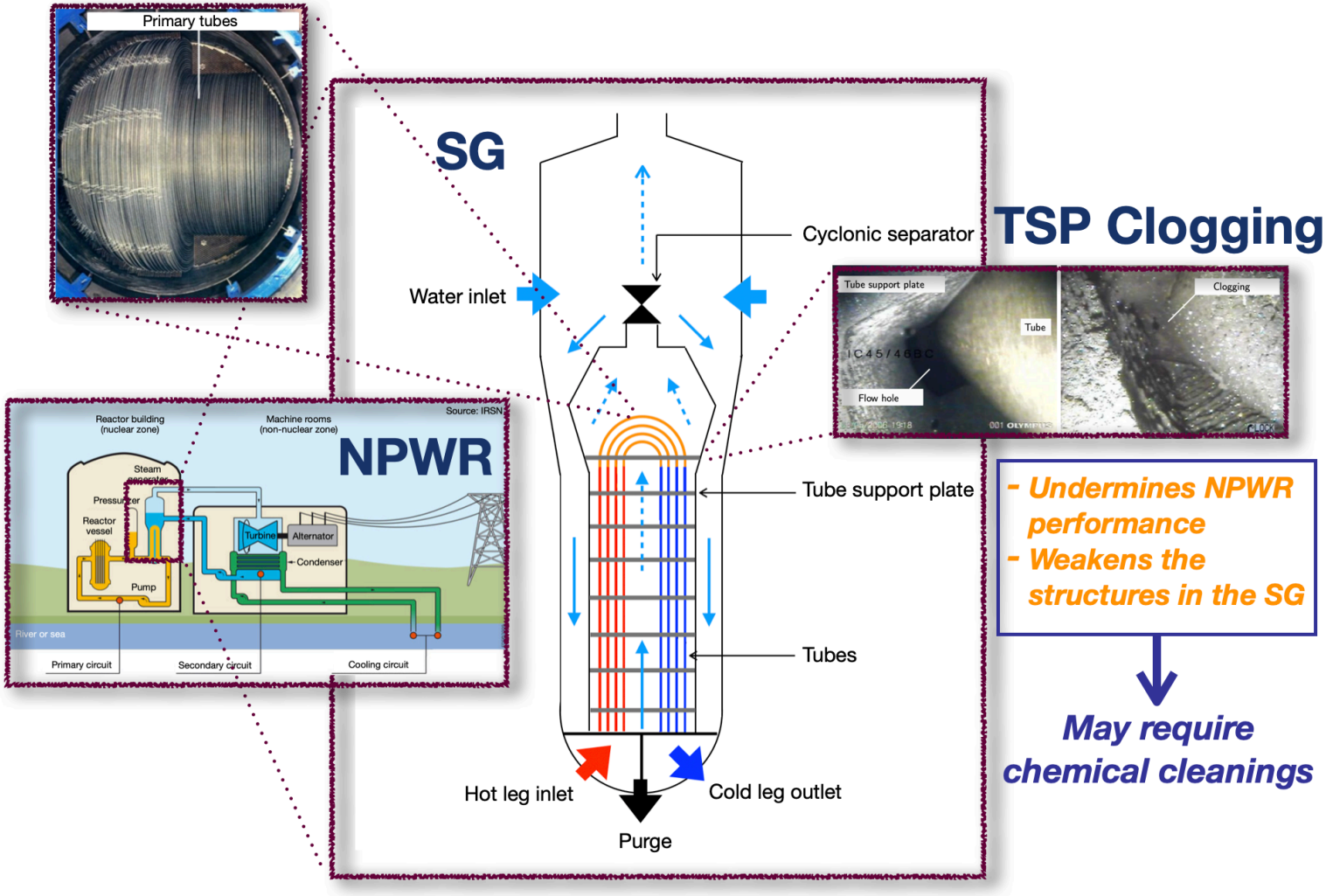


Figure 1: Nuclear pressurized water reactor (NPWR) scheme, an SG and example of video examination (TVE) of a tube support plate (TSP) during a NPWR outage (© IRSN, EDF)

Notions of prognostics

- EDF Engineers may have to schedule SG chemical cleanings
- Degradation level of the system $(t \mapsto \Delta(t))$ → predict the remaining useful life (RUL) [Vachtsevanos et al., 2006] for a fixed threshold $\Delta_* \in \mathbb{R}_+$:

$$\text{RUL}(\Delta_*) = \underset{t > t_P}{\text{argmin}}\{\Delta(t) \geq \Delta_*\} \tag{1}$$

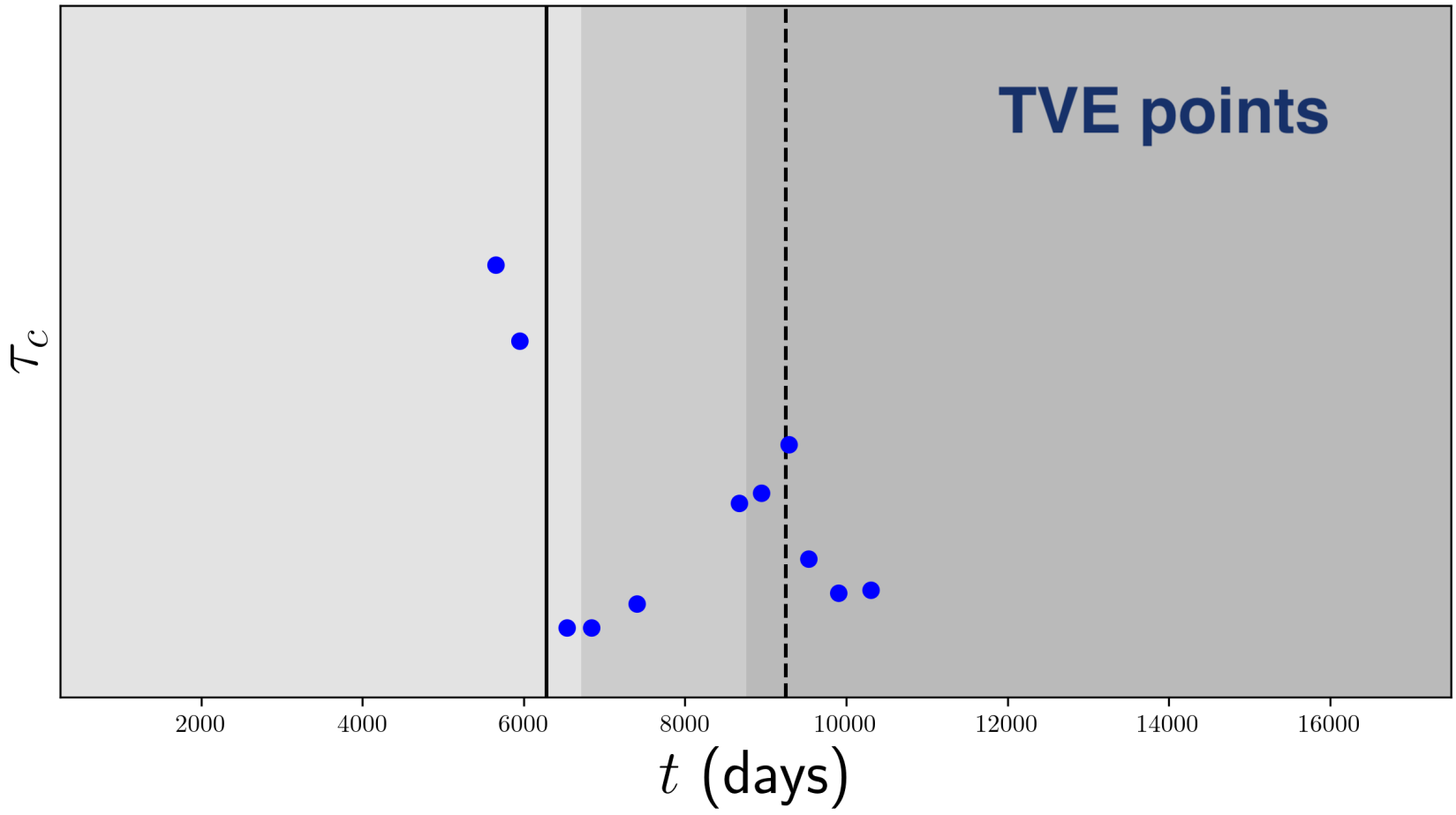
  where t_P is the present time.

  → Relies usually on physics-based simulation codes, and/or data driven methods

- RUL prediction with each individual tool could lack robustness, especially for complex operating systems
- For clogging, $\Delta = \tau_c$

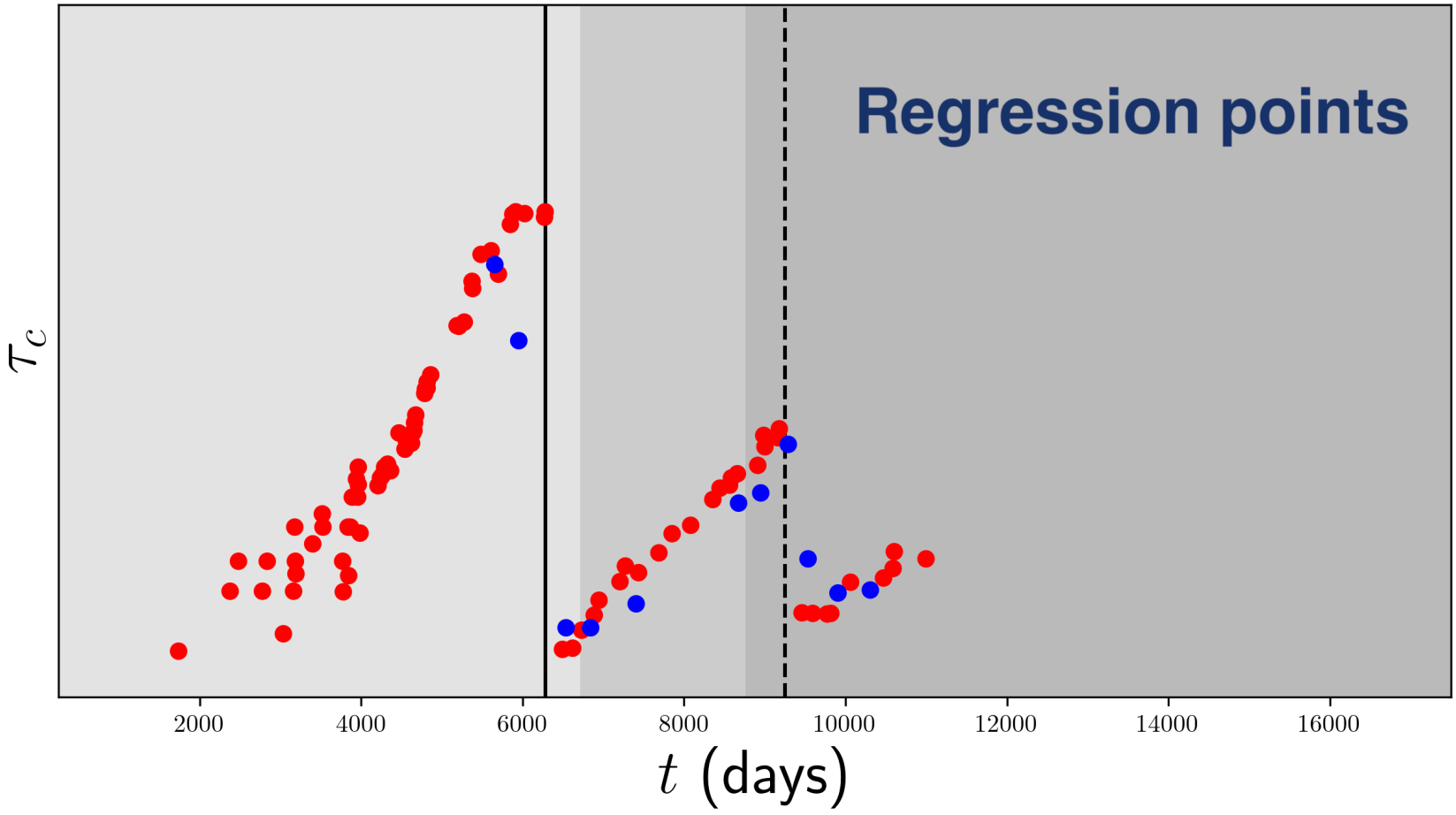
  Idea: *build a hybrid framework for evaluating the SG clogging RUL using the physics-based model and the data-driven models*

Available clogging-related informations



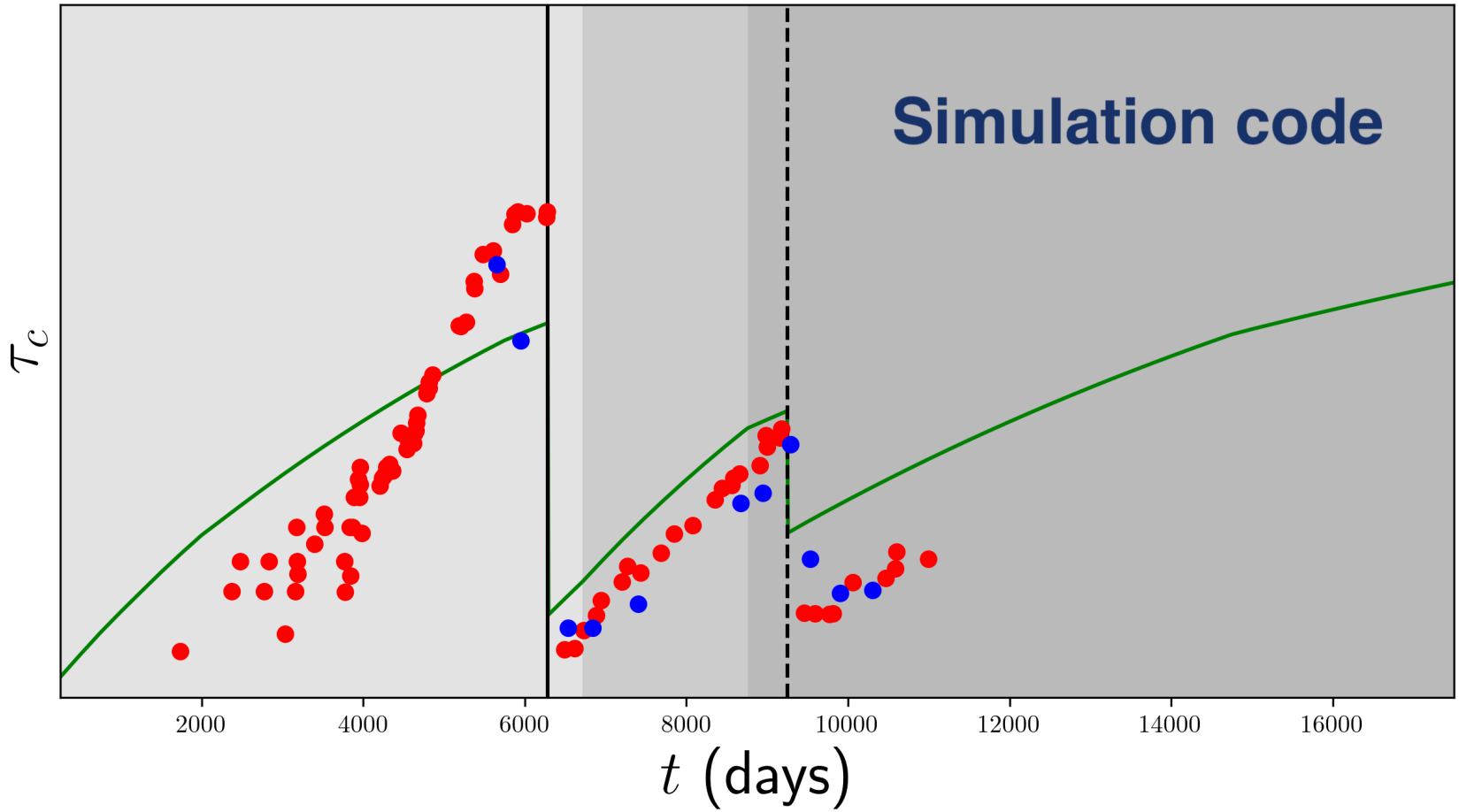
- *Scarce* televised video examinations (TVEs) monitoring data

Available clogging-related informations



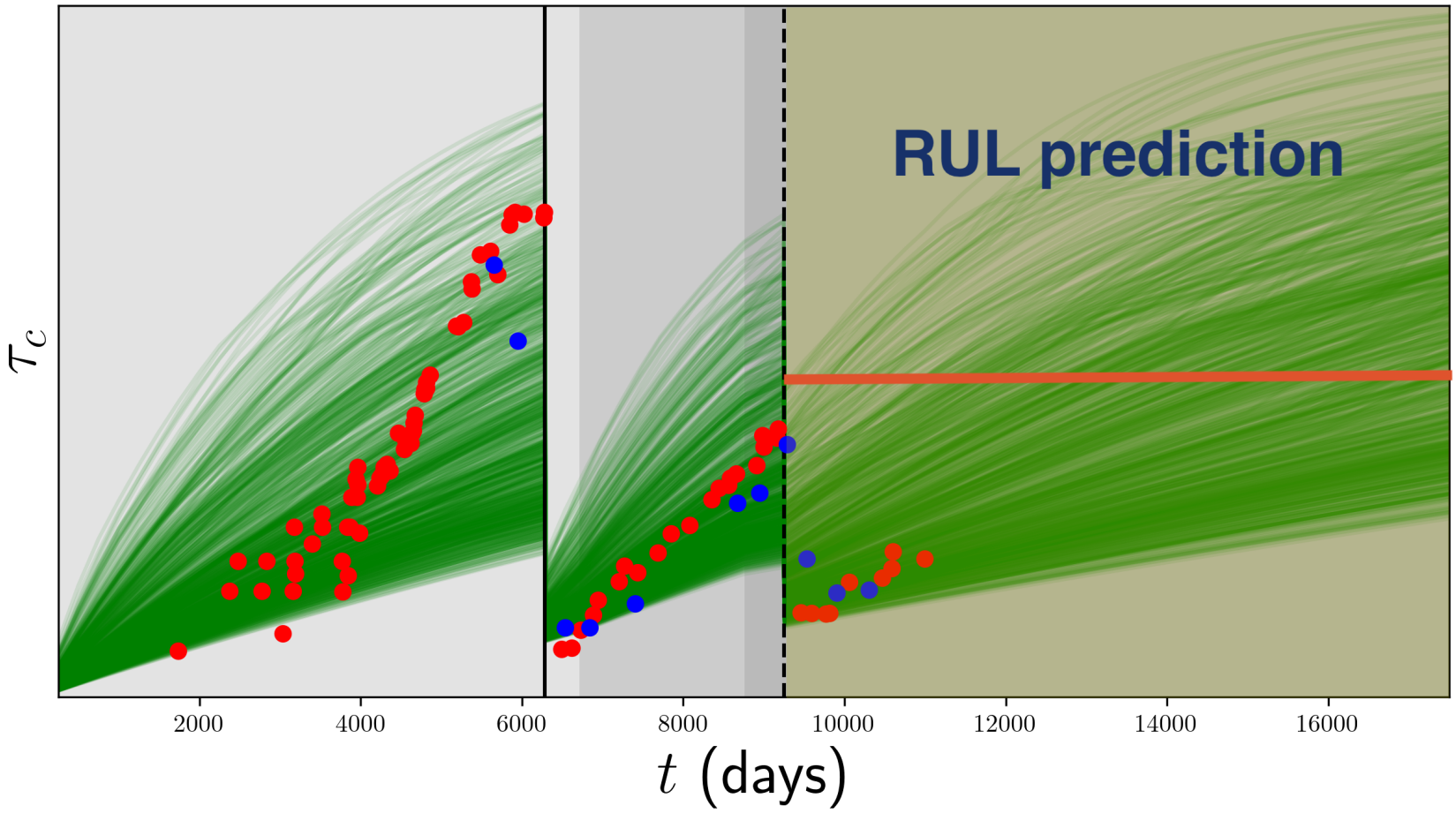
- Data-driven regression algorithms [Pinciroli et al., 2021] based on the TVEs

Available clogging-related informations



- Clogging physics model developed by [Prusek et al., 2013]
- Subsequent numerical simulation code THYC-Puffer-DEPO [Feng et al., 2023]

Available clogging-related informations



How to make use of the available knowledge for achieving reliable RUL predictions?

Digital twin for a SG

- Use of a digital twin (DT) [AIAA, 2021] platform for predicting the state of health of a steam generator
- Enabling informed decision-making using all available asset information

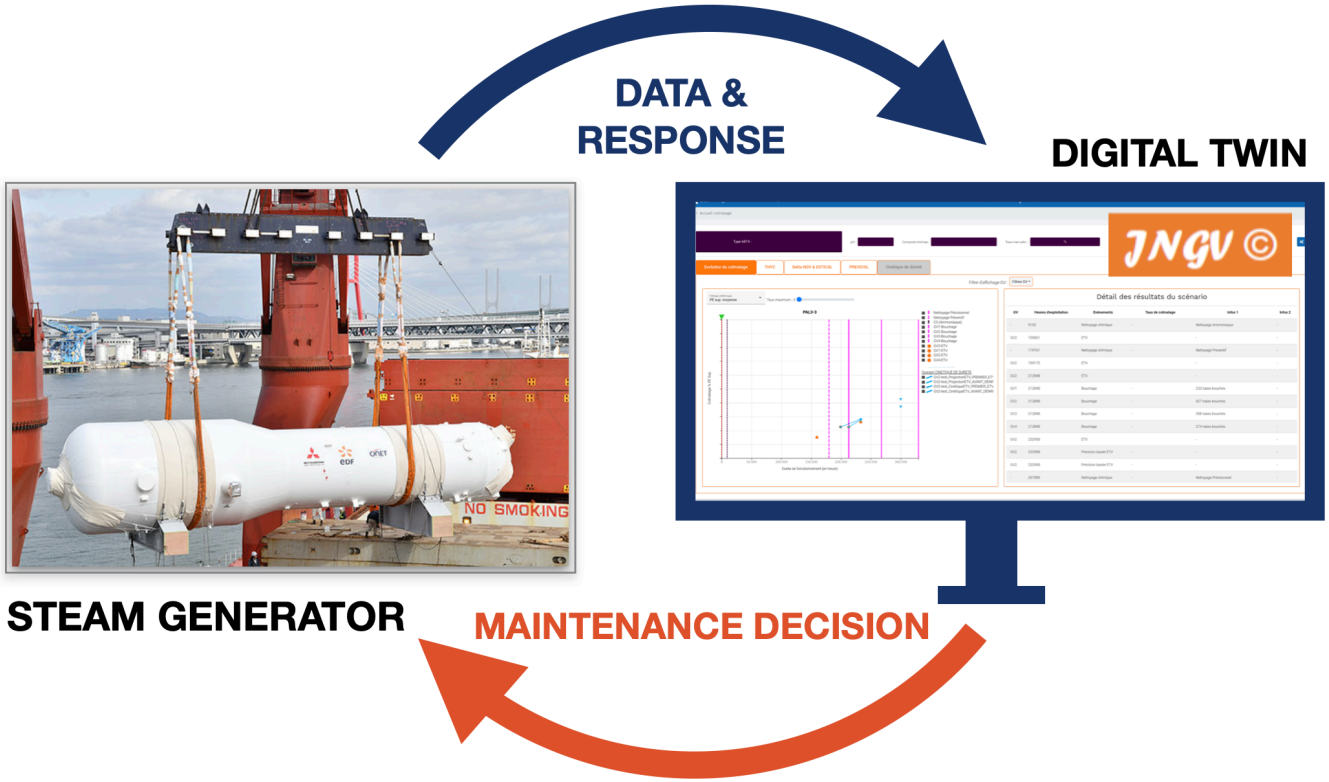


Figure 2: DT architecture for a SG [Deri et al., 2021] (”Jumeau Numérique Générateur de Vapeur” - JNGV)

Hybrid framework for clogging prognostics

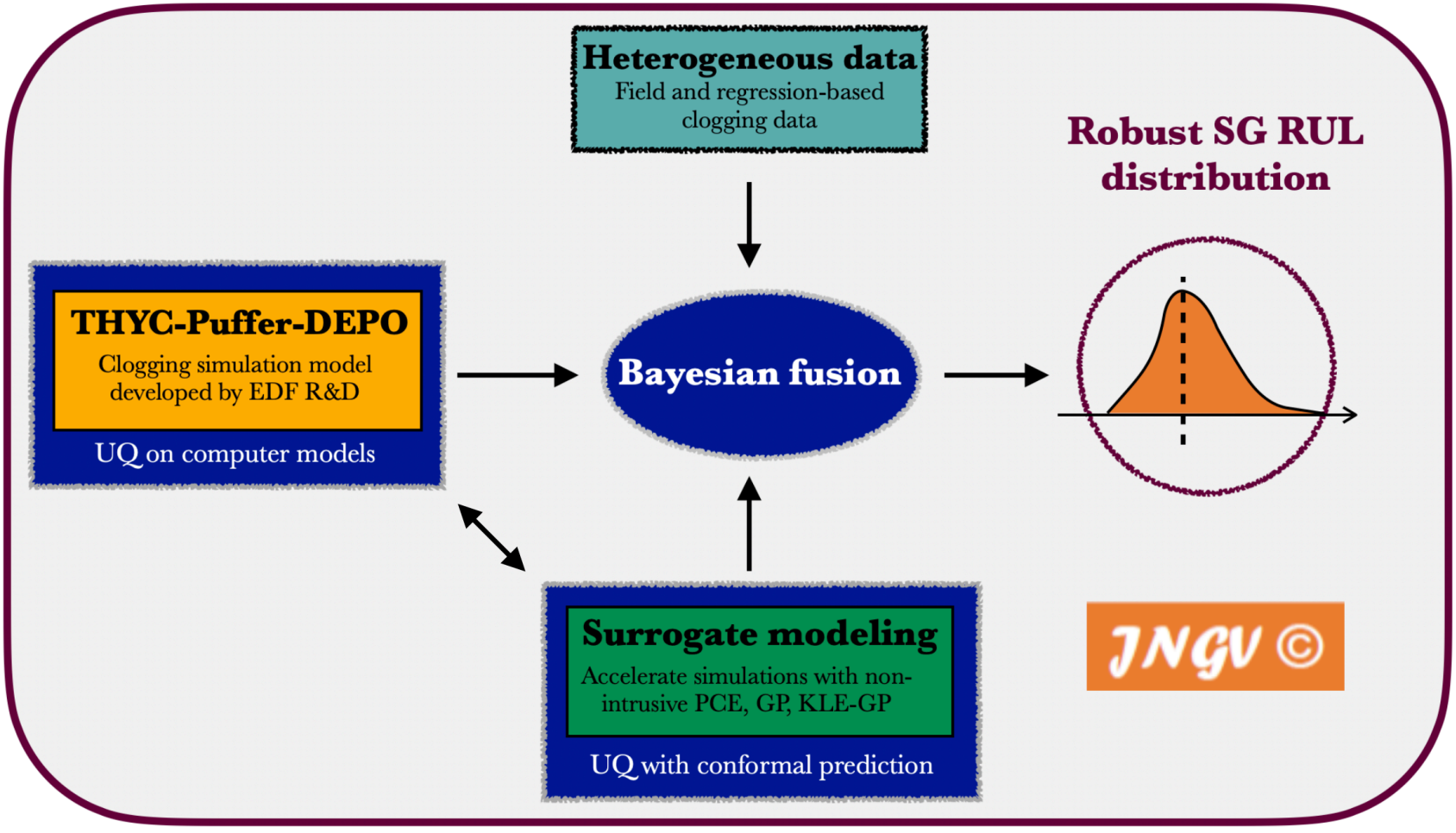


Figure 3: Hybrid framework for a specific SG clogging prognostics

DT integration for clogging prognostics

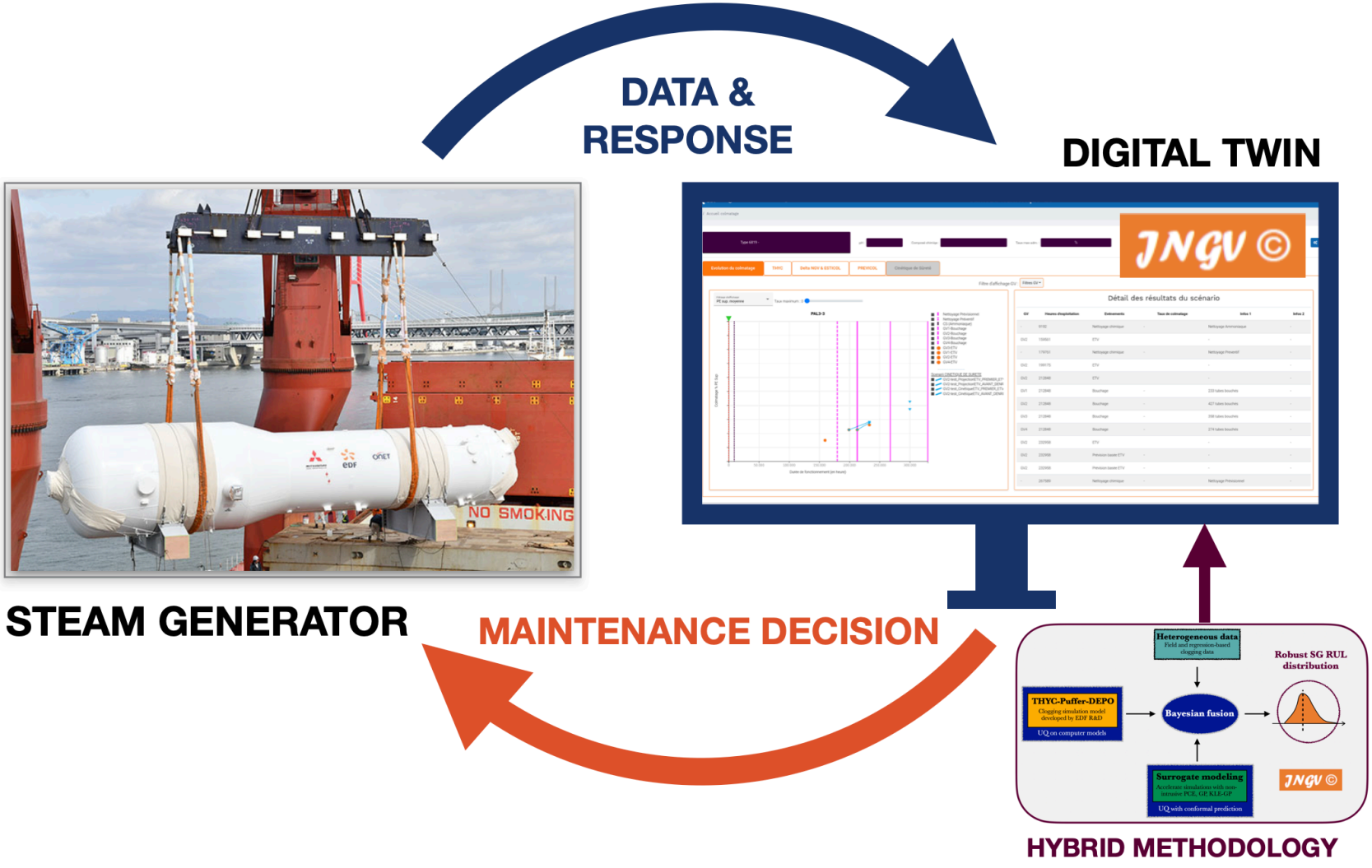


Figure 4: Integration of the hybrid framework in the JNGV

Summary of the introduction

- Available tools: clogging simulation code, field data, regression data, EDF expert and field knowledge
- Objective: estimate as accurately as possible the probabilistic RUL for SG clogging by developing a hybrid framework involving all the available tools
- Challenges: complex physics,heavy simulation code, inherent uncertainty, operational variability, field data scarce both in time and space, different degrees of data fidelity, costly simulation code

Proposed solution → ***Bayesian fusion methodology based on the clogging simulation code, its surrogate models and the heterogeneous data***

Outline

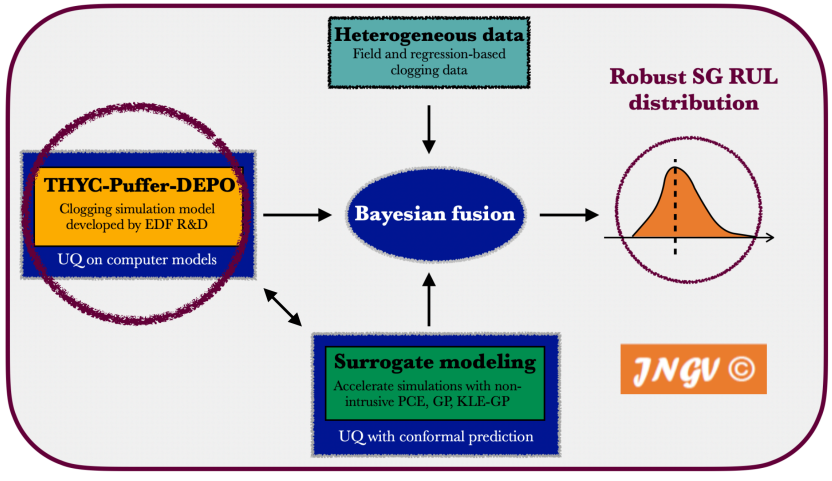
Hybrid framework for the JNGV

Objectives:

- Evaluate uncertainty on EDF clogging simulation code
- Extend the preliminary work of [Lefebvre et al., 2023]
- Obtain a first RUL distribution based on the physical tool

Methodology:

- *State of the art:*
  - → Uncertainty analysis (UA) of industrial simulation codes [De Rocquigny et al., 2008]
  - → High-performance computing
  - → Surrogate models, sensitivity analysis
- *Strategy:* Couple the EDF code with the OpenTURNS UQ Python module and compute different sensitivity analysis indices



Clogging physical model

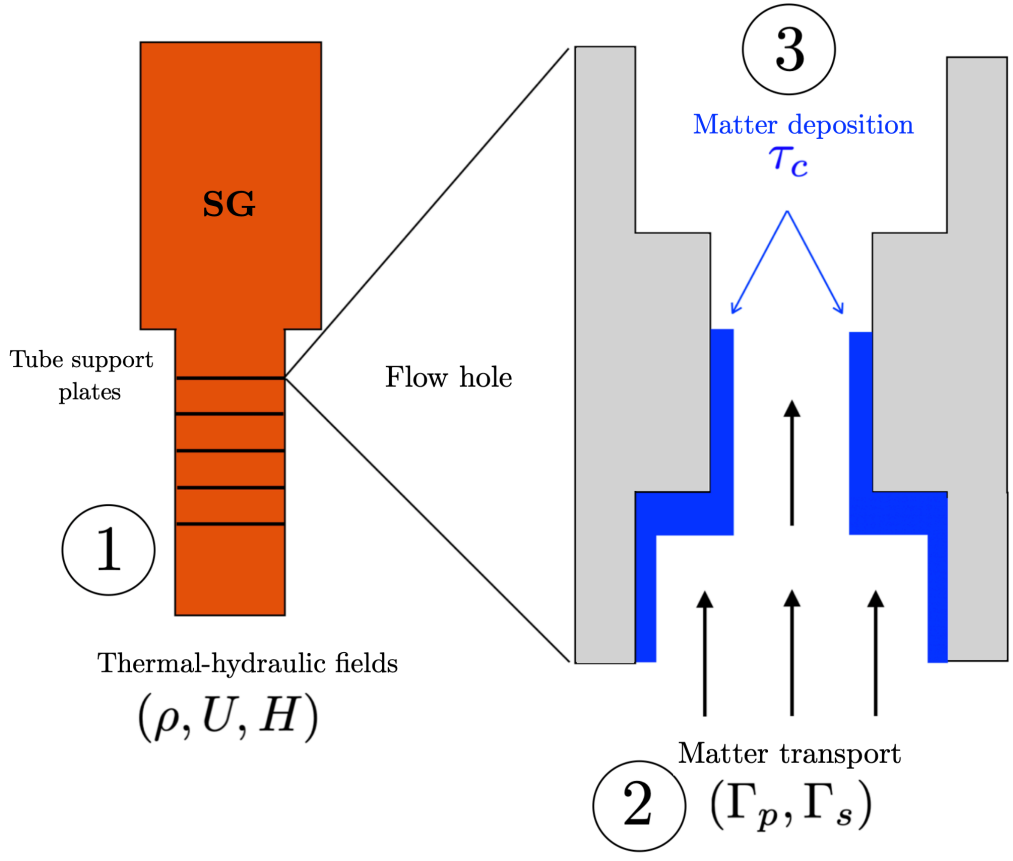


Figure 5: Clogging physical model

- Clogging results from two main mechanisms → *vena contracta* and *flashing* [Prusek et al., 2013]
- Long-term clogging model [Feng et al., 2023] → must change stationary thermohydraulics + compute chemical conditioning

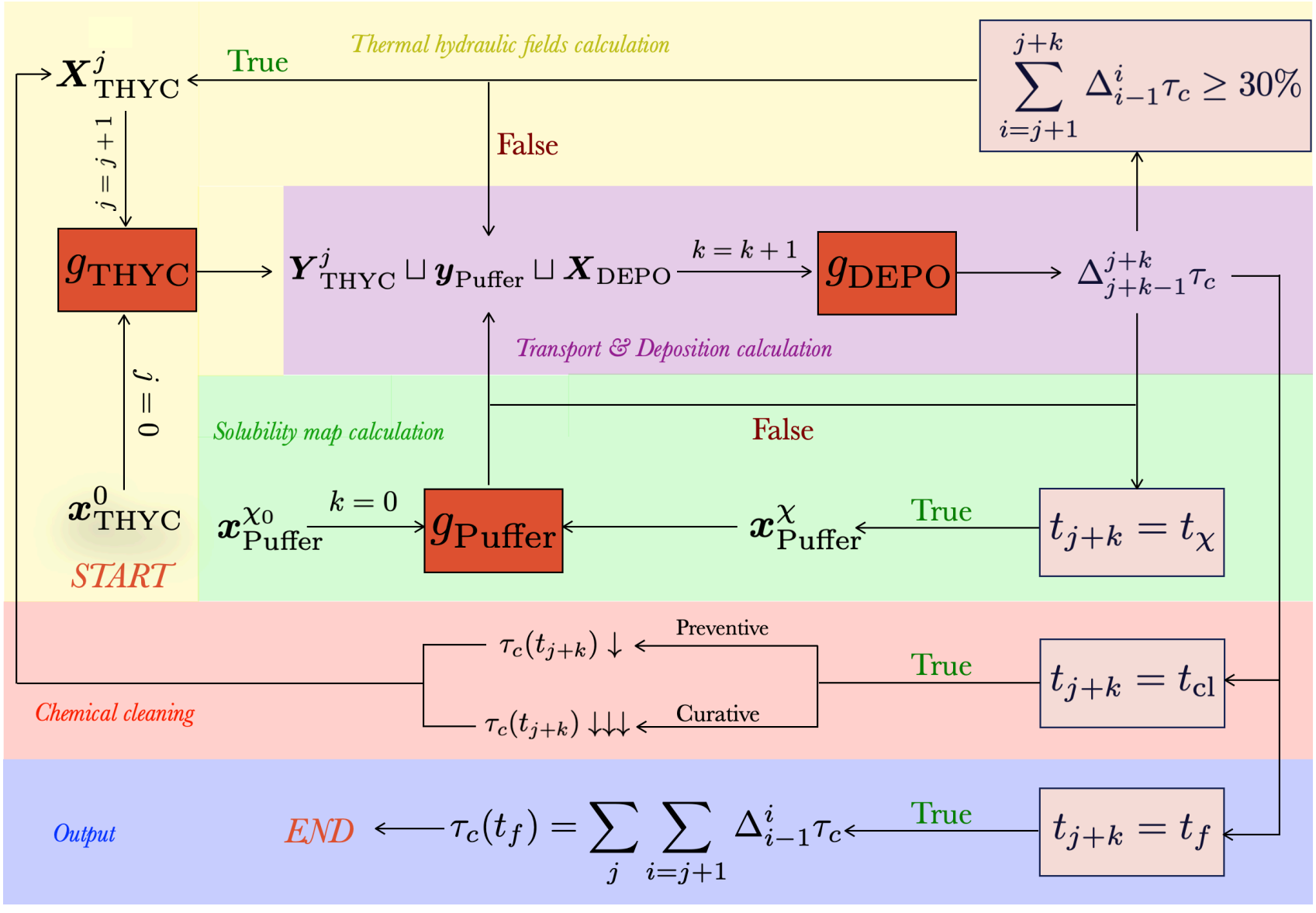
Clogging computational model

- THYC-Puffer-DEPO (TPD) [Feng et al., 2023] is the chaining of 3 codes:
  - → THYC [David, 1999] is based on a finite-volume numerical scheme for the two-phase conservation equations
  - → Puffer is an in-house chemical code allowing to compute the solubility of iron oxides as a function of pH
  - → DEPO [Prusek et al., 2013] is the deposit module, solving the transport and clogging equations with iterative finite-differences schemes methods
- Allows to simulate SG clogging on entire lifespan of the asset integrating past chemical cleanings and predicting future τ_c states → Unitary call is $\sim$ 5h on HPC infrastructure

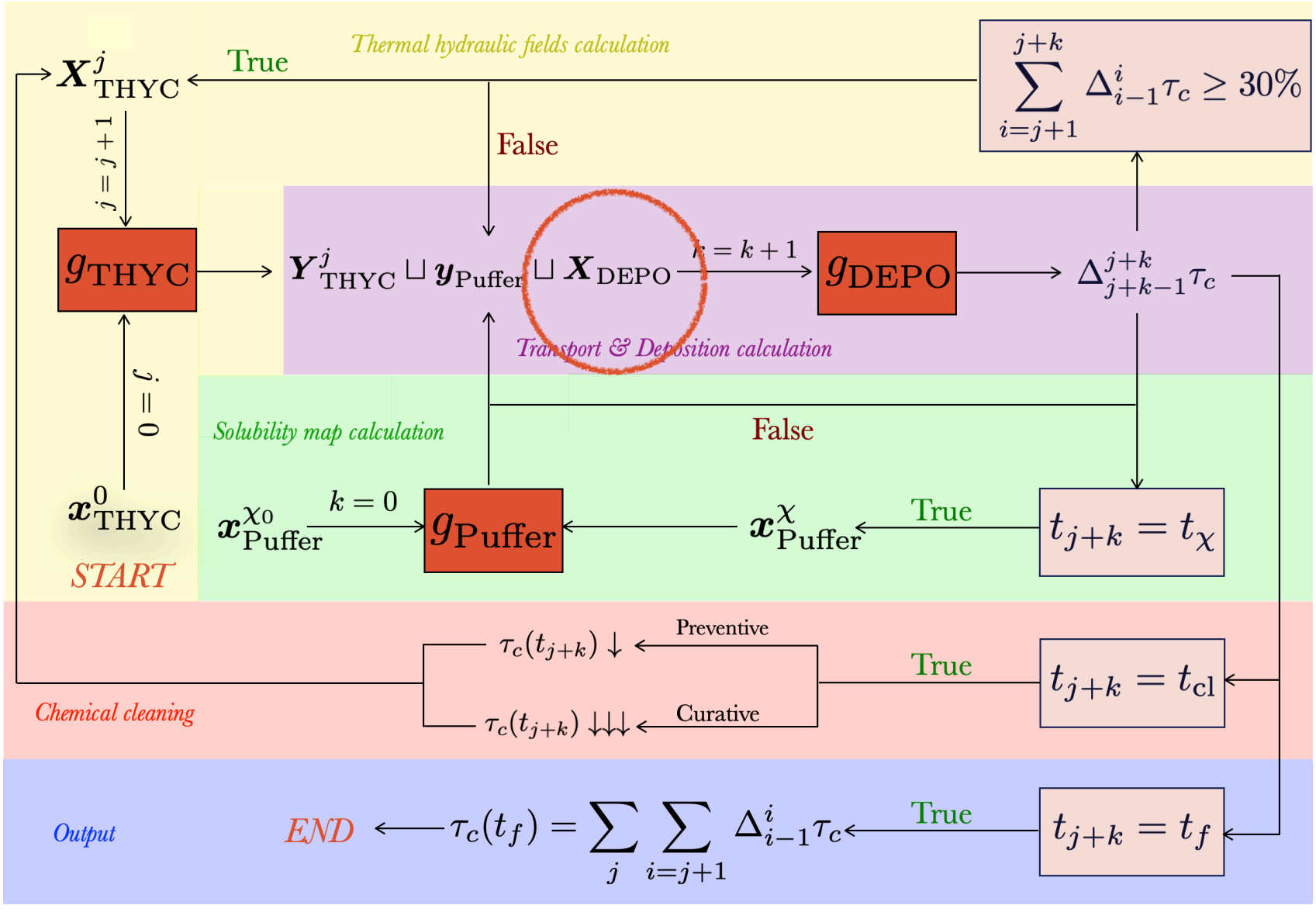
We denote it as a function $g_{\text{TPD}} =: g : \mathcal{X} \subset \mathbb{R}^d \to \mathbb{R}^N$ with $N \simeq 10^2$, TPD simulates a trajectory of τ_c for each input $\boldsymbol{X}$,

$$g(\boldsymbol{X}) = (g(t_1, \boldsymbol{X}), \dots g(t_N, \boldsymbol{X})) \in \mathbb{R}^N \tag{2}$$

TPD simulation chain



TPD simulation chain



Initial Design of Experiments (DoE)

- $d = 7$ input variables of the deposit module g_{DEPO} with uncertainty:

$$\boldsymbol{X}_{\text{DEPO}} =: \boldsymbol{X} = (\alpha, \beta, \epsilon_e, \epsilon_c, d_p, \Gamma_p(0), a_v) \sim p_{\boldsymbol{X}} = \otimes_{i=1}^{d} p_{X_i},$$

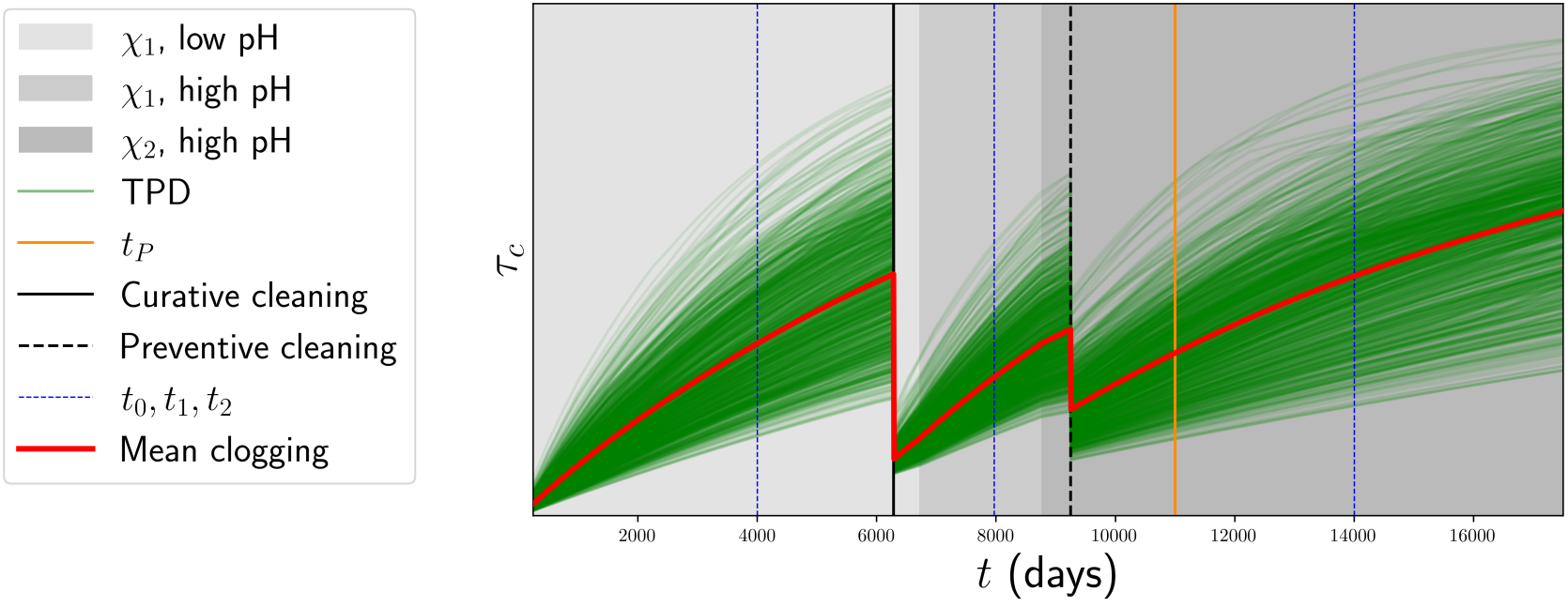
with supports and distributions provided and validated [Lefebvre et al., 2023]

- $n = 1000$ i.i.d. Monte Carlo samples drawn according to the distributions below, $\text{DoE}_g^{p_{\boldsymbol{X}}} = \{(\boldsymbol{X}^{(i)}, g(\boldsymbol{X}^{(i)}))\}_{i=1}^{n}$

Variable	Physical meaning	Distribution
α	First empirical correlation parameter	$\mathcal{N}(101.6, 4.0)$
β	Second empirical correlation parameter	$\mathcal{N}(0.0233, 0.0005)$
ϵ_e	Porosity of fouling deposits	$\mathcal{T}(0.2, 0.3, 0.5)$
ϵ_c	Porosity of clogging deposits	$\mathcal{T}(0.01, 0.05, 0.3)$
d_p	Diameter of iron oxide particles (m)	$\mathcal{T}(0.5, 5.0, 10.0) \times 10^{-6}$
$\Gamma_p(0)$	Initial solid mass fraction	$\mathcal{T}(1.0, 4.5, 8.0) \times 10^{-9}$
a_v	Calibration parameter	$\mathcal{T}(0.1, 7.8, 12) \times 10^{-4}$

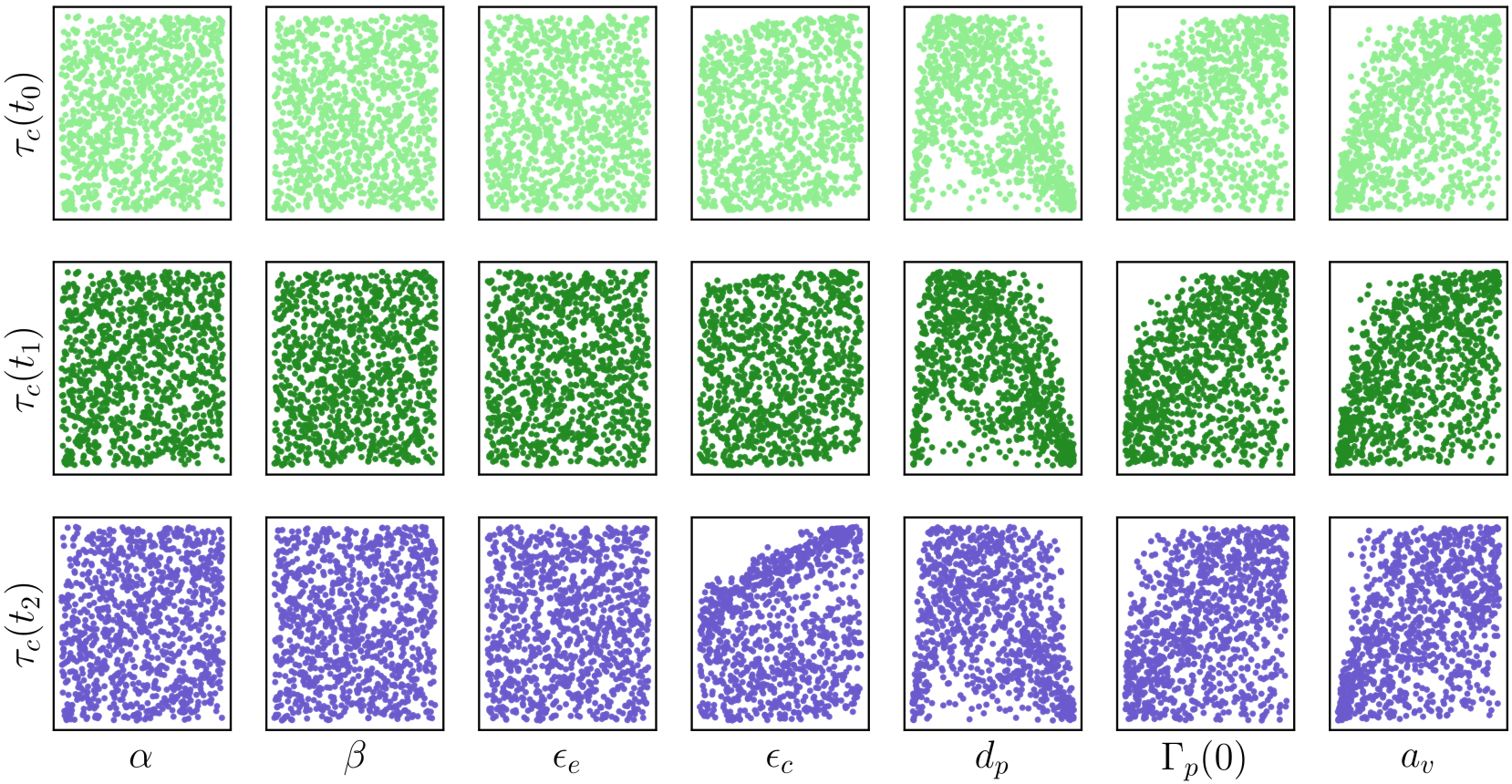
Table 1: Uncertain input variables, physical meaning and distribution

Output results of the uncertainty propagation



- High-level of uncertainty observed throughout the simulation times + very high uncertainty on RUL window

Scatter-plots in rank space



- Linear correlations of $\Gamma_p(0)$ and a_v during all simulation times
- Non-linear correlations of d_p → confirms findings in [Lefebvre et al., 2023], novel finding: non-linear correlation of ϵ_c that evolves in time

Given-data sensitivity analysis with HSIC

- Hilbert-Schmidt Independence Criterion (HSIC) [Gretton et al., 2005], kernel method
  - → Evaluates sensitivity of a single-input in a given-data context
- Theoretical result for all $i \in \{1, \dots, d\}, k \in \{1, \dots, N\}$:

$$\text{HSIC}(X_i, g(t_k, \boldsymbol{X})) = 0 \iff X_i \text{ and } g(t_k, \boldsymbol{X}) \text{ are independent} \quad (3)$$

- Index disposes of U-stat and V-stat estimators [Da Veiga, 2015] + hypothesis testing with corresponding p-value
- Normalized R^2_{HSIC} index is better suited for interpretation:

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

Results

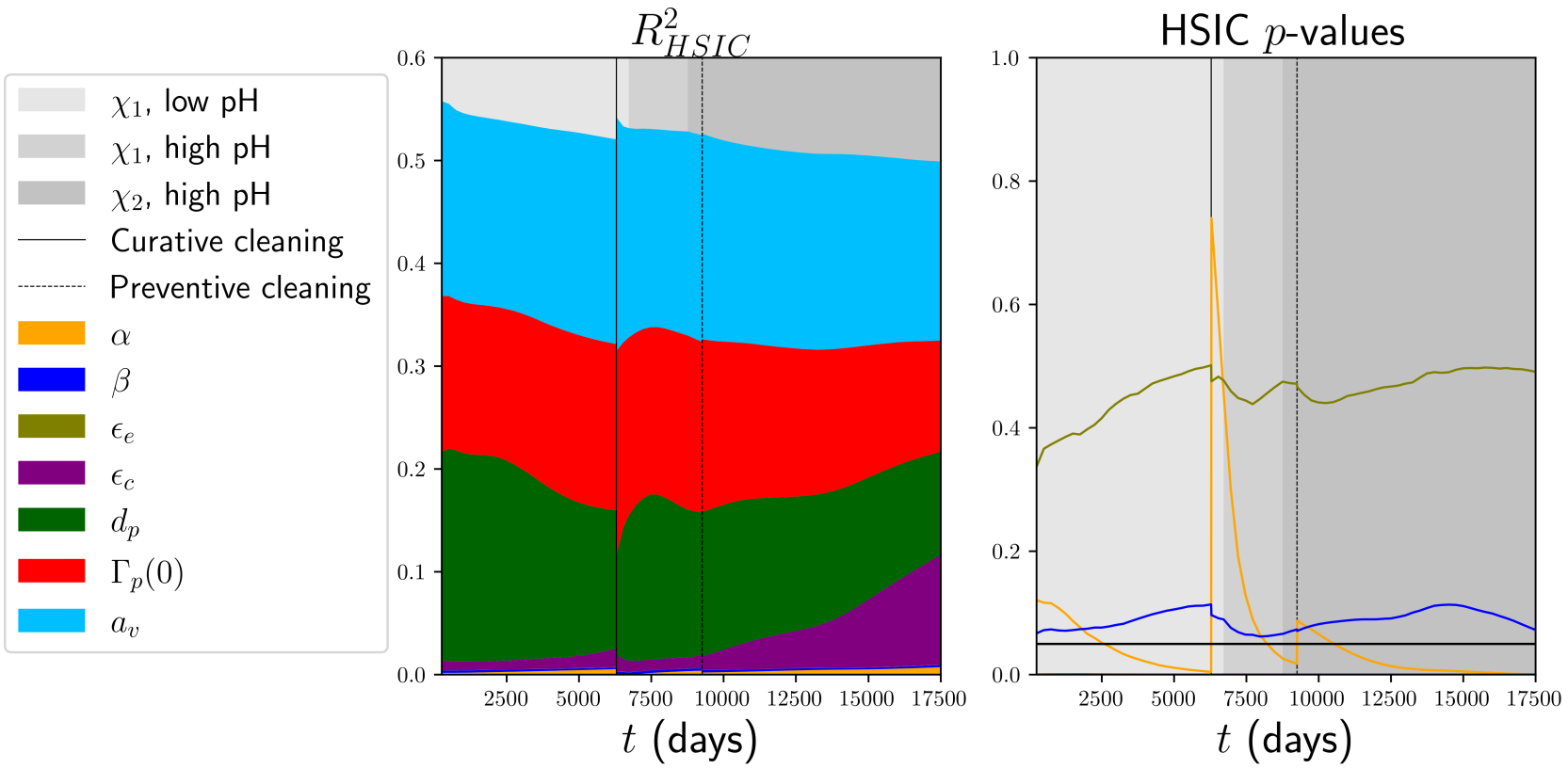


Figure 6: Time variation of the estimated R^2_{HSIC} indices and associated p-values

Summary of part 2.

Take-home messages:

- Uncertainty analysis on TPD with expert-informed p_X reveals a high level of uncertainty in prognostics
- Sensitivity analysis results (both HSIC and Sobol') are robust with the preliminary work [Lefebvre et al., 2023]
- New findings related to the influence of clogging porosity ϵ_c in high pH paves the way for future clogging modeling

Contributions:

- Paper [Jaber et al., 2025b] published in the **International Journal of Uncertainty Quantification**
- `Github repository` with scripts based on the OpenTURNS Python module [Baudin et al., 2017]
- Poster presentation at MASCOT-NUM 2023

Outline

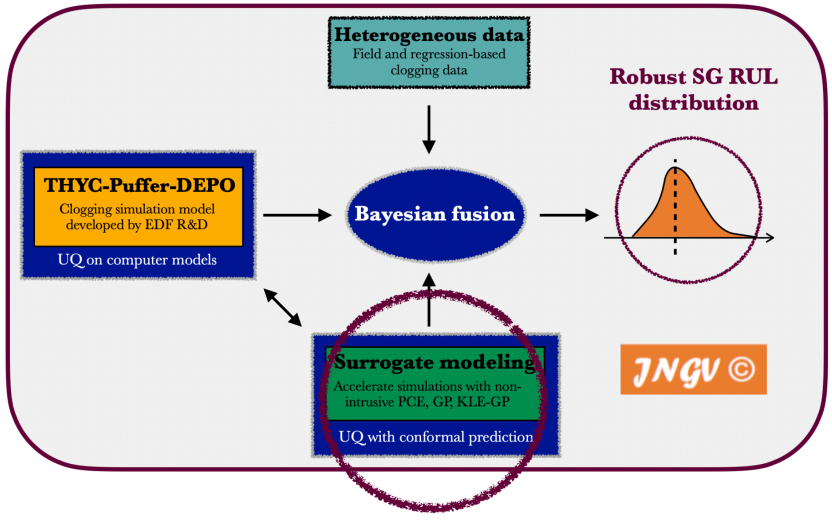
Hybrid framework for the JNGV

Objectives:

- Obtain fast evaluations of TPD trajectories for downstream tasks
- Ensure the validity of these evaluations with proven methodologies

Methodology:

- *State of the art:*
  - → Vector polynomial chaos expansions, Gaussian processes, Reduced order models
  - → Predictivity coefficients, Bayesian credibility intervals [Acharki et al., 2023]
- *Strategy:* Build a variety of different surrogate models, develop a validation methodology for Gaussian processes based on conformal prediction



Surrogate modeling

- General idea is to create a model $\widehat{g} \approx g$ that enables faster evaluations of the computer simulation model [Sullivan, 2015]
  - → Purpose is to sample rapidly from the *solution manifold*:

$$\mathcal{M}_{\boldsymbol{X}} = \{g(\boldsymbol{X}),\ \boldsymbol{X} \sim p_{\boldsymbol{X}} \in \mathcal{P}(\boldsymbol{X})\} \tag{4}$$

- Depending on the nature of the computer code g and its outputs, one can adopt an:
  - → *Intrusive* approach: modifying the source-code g (e.g. Reduced basis methods) [Le Maître and Knio, 2010] → approximation guarantees for many families of problems
  - → *Non-intrusive* approach [De Rocquigny et al., 2008]: using methods similar to supervised learning techniques → more useful in industry where there are legacy codes
- Problem specific and task-oriented → method should require validity in well chosen scenarios, specific validation methodologies have to be developed

Stochastic time-collocation DoE

- Augment N to $D \gg N$ post-simulations with the help of linear interpolators $\mathcal{G} : \mathbb{R}^N \to \mathbb{R}^D$ (clogging is strictly increasing) to get:

$$\mathcal{G}(g(\boldsymbol{X})) = \left(\sum_{k=1}^{N} g(t_k, \boldsymbol{X}) \varphi_k(t_1), \ldots, \sum_{k=1}^{N} g(t_k, \boldsymbol{X}) \varphi_k(t_D) \right) =: Z(\boldsymbol{X}) \in \mathbb{R}^D, \tag{5}$$

  where φ_k are linear interpolators

- Choose a uniform probability density on the time-indices and draw independent samples:

$$(t, \boldsymbol{X}) \sim \mathcal{U}\{1, \ldots, D\} \otimes p_{\boldsymbol{X}}$$

- Apply to all $i \in \{1, \ldots, n\}$, the following strategy:

$$(t^{(i)}, \boldsymbol{X}^{(i)}) \xrightarrow[\text{interpolation}]{\mathcal{L}} (\boldsymbol{X}^{(i)}, \mathcal{G}(g(\boldsymbol{X}^{(i)}))) \xrightarrow[\text{projection}]{\text{pr}_i} ((t^{(i)}, \boldsymbol{X}^{(i)}), \text{pr}_i \circ Z(\boldsymbol{X}^{(i)}))$$

Gaussian Process (GP) surrogate

- DoE with a *scalar output* $\mathcal{D}_n = \{((t^{(i)}, \boldsymbol{X}^{(i)}), \mathrm{pr}_i \circ Z(\boldsymbol{X}^{(i)}))\}_{i=1}^n$, construct a regular scalar GP [Rasmussen and Williams, 2006]
  → split $\mathcal{D}_{n_{\text{train}}} \cup \mathcal{D}_{n_{\text{test}}}$
- We choose a prior $\mathcal{G} \sim \mathcal{GP}(m_\beta, \gamma_\varphi)$ and compute and optimize the conditioned $\widetilde{\mathcal{G}} = \mathcal{G} \mid \mathcal{D}_{n_{\text{train}}} \sim \mathcal{GP}(m_{\beta^*}, \gamma_{\varphi^*})$ using maximum likelihood estimator for the prior hyperparameters $\theta = (\beta, \varphi)$
- GP metamodel of TPD is then estimated by parametrizing the first marginal of the GP of Z, to the time indices of TPD, in other words:

$$\widehat{g}(\boldsymbol{X}) := (m_{\beta^*}(t_1, \boldsymbol{X}), \ldots, m_{\beta^*}(t_N, \boldsymbol{X})) \in \mathbb{R}^N \tag{6}$$

- Validation method: compute the predictivity coefficient on $\mathcal{D}_{\text{test}}$ at each time step $Q^2(t_k)$, and then average it:

$$Q^2(t_k) = 1 - \sum_{j=1}^{n_{\text{test}}} \frac{\left| g(t_k, \boldsymbol{X}^{(j)}) - m_{\beta^*}(t_k, \boldsymbol{X}^{(j)}) \right|^2}{\widehat{\mathrm{Var}}(g(t_k, .))}, \quad \overline{Q^2} = \frac{1}{N} \sum_{k=1}^{N} Q^2(t_k) \tag{7}$$

Results for different priors

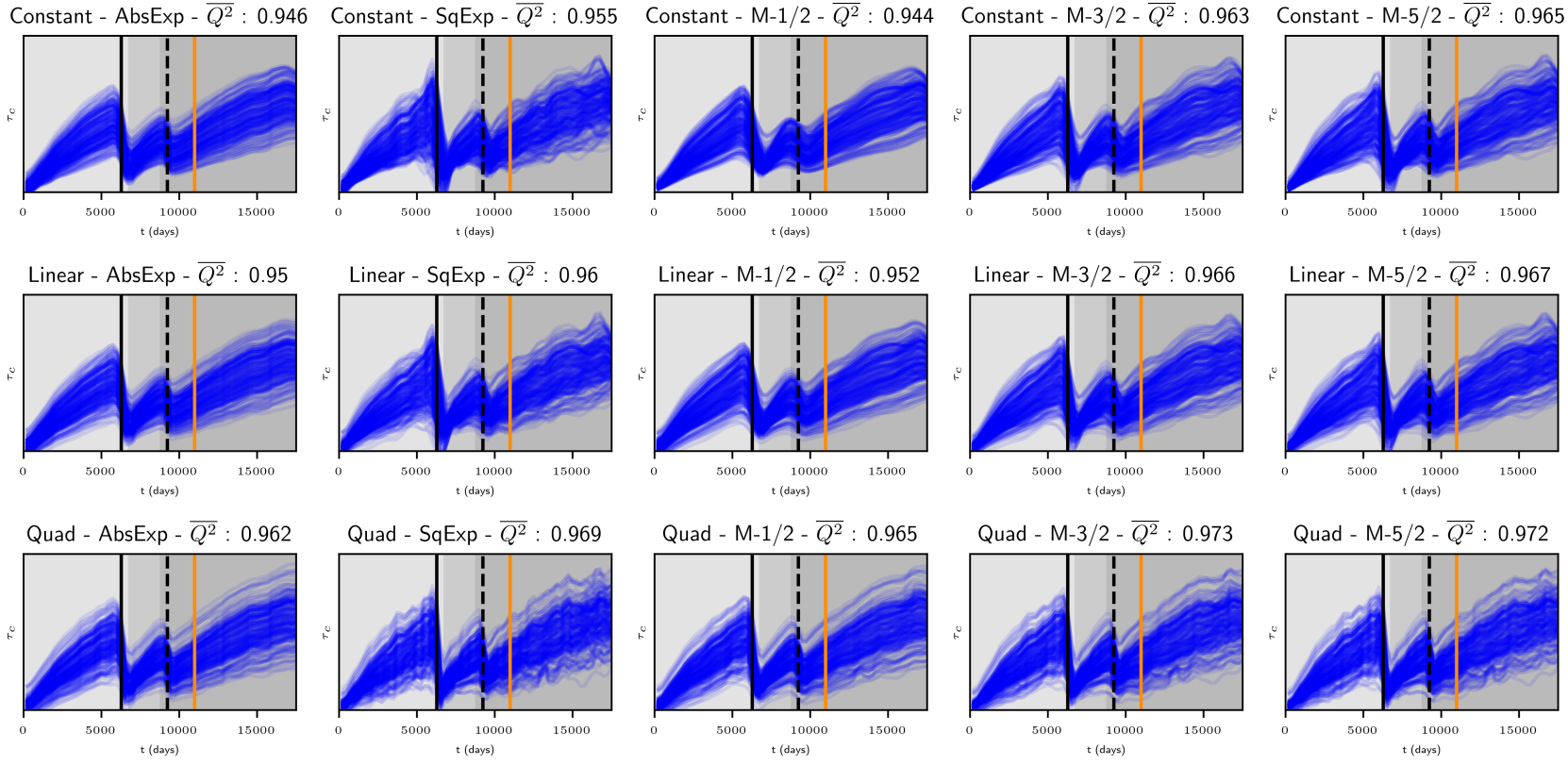


Figure 7: GP surrogate models of TPD with time stochastic collocation method and with different sets of priors

- All surrogate models have a very competitive predictive coefficient $\overline{Q^2} \sim 0.9$

  → *How to differentiate between them, which one is more robust? For which application?*

Conformal Prediction (CP)

- CP paradigm [Vovk et al., 2005] is a generic, model-agnostic theory allowing to build *prediction sets* for machine learning models like $\widehat{g}$ with frequentist coverage guarantees

Definition

[Vovk et al., 2005] Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space and $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$. Let $n \in \mathbb{N}$ and $\mathcal{D}_n = \{\boldsymbol{Z}^{(1)}, \ldots, \boldsymbol{Z}^{(n)}\} \in \mathcal{Z}^n$ a training sample. For $\alpha \in (0, 1)$, a conformal predictor of coverage $1 - \alpha$ is any measurable function of the form:

$$\begin{aligned} C_\alpha \colon \mathcal{Z}^n \times \mathcal{X} &\to 2^{\mathcal{Y}} \\ (\mathcal{D}_n, \boldsymbol{X}) &\mapsto C_{n,\alpha}(\boldsymbol{X}), \end{aligned} \tag{8}$$

s.t. for any new couple of points $\boldsymbol{Z}^{(n+1)} = (\boldsymbol{X}^{(n+1)}, Y^{(n+1)}) \in \mathcal{Z}$ (marginal coverage property):

$$\mathbb{P}\left(Y^{(n+1)} \in C_{n,\alpha}(\boldsymbol{X}^{(n+1)})\right) \geq 1 - \alpha \tag{9}$$

Jackknife+/minmax interval estimators

- There are three types of conformal estimators [Angelopoulos and Bates, 2023] → full, split and cross → last one is adapted for *small dataset*, i.e $n \propto 10^2$
- $\mathcal{D}_n$ an i.i.d. design of experiments, with scalar outputs
- $\widehat{g}$ a surrogate model trained on $\mathcal{D}_n$, $\widehat{g}_{-i}$ Leave-One-Out (LOO) surrogate model trained on $\mathcal{D}_n \backslash \{i - \text{th data-point}\}$
- With empirical quantiles of LOO residuals → can build interval estimators: Jackknife+ $\widehat{C}^{\text{J+}}_{n,\alpha}$ and Jackknife-minmax $\widehat{C}^{\text{J-minmax}}_{n,\alpha}$ [Barber et al., 2021]:

Estimators	$\widehat{C}^{\text{J+}}_{n,\alpha}$	$\widehat{C}^{\text{J-minmax}}_{n,\alpha}$
Marginal coverage	$\alpha \in (0, 1/2)$	$\alpha \in (0, 1)$
$+$	Cross-validation method Fast to compute	Coverage property
$-$	**Constant width**	Width is too large

Table 2: Pros and cons of main cross-conformal estimators

GP credibility intervals

- Credibility intervals of the posterior $\widetilde{\mathcal{G}}$, for any new point $\boldsymbol{X}^{(n+1)} \in \mathcal{X},\ \alpha \in (0,1)$ and all $k \in \{1,\dots,N\}$:

$$\mathcal{CR}_\alpha(t_k, \boldsymbol{X}^{(n+1)}) = \left[\, m_{\boldsymbol{\beta}^*}(t_k, \boldsymbol{X}^{(n+1)}) \pm u_{1-\alpha/2}\gamma_{\boldsymbol{\varphi}^*}(t_k, \boldsymbol{X}^{(n+1)}) \,\right]$$

- Has the training-conditional coverage property (stronger than marginal):

$$\mathbb{P}\left(g(t_k, \boldsymbol{X}^{(n+1)}) \in \mathcal{CR}_\alpha(t_k, \boldsymbol{X}^{(n+1)}) \mid \mathcal{D}_{n_{\text{train}}}\right) = 1-\alpha \qquad (10)$$

- *However*, the above equality relies on **two** hypotheses:
  1. The output is modeled by a GP
  2. The prior mean and covariance functions $m_{\boldsymbol{\beta}}, \gamma_{\boldsymbol{\varphi}}$ are suitable
- No generic way to test **misspecification** when metamodeling black-box computer codes → *a real challenge for industrial application of UQ methodology*

Conformalized GPs

- Fix for simplicity a certain $k \in \{1, \ldots, n\}$ and denote $\widetilde{g}(.) = m_{\beta^*}(t_k, .)$ and similarly for $\widetilde{\gamma}$
- Define the Leave-One-Out-Gaussian (LOOγ) error (with $\varepsilon > 0$ a small constant):

$$R_i^{\mathrm{LOO}\gamma} := \frac{|g(\boldsymbol{X}^{(i)}) - \widetilde{g}_{-i}(\boldsymbol{X}^{(i)})|}{\max\left(\varepsilon, \widetilde{\gamma}_{-i}(\boldsymbol{X}^{(i)})\right)}, \quad \forall i \in \{1, \ldots, n\} \tag{11}$$

Main result and consequences [Jaber et al., 2025a]:

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

- Coverage property still verified for $\alpha \in (0, 1/2)$
- Intervals have adaptive width → more informative
- No hypotheses for interpreting the interval!
- The J+GP-minmax variant has the same properties

Example on a GP

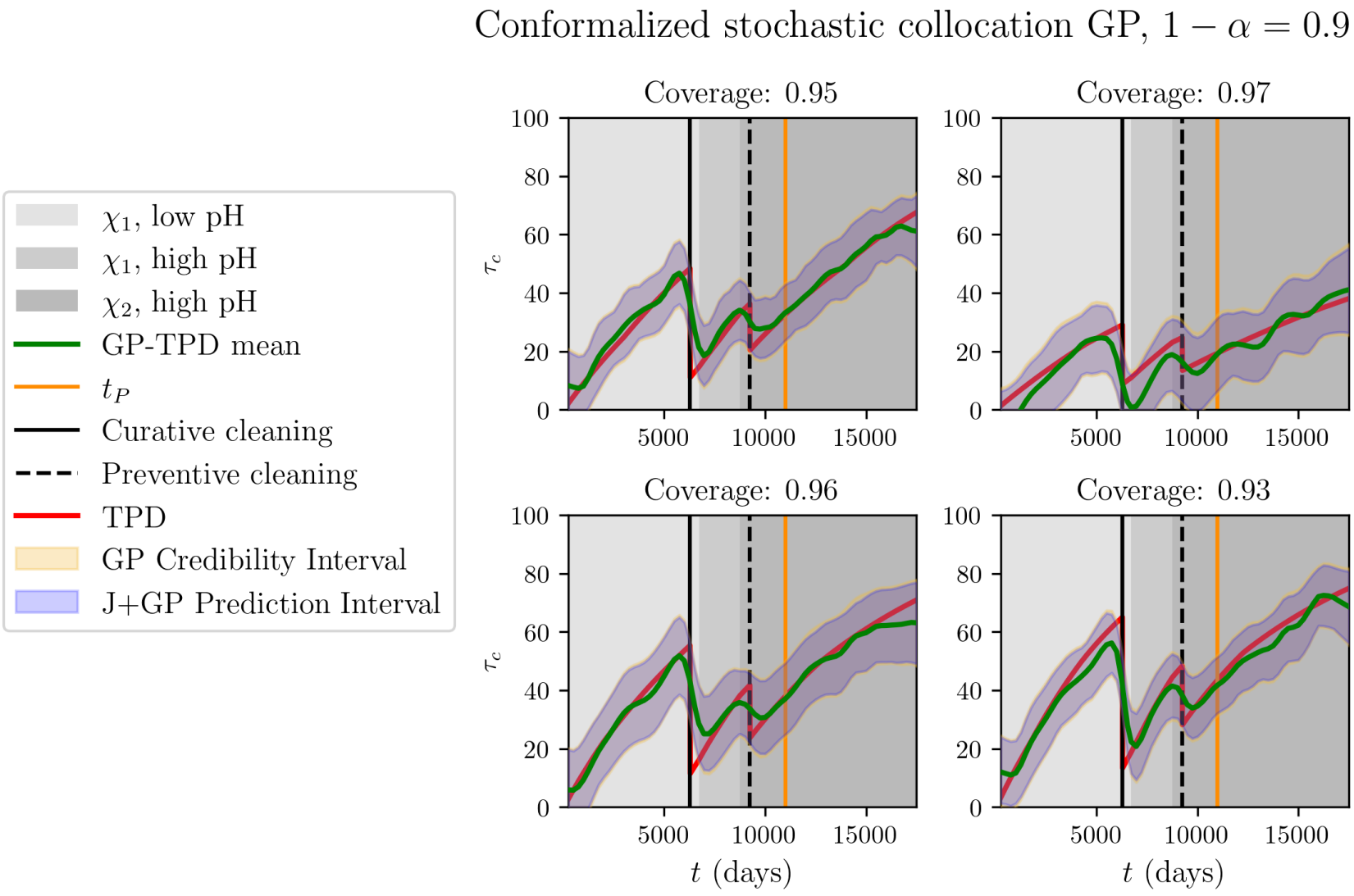


Figure 8: Stochastic GP metamodel of TPD with a linear trend and a Matérn-$1/2$ kernel

Results on different GPs

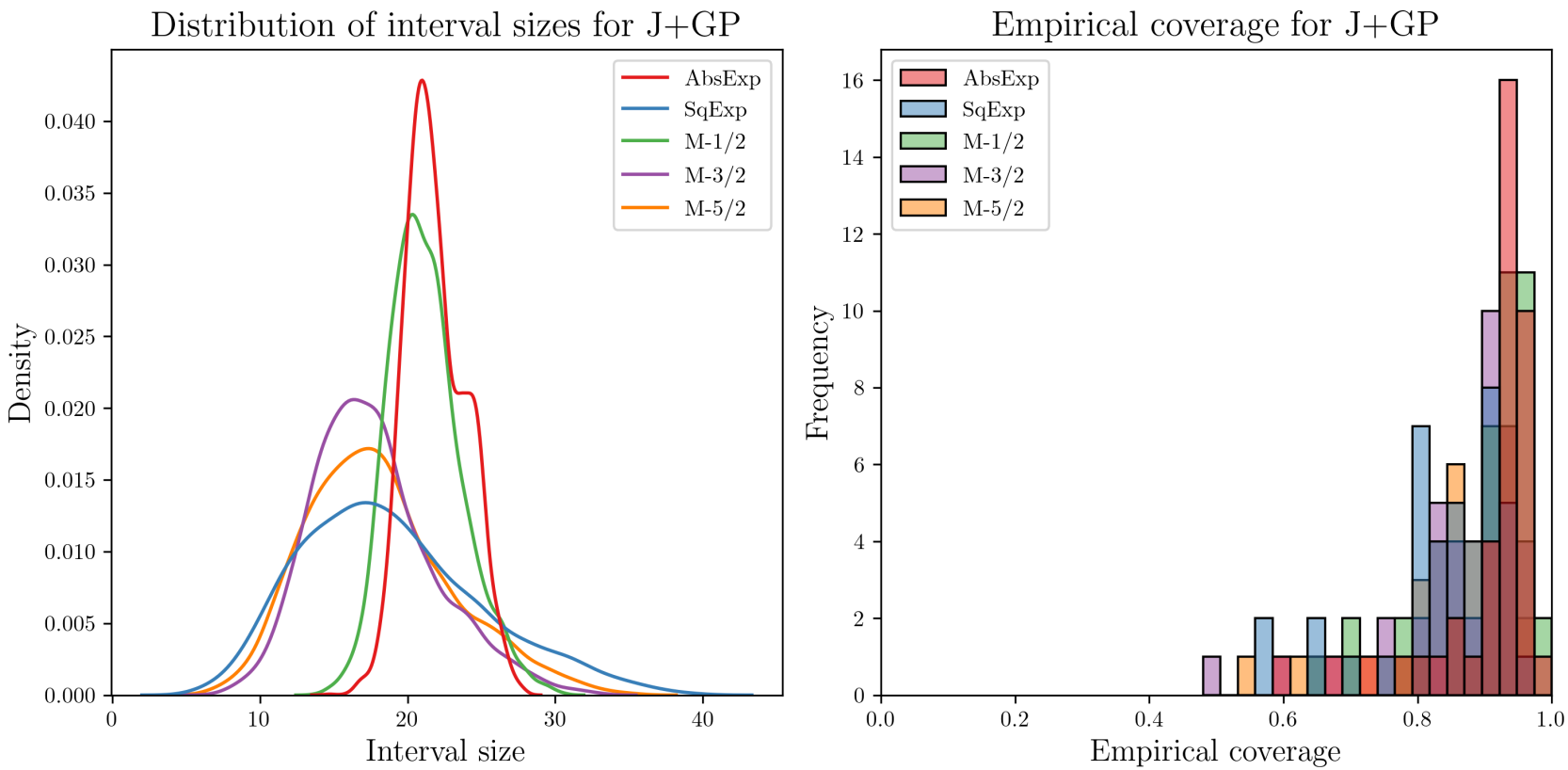


Figure 9: Distribution of interval sizes of the J+GP conformal predictor and empirical coverage distributions for a stochastic collocation GP with a linear trend and different kernels

Summary of 3

Take-home messages:

- → TPD can be approximated using a variety of non-intrusive surrogate modeling techniques
- → Stochastic time-collocation GP method can be applied for time-interpolating surrogates if monotonicity properties of the code are satisfied
- → Cross-CP can be applied to go beyond the predictivity coefficient for qualifying non-intrusive GP surrogate models

Contributions:

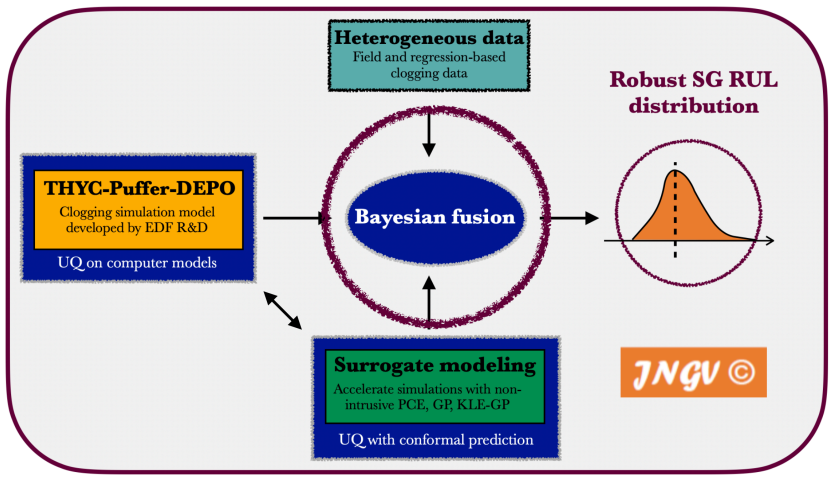
- Work on CP+GP with Dr. Vincent BLOT at the CEMRACS 2023 summer school on Scientific Machine Learning; Oral communication at SIAM-UQ 2024
- Paper [Jaber et al., 2025a] co-authored published in the **Journal of Machine Learning for Modeling and Computing**
- `Github repository` with plug-and-play scripts based on the OpenTURNS and MAPIE [Cordier et al., 2023] Python libraries

Outline

Hybrid framework for the JNGV

Objectives:

- Reduce the parametric uncertainty of TPD using a Bayesian methodology to get a more informed RUL distribution
- Use the heterogeneous data (field and regression-based), the information from the UA on TPD; the surrogates



Methodology:

- *State of the art:*
  - → Data assimilation
  - → Bayesian calibration
  - → MCMC algorithms, ensemble methods
- *Strategy:* Generalize the Bayesian modeling in [Keller et al., 2022] to heterogeneous groups of data, include it in an offline data assimilation methodology

Available tools

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

$$g(\boldsymbol{x}_0) = (g(t_1, \boldsymbol{x}_0), \ldots, g(t_N, \boldsymbol{x}_0)) \tag{12}$$

  and $\mathrm{pr}_\ell \circ g(\boldsymbol{X}) := g(t_\ell, \boldsymbol{X})$ models $\Delta(t_\ell)$ → grey-box, physics known with non-intrusive surrogate modeling strategy $\widehat{g}$

- q *heterogeneous degradation data groups* (from different sensors, statistical models,...) $\mathcal{D} = (\boldsymbol{y}^1, \ldots, \boldsymbol{y}^q)$ with different sizes $\boldsymbol{y}^i \in \mathbb{R}^{m_i}$ → 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:

$$\boldsymbol{y}^i(t_\ell) = \Delta(t_\ell) + \eta_\ell^i \tag{13}$$

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

- ***How to fuse these tools for hybrid RUL estimation of the industrial 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 $\boldsymbol{Y} = g(\boldsymbol{X}) + \epsilon = \widehat{g}(\boldsymbol{X})$ are the computer model emulator outputs on a prescribed time window and $\boldsymbol{X}$ are uncertain parameters that do not vary over time
- The goal of assimilation is to estimate the distribution $p(\boldsymbol{Z} \mid \mathcal{D})$
- Use *modular* approach on each time window:
  1. Tailored Bayesian model updating (BMU) to obtain a posterior distribution $p_{\boldsymbol{X}|\mathcal{D}}$ for the input parameters: $(\boldsymbol{X}|\mathcal{D}) \sim p_{\boldsymbol{X}|\mathcal{D}}$.
  2. Apply smoothing to estimate $p(\boldsymbol{Y}|\boldsymbol{X} \sim p_{\boldsymbol{X}|\mathcal{D}}, \mathcal{D})$, yielding the assimilated posterior approximation:

$$p(\boldsymbol{Z}|\mathcal{D}) \approx p(\boldsymbol{Y}|\boldsymbol{X} \sim p_{\boldsymbol{X}|\mathcal{D}}, \mathcal{D}) p_{\boldsymbol{X}|\mathcal{D}}. \tag{14}$$

Illustration of the methodology

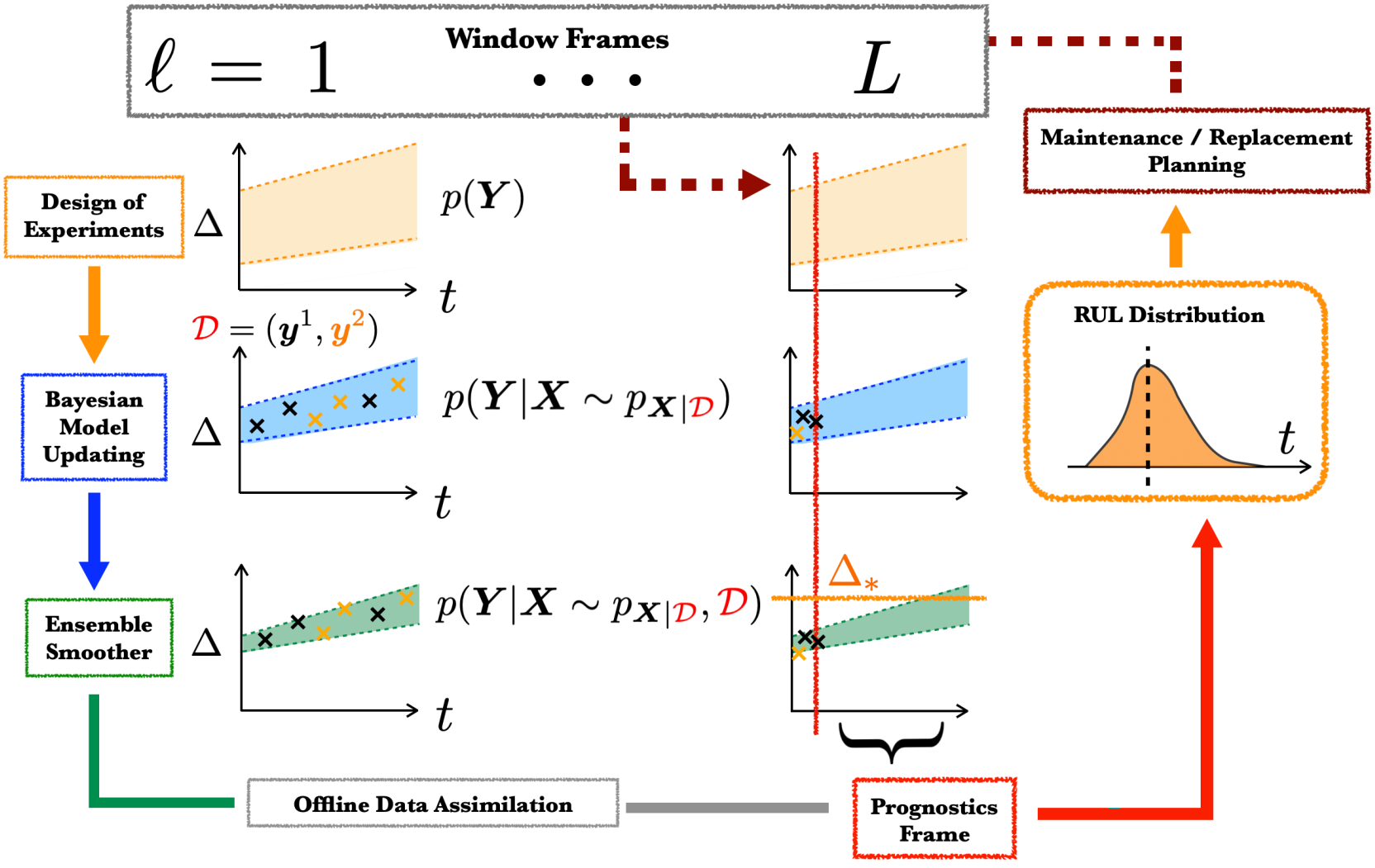


Figure 10: A sketch of the offline data assimilation methodology

On the prognostics window

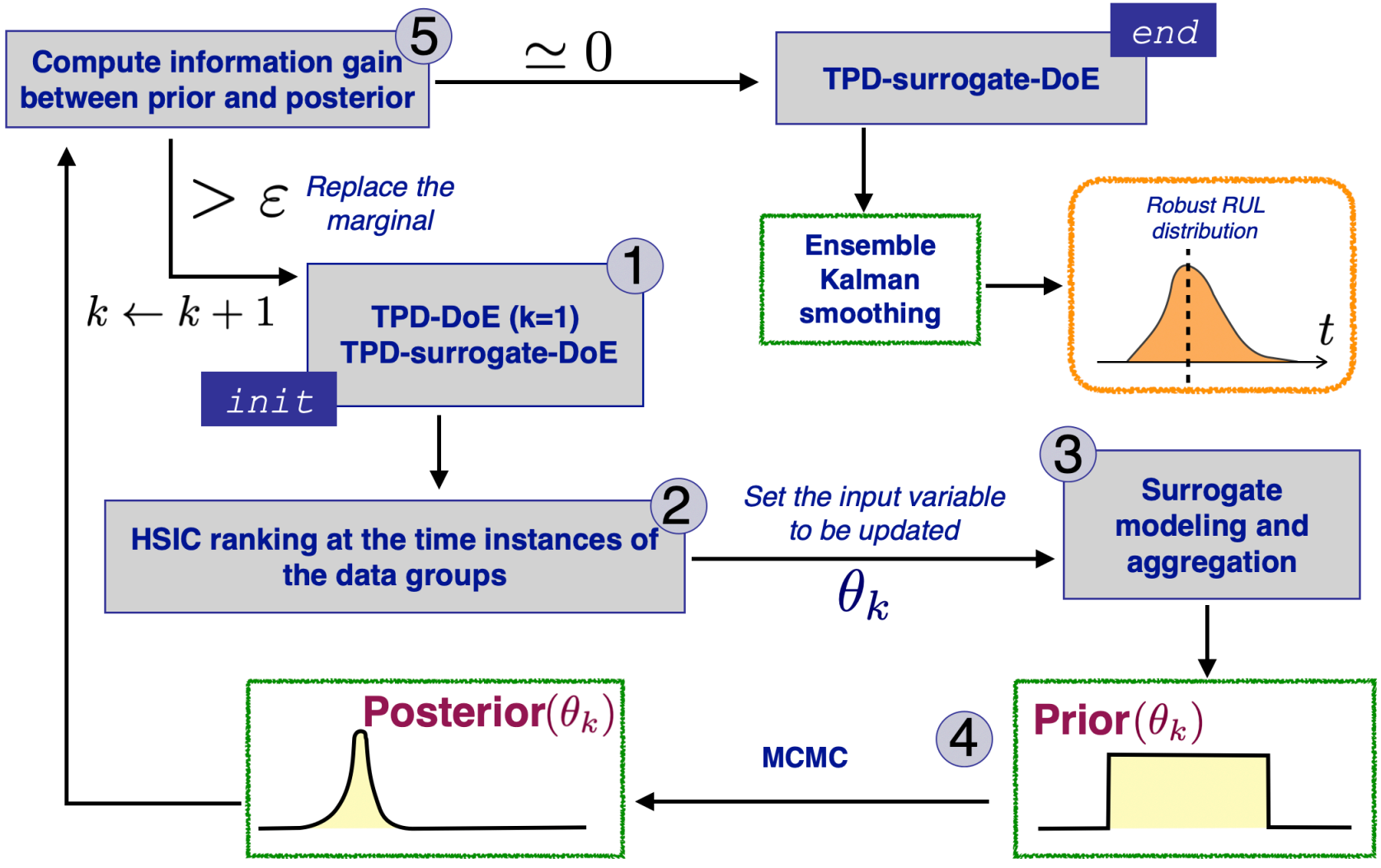
- Assimilation is performed offline until the prognostics window $\ell = L$ is reached, then the RUL distribution can be computed:

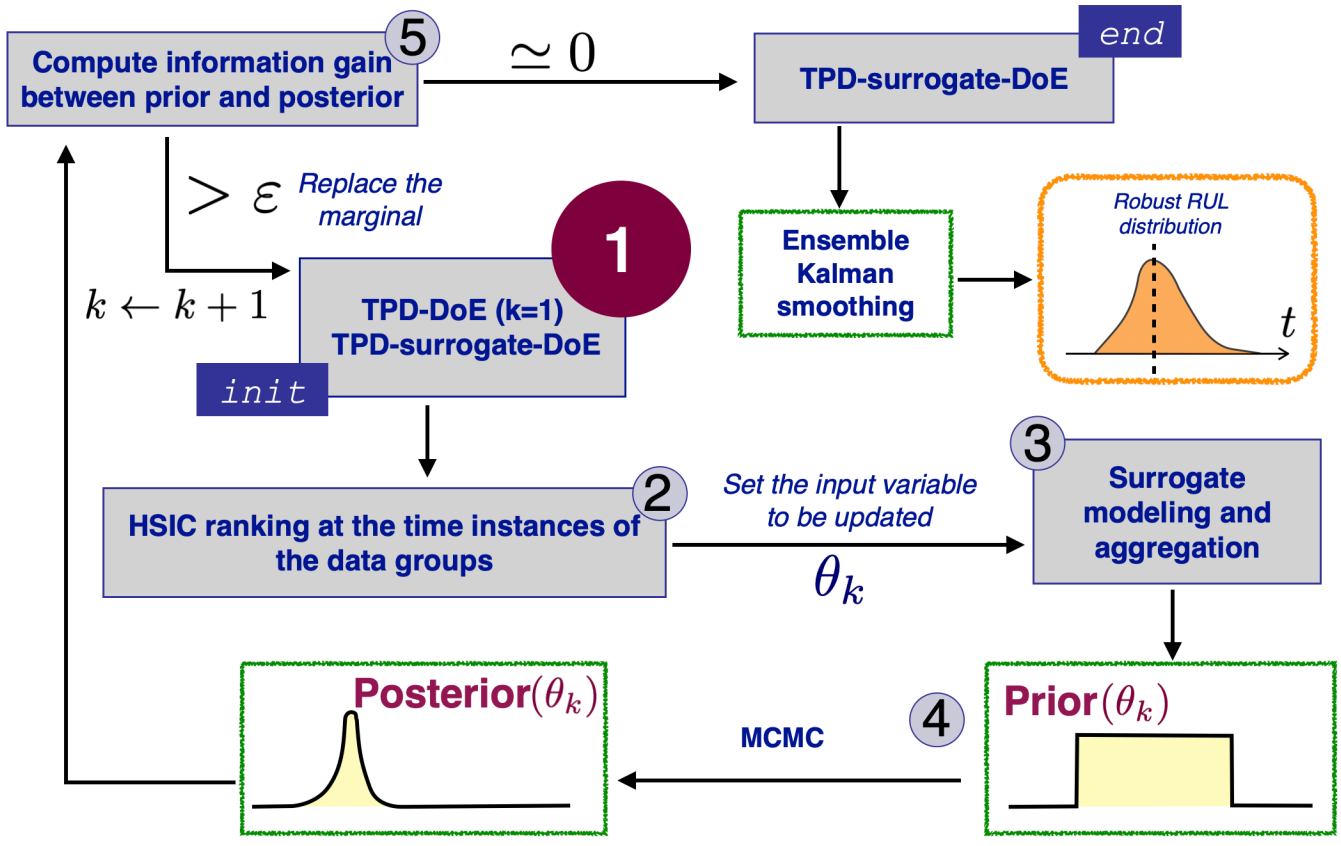
$$\mathbb{P}(\mathsf{RUL}(\Delta_*) \leq t_j \mid \mathcal{D}) = \int_{\mathbb{R}} \mathbf{1}\{\mathsf{pr}_{j+1}(\boldsymbol{y}) \geq \Delta_*\} p(\boldsymbol{y}|\mathcal{D}, \boldsymbol{X} \sim p_{\boldsymbol{X}|\mathcal{D}}) d\boldsymbol{y}, \tag{15}$$

- In practice this probability is estimated using a Monte Carlo ensemble $\{(\boldsymbol{X}^{(i)}, \widehat{g}(\boldsymbol{X}^{(i)}))\}_{i=1}^n \sim p_{\boldsymbol{X}|\mathcal{D}} \otimes \widehat{g}\#p_{\boldsymbol{X}|\mathcal{D}}$:

$$\mathbb{P}(\mathsf{RUL}(\Delta_*) \leq t_j \mid \mathcal{D}) \approx \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{\mathsf{pr}_{j+1} \circ \widehat{g}(\boldsymbol{X}^{(i)}) \geq \Delta_*\} \tag{16}$$

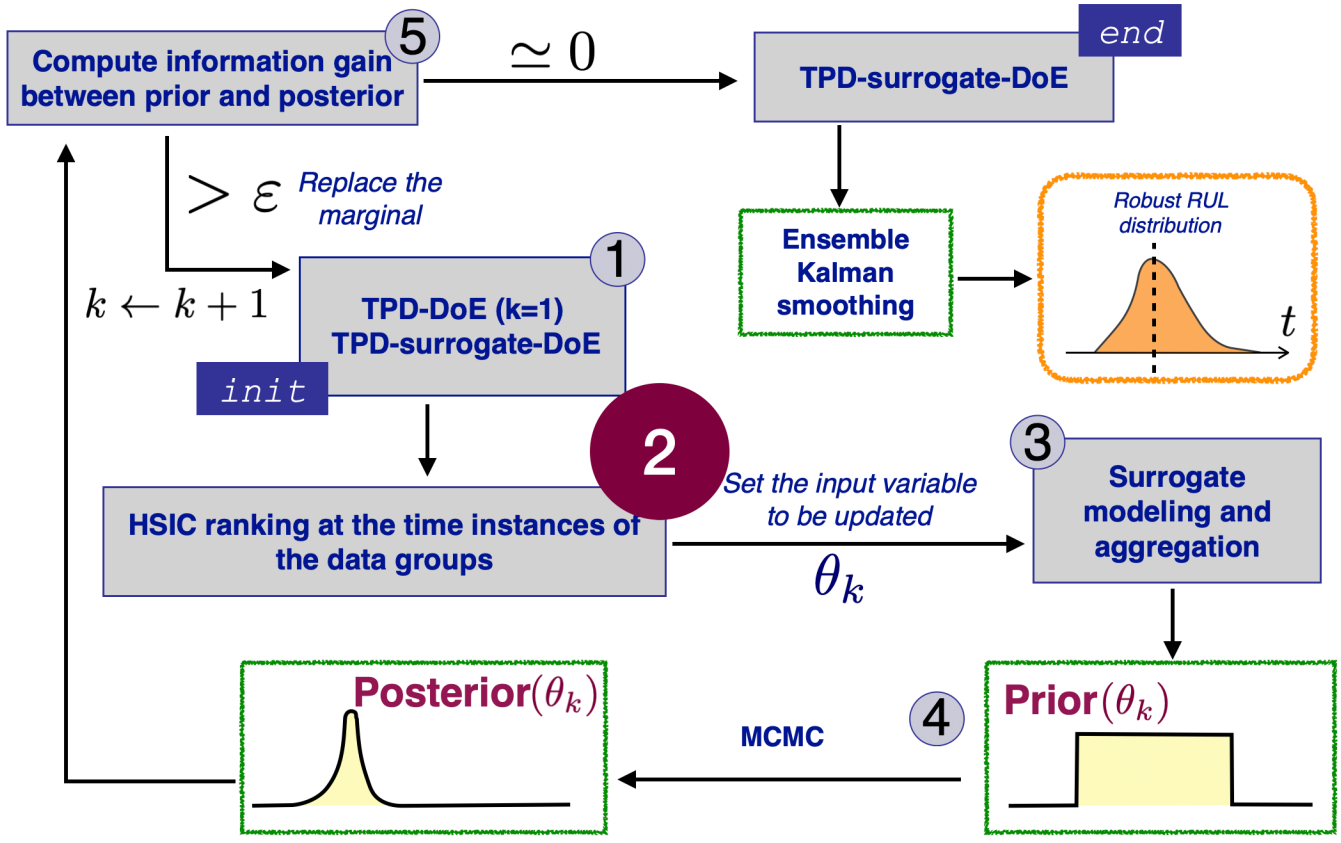
The BMU algorithm



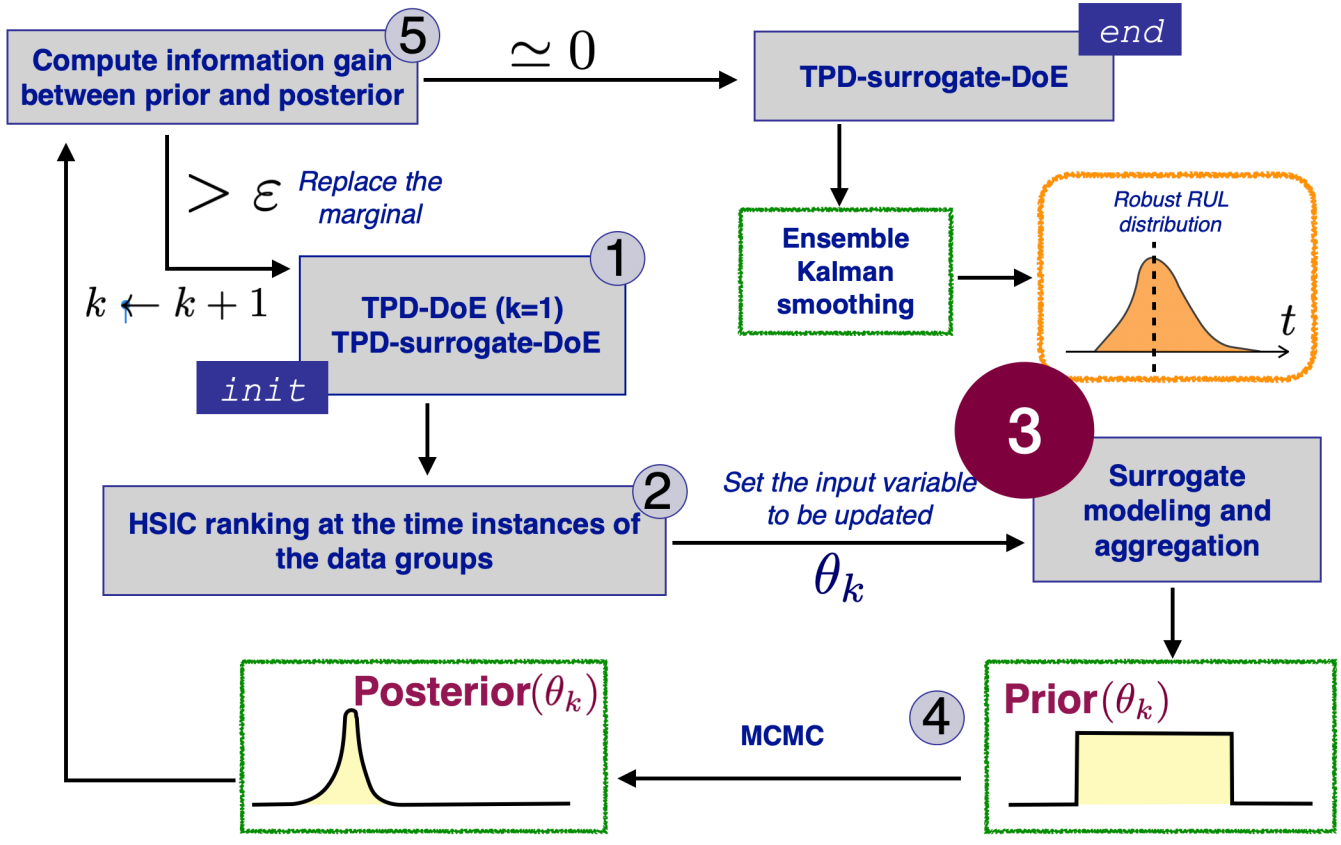


Perform k iterations where $1 \leq k \leq d$:

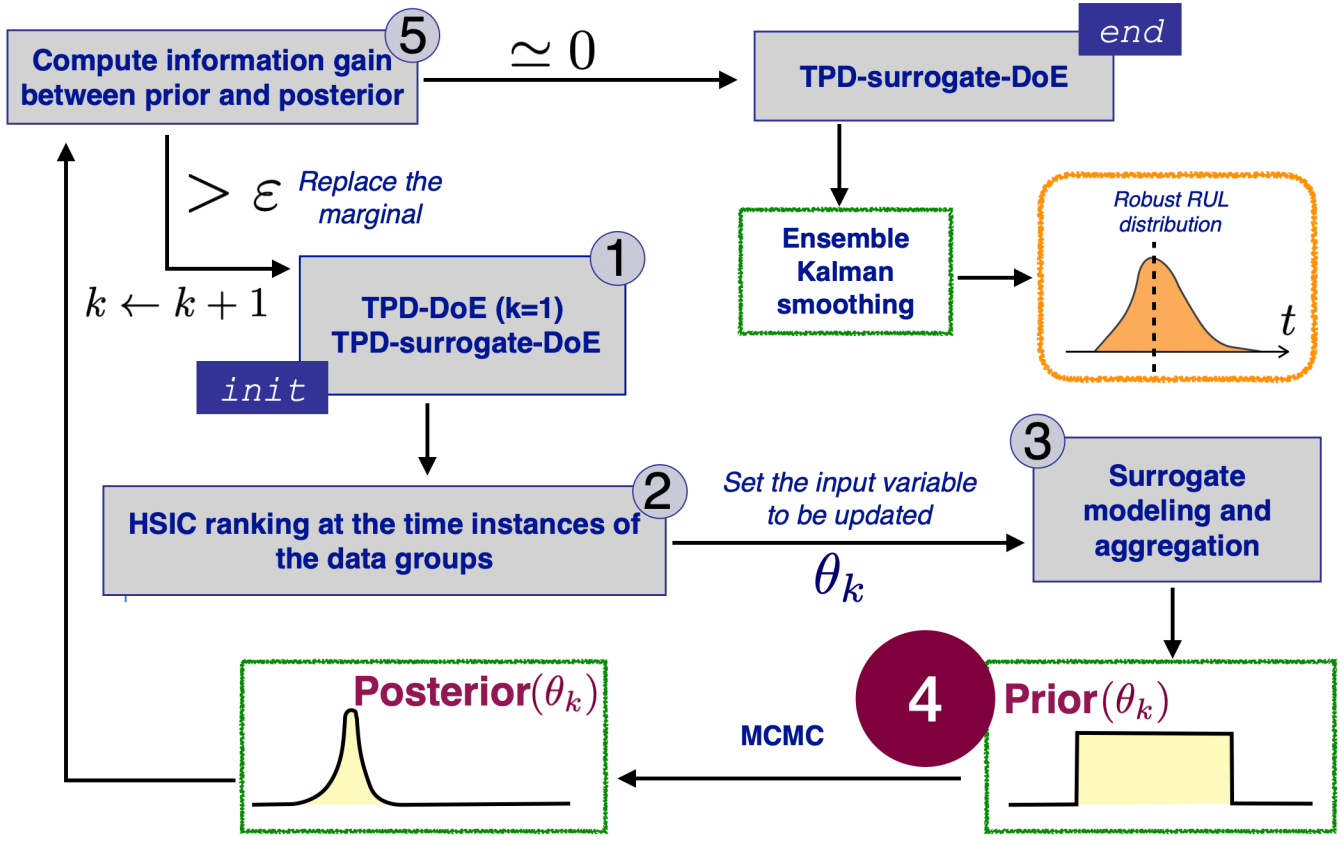
1. If $k = 0$, assume uniform independent priors $\mu_{\boldsymbol{X},k} \simeq \mathcal{U}[-1,1]^{\otimes d}$ → generate a design of experiments $\text{DoE}_g^{\mu_{\boldsymbol{X},k}} = \{(\boldsymbol{X}^{(j)}, \widehat{g}(\boldsymbol{X}^{(j)}))\}_{1 \leq j \leq n}$



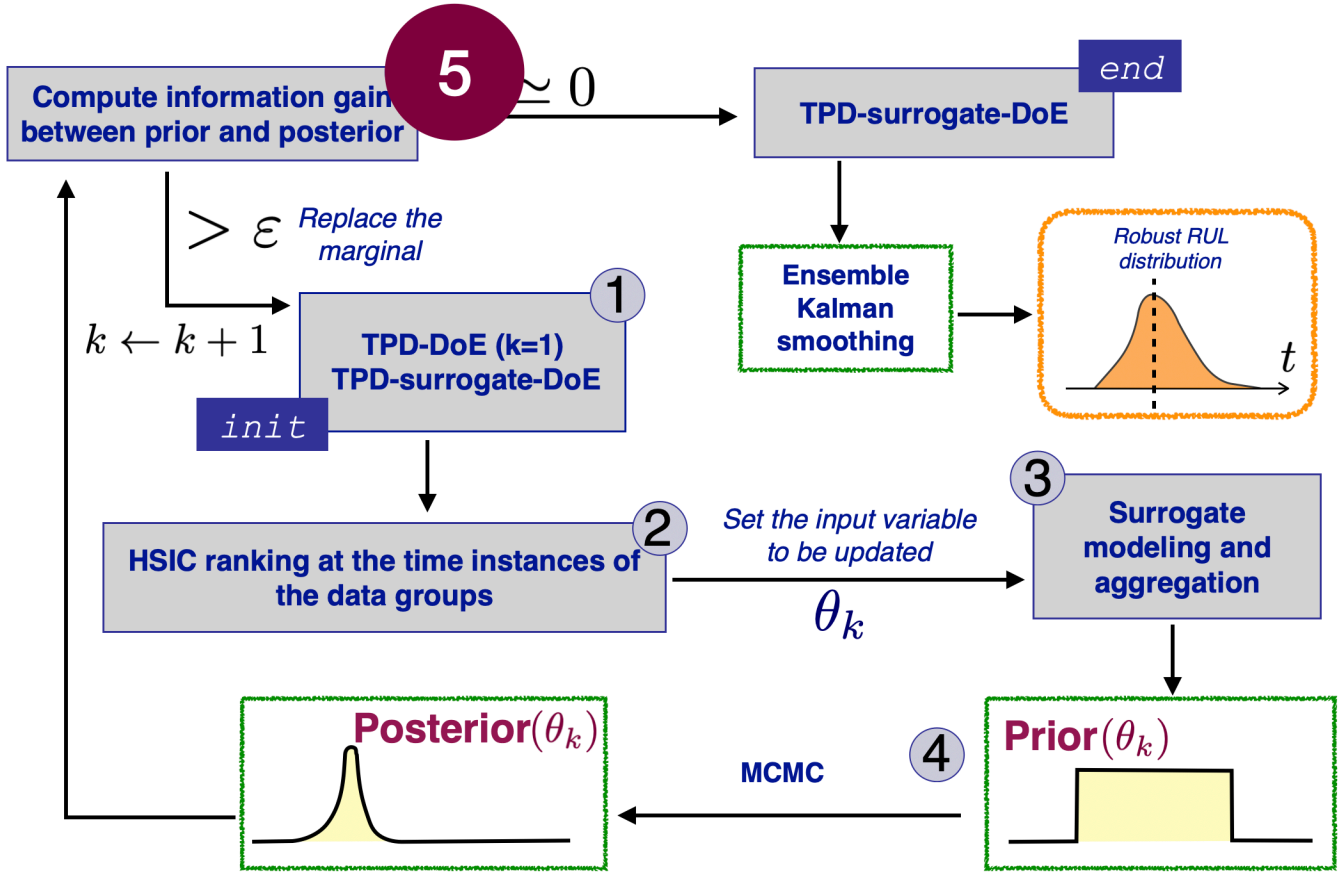
2. Compute **HSIC indices** [Gretton et al., 2005] between input variables and outputs at data time instances → evaluate local sensitivity



3. If g is time-costly, build and validate p metamodels $\widehat{\boldsymbol{g}} = (\widehat{g}^{(1)}, \ldots, \widehat{g}^{(p)})$ with chosen strategy → 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



4. Estimate the posterior distribution $\widehat{p}(\theta_k|\mathcal{D}, \boldsymbol{u}_{0,k})$ with an Monte Carlo Markov Chain (MCMC) sampling procedure



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 $\widehat{p}(\theta_k|\mathcal{D}, \boldsymbol{u}_{0,k})$ and continue $k \leftarrow k+1$
   - Otherwise, stop and obtain an *updated* RUL prediction by computing $g\#\mu_{\boldsymbol{x},k^*}$

Bayesian updating step

$$p(\theta_k|\mathcal{D}, \boldsymbol{u}_{0,k}) \propto \frac{1}{M}\sum_{r=1}^{M}\prod_{i=1}^{q}\|\boldsymbol{y}^i - \langle \boldsymbol{w}^{(r)}, \widehat{\boldsymbol{g}}(\boldsymbol{u}_{0,k}, \theta_k)\rangle\|^{-m_i} \tag{17}$$

- Use Random Walk Metropolis-Hastings (RWMH) MCMC algorithm [Sullivan, 2015] to sample from (17)
- Monte-Carlo integration using sample $\{\boldsymbol{w}^{(r)}\}_{r=1}^{M}$ from the Dirichlet-$\mathbf{1}_p$ distribution on the simplex → integrate hyperparameter
- Updated densities are conditioned on nominal values $\boldsymbol{u}_{0,k}$ of *other* $d-1$ *input variables* → future work on how to integrate uncertainty

Ensemble Kalman smoothing step

- 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 relevant paradigm for this is smoothing, i.e getting estimates from the full-posterior distribution $p(g(t_1, \boldsymbol{X}), \ldots, g(t_k, \boldsymbol{X})|\mathcal{D})$ → Ensemble Kalman smoothing (EnKS) [Evensen and Van Leeuwen, 2000]

Results: posterior distributions

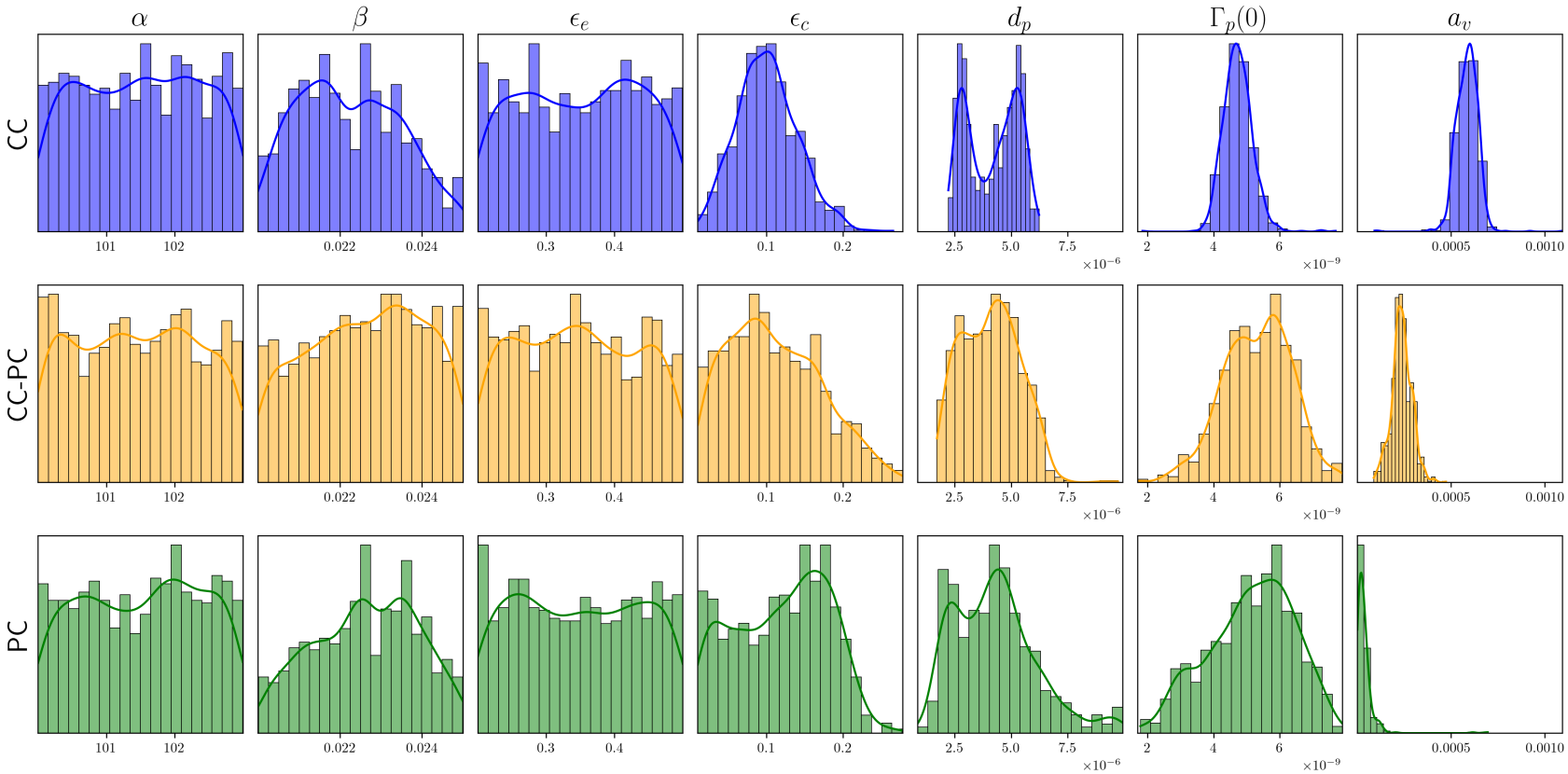


Figure 11: Posterior distributions of TPD clogging simulation code

- $L = 3$ time windows corresponding to cleanings CC/CC-PC/PC
- $5/7$ distributions are informed by the data, distinct modes for a_v

Results: posterior trajectories

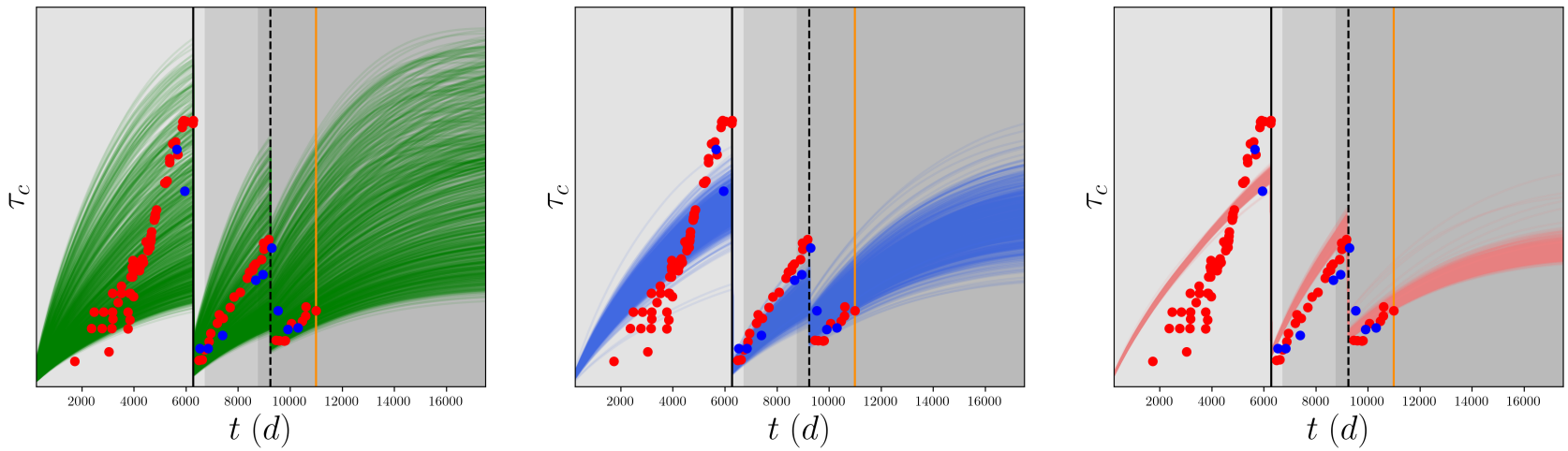
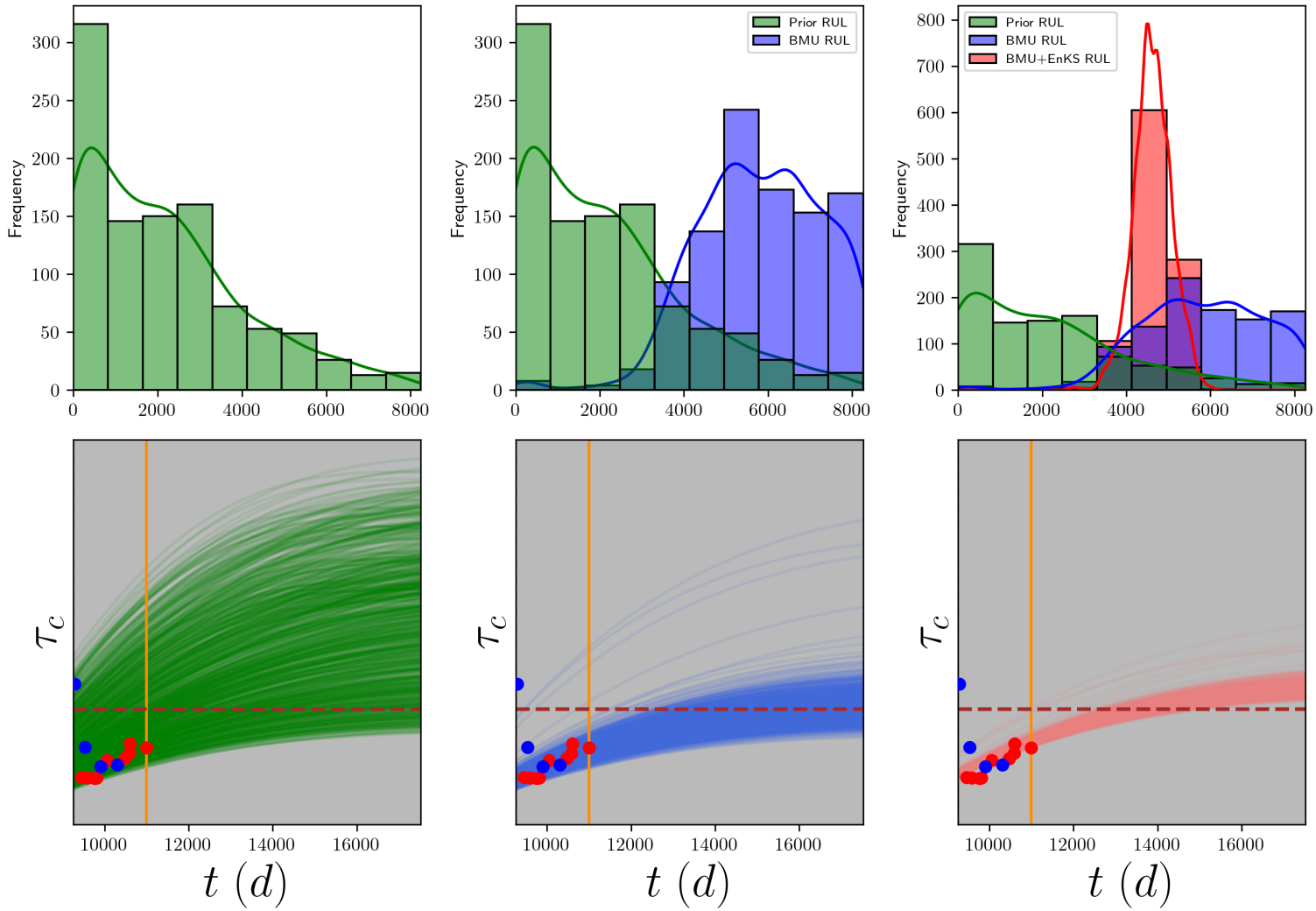


Figure 12: Prior/posterior and smoothed TPD emulations with Karhunen-Loève expansion as main surrogate

Results: posterior RUL



- RUL prediction uncertainty substantially reduced and mean of the distribution shifted compared to the prior

→ ***More informed maintenance planning!***

Summary of 4

Take-home messages:

- → Offline data assimilation generic modular approach involving iterative BMU, TPD emulators, heterogeneous data groups and ensemble Kalman smoothing
- → Reduces parametric uncertainty of the TPD - RUL prediction, more informed decision making

Contributions:

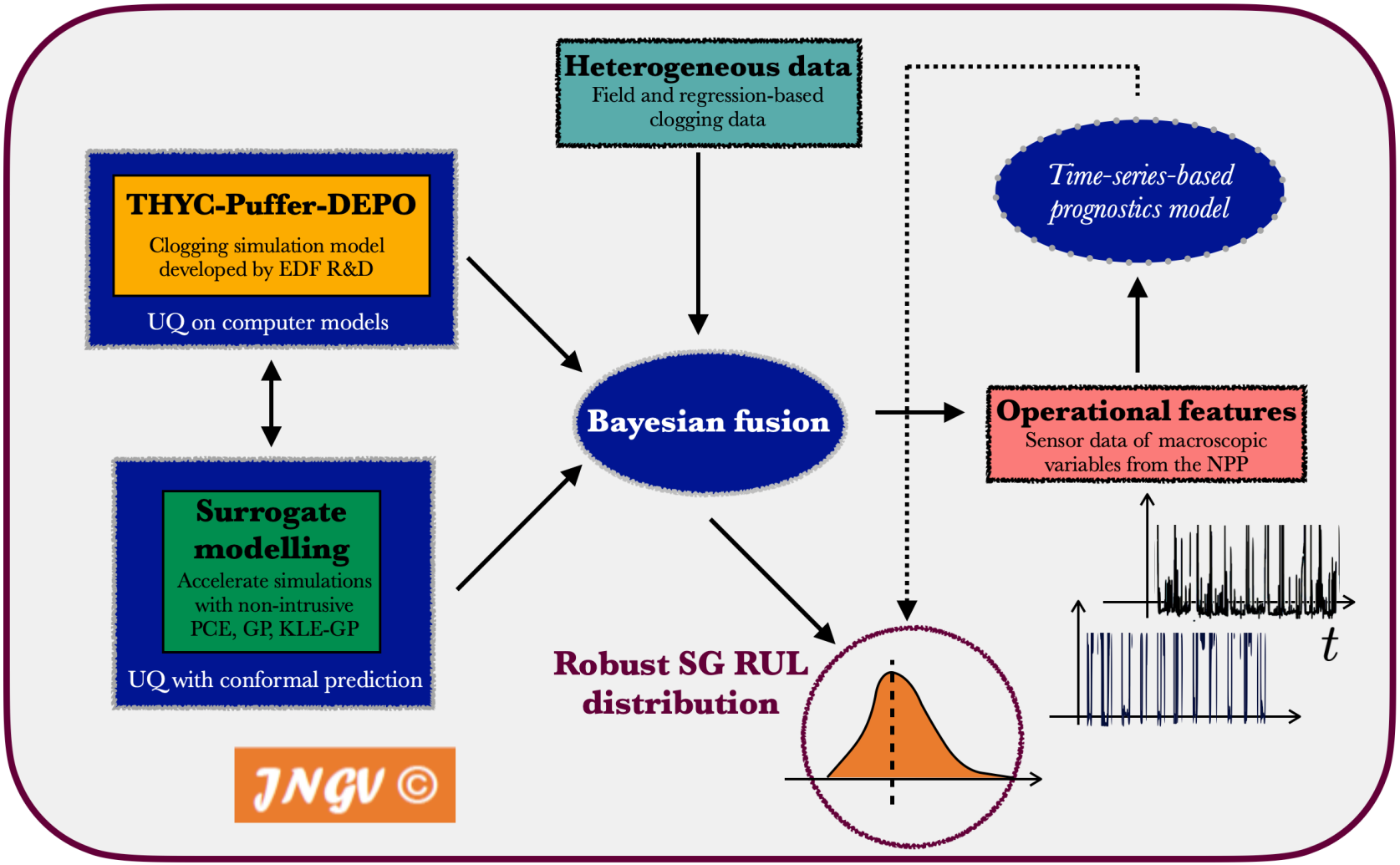
- Paper [Jaber et al., 2025c] under review in **Journal of Reliability Engineering & System Safety**
- Reproducible code in `GitHub repository`
- Communications: INI Workshop on calibrating prediction uncertainty - Cambridge, U.K, Mathematical foundations of digital twins - CIRM, Marseille

Outline

Key take-aways

- Hybrid framework methodology for reliable RUL distribution involving:
  - → Clogging simulation code TPD with parametric uncertainty
  - → Different non-intrusive surrogate modeling techniques of TPD and validation of GPs enhanced with CP
  - → Bayesian fusion methodology based on an offline data assimilation framework
- Three papers on each subject [Jaber et al., 2025b,a,c] and numerous communications in international conferences and workshops
- Exploration of time-series prognostic models to combine NPWR monitoring data with simulation outputs (in manuscript)

Full hybrid framework



Some scientific perspectives

- Enable online (real-time) data assimilation by developing non-destructive measurements to update TPD model continuously and time-series prognostics models
- Create dedicated experimental validation datasets and strengthen collaboration between experimentalists and modelers to better calibrate and benchmark physical formulations and closure laws
- Benchmark clogging models at different fidelity levels, perform higher-order sensitivity analyses of THYC closure laws, and propagate parametric uncertainty across THYC, Puffer, and DEPO (e.g., via stochastic field representations)

Industrial perspectives

- Embed the UA and hybrid modules into EDF's SG DT (JNGV) platform, provide clear dashboards, secure data pipelines and user workflows, and deploy the method across the SG fleet
- The offline data assimilation approach can be adapted to other SG failure modes (if a physical model is possible) and to DTs in different industries
- Develop Verification and Validation procedures, regulatory-aligned validation and explainability tools to build operator and regulator trust in nuclear DTs

Thank you for your attention!

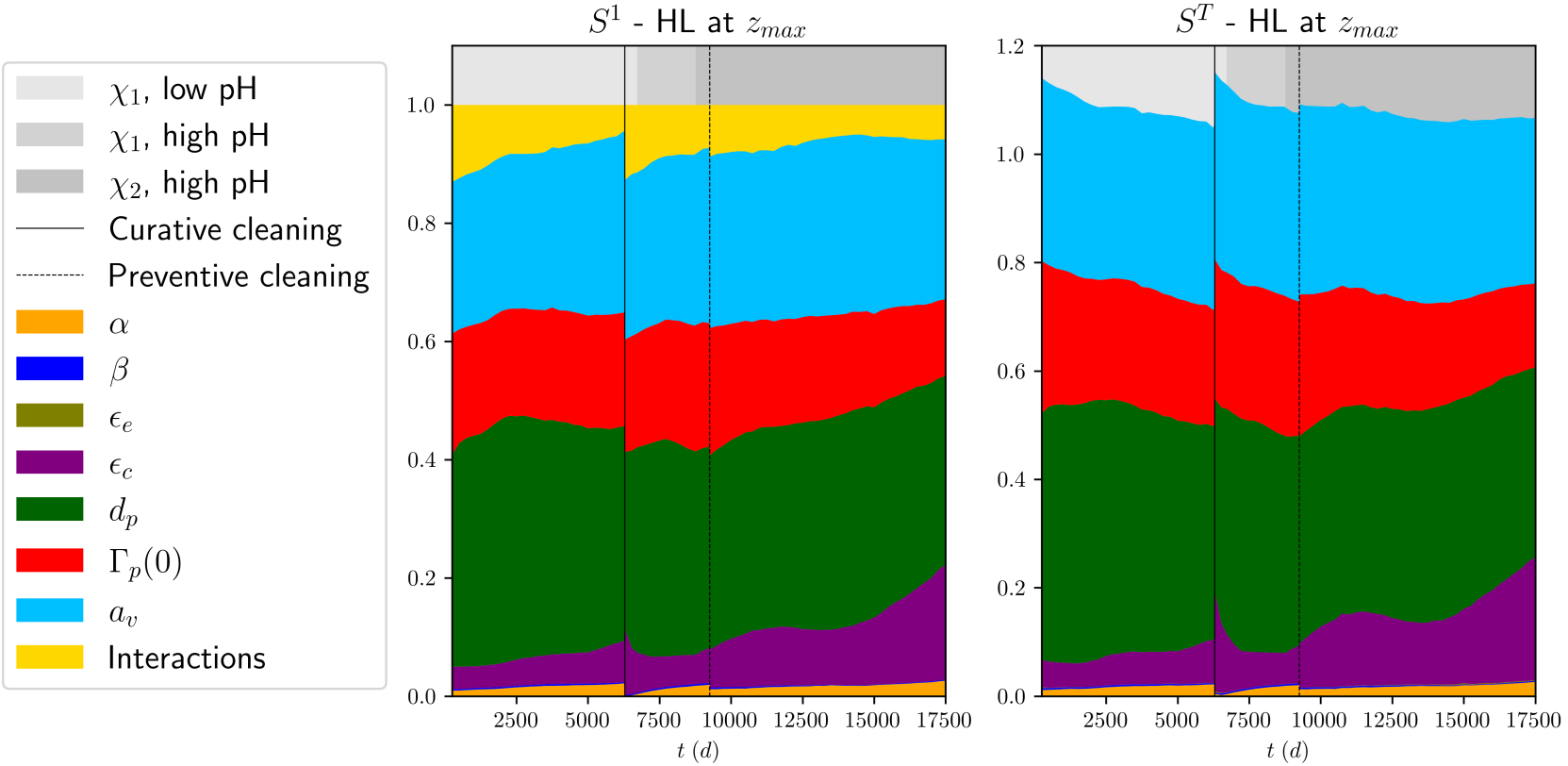
[Jaber et al., 2025b] ***Jaber, E.***, *Chabridon, V., Remy, E., Baudin, M., Lucor, D., Mougeot, M., Iooss, B., Sensitivity Analyses of a Multi-Physics Long-Term Clogging Model For Steam Generators,* ***International Journal of Uncertainty Quantification****, 2025, DOI: 10.1615/Int.J.UncertaintyQuantification.2024051489*

[Jaber et al., 2025a] ***Jaber, E.***, *Blot, V., Brunel, N., Chabridon, V., Remy, E., Iooss, B., Lucor, D., Mougeot, M., Leite, A., Conformal Approach to Gaussian Process Surrogate Evaluation with Marginal Coverage Guarantees,* ***Journal of Machine Learning for Modeling and Computing****, 2025, DOI: 10.1615/JMachLearnModelComput.2025054687*

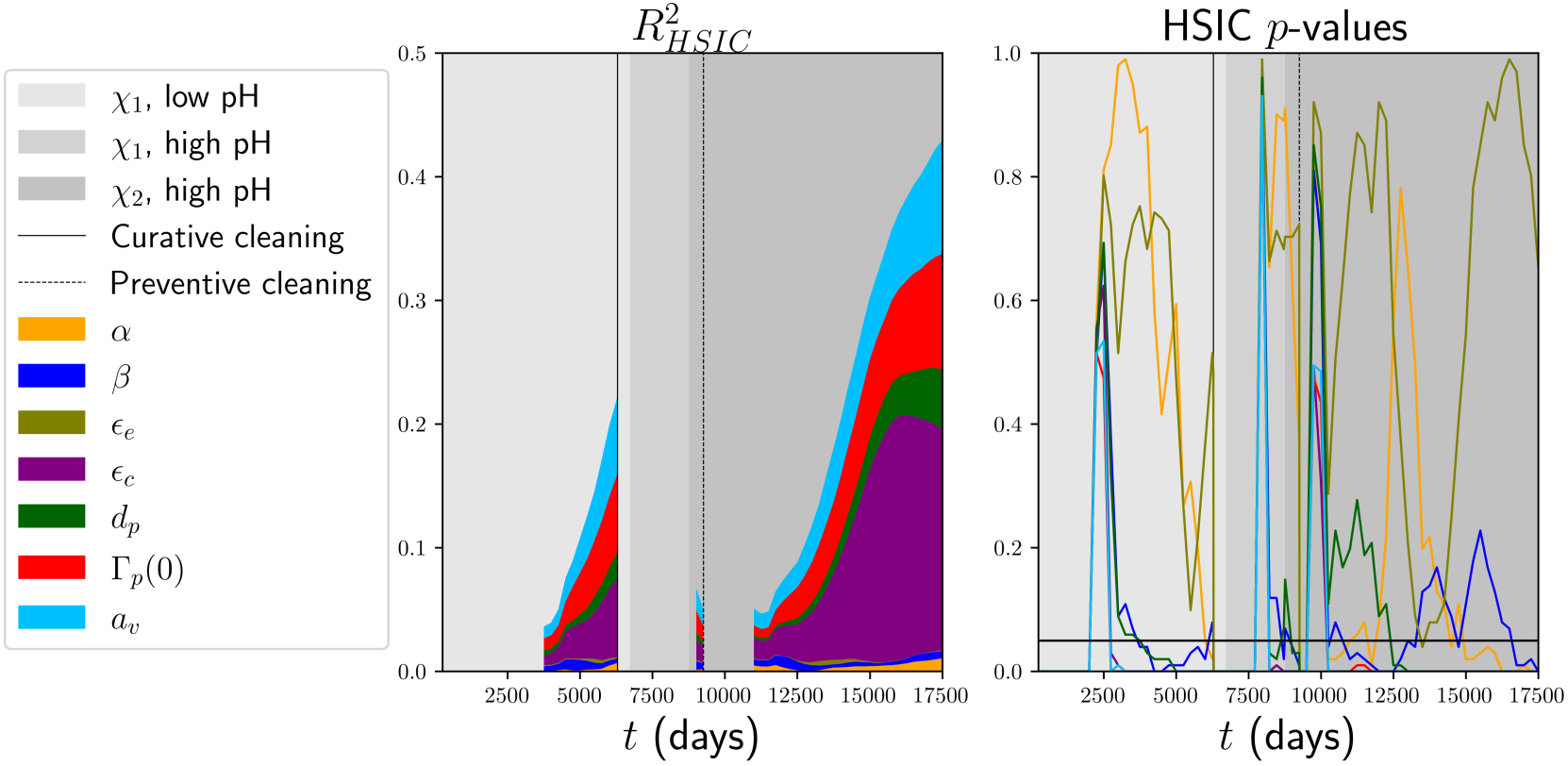
[Jaber et al., 2025c] ***Jaber, E.***., *Remy, E., Chabridon, V., Lucor, D., Mougeot, M., Fusion of heterogeneous data for robust degradation prognostics, ArXiV 2506.05882, 2025,* ***Reliability Engineering and System Safety****, under review*

Outline

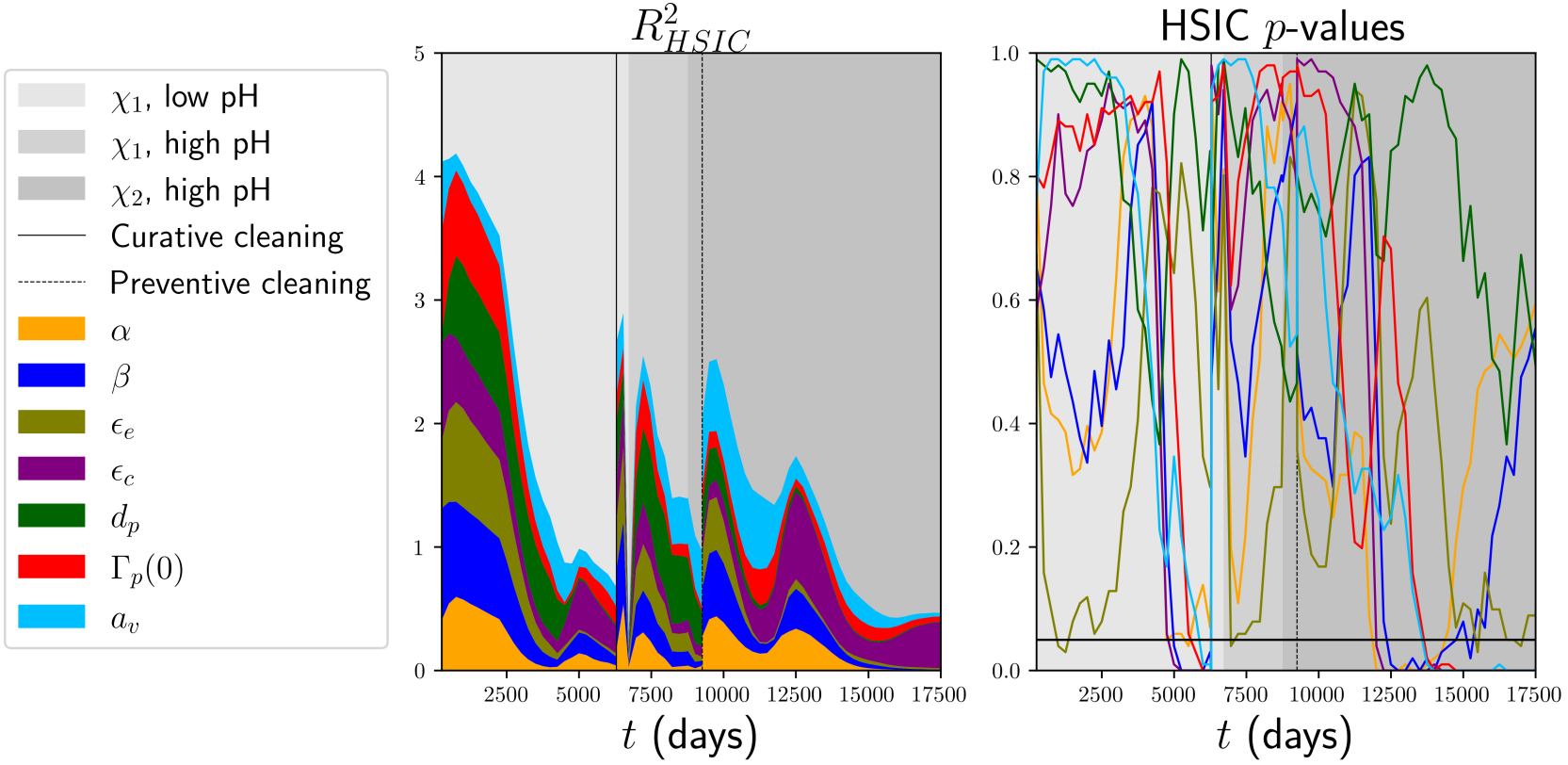
Sobol' indices



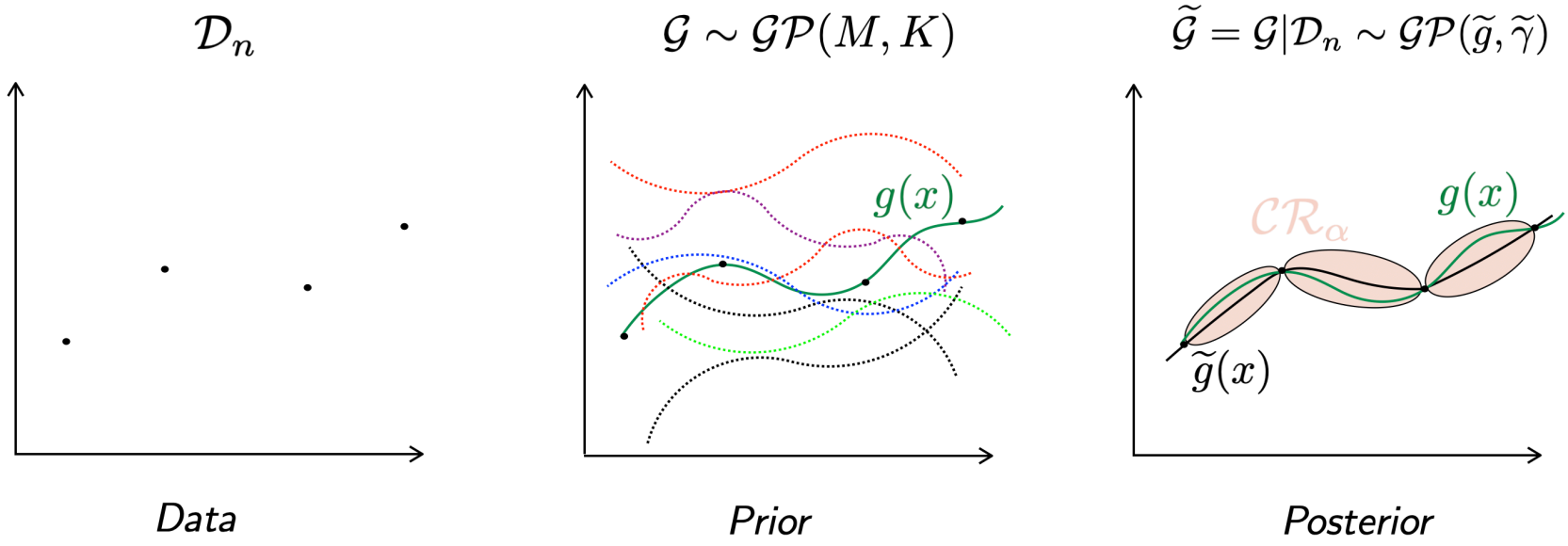
Target HSIC-indices



Conditional HSIC-indices



Gaussian processes



- GP metamodel prediction is the *posterior mean* $\widehat{g} = \widetilde{g}$
- Notion of uncertainty through the posterior covariance $\widetilde{\gamma}$ and the Gaussian structure of the metamodel

Bayesian updating step

Proposition

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

$$p(\theta|\boldsymbol{y}) \propto \|\boldsymbol{y} - f(\theta)\|^{-m} \tag{18}$$

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

$$p(\theta|\boldsymbol{y}^1, \ldots, \boldsymbol{y}^q) \propto \prod_{i=1}^{q} \|\boldsymbol{y}^i - f(\theta)\|^{-m_i} \tag{19}$$

Proof. Bayes' theorem and simplifications

Ensemble Kalman smoothing

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

$$y^i(t_\ell^i) = g(t_\ell^i, \boldsymbol{X}) + \eta_i, \tag{20}$$

  where the variance R^i of the noise *is known*
- Define the ensemble mean and anomalies:

$$\overline{g}(t) = \frac{1}{n}\sum_{k=1}^{n} g(t, \boldsymbol{X}^{(k)}), \quad A^{(k)}(t) = g(t, \boldsymbol{X}^{(k)}) - \overline{g}(t) \tag{21}$$

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

$$\widehat{C}(t, t_\ell^i) = \frac{1}{n-1}\sum_{k=1}^{n} A^{(k)}(t) A^{(k)}(t_\ell^i) \tag{22}$$

Ensemble Kalman smoothing

- Define the Kalman gain for each t in the time-window:

$$K(t) = \frac{\widehat{C}(t, t_\ell^i)}{\widehat{C}(t, t_\ell^i) + R^i} \tag{23}$$

- 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:

$$g(\boldsymbol{X}^{(k)}) \leftarrow g(\boldsymbol{X}^{(k)}) - K_t \cdot d_\ell^{(k)} \tag{24}$$

- Process repeated sequentially across all observation times t_ℓ^i, for all data groups i

Acharki, N., Bertoncello, A., and Garnier, J. (2023). Robust prediction interval estimation for gaussian processes by cross-validation method. *Computational Statistics & Data Analysis*, 178:107597.

AIAA (2021). Digital Twin: definition & value - An AIAA and AIA Position Paper.

Angelopoulos, A. N. and Bates, S. (2023). Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning*, 16(4):494–591.

Barber, R. F., Candès, E. J., Ramdas, A., and Tibshirani., R. J. (2021). Predictive inference with the jackknife+. *Annals of Statistics*, 49:486–507.

Baudin, M., Dutfoy, A., Iooss, B., and Popelin, A. (2017). Open turns: An industrial software for uncertainty quantification in simulation. In Ghanem, R., Higdon, D., and Owhadi, H., editors, *Handbook on Uncertainty Quantification*, pages 2001–2038.

Cordier, T., Blot, V., Lacombe, L., Morzadec, T., Capitaine, A., and Brunel, N. (2023). Flexible and Systematic Uncertainty Estimation with Conformal Prediction via the MAPIE library. In *Conformal and Probabilistic Prediction with Applications*.

Da Veiga, S. (2015). Global sensitivity analysis with dependence measures. *Journal of Statistical Computation and Simulation*, 85:1283–1305.

David, F. (1999). Three Dimensional Thermal-Hydraulic Simulation In Steam Generators With THYC Exchangers Code - Application To The UTSG Model 73/19. Proceedings of the Ninth International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-9).

De Rocquigny, E., Devictor, N., and Tarantola, S., editors (2008). *Uncertainty in industrial practice - A guide to quantitative uncertainty managment*. Wiley and Sons.

Deri, E., Varé, C., and Wintergerst, M. (2021). *Development of Digital Twins of PWR Steam Generators: Description of Two Maintenance-Oriented Use Cases*, volume Volume 1: Operating Plant Challenges, Successes, and Lessons Learned; Nuclear Plant Engineering; Advanced Reactors and Fusion; Small Modular and Micro-Reactors Technologies and Applications of *International Conference on Nuclear Engineering*.

Evensen, G. and Van Leeuwen, P. J. (2000). An ensemble Kalman smoother for nonlinear dynamics. *Monthly Weather Review*, 128(6):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.

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

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. (2025a). Conformal approach to Gaussian process surrogate evaluation with marginal coverage guarantees. *Journal of Machine Learning for Modeling and Computing*.

Jaber, E., Chabridon, V., Remy, E., Baudin, M., Lucor, D., Mougeot, M., and Iooss, B. (2025b). Sensitivity Analyses of a Multi-Physics Long-Term Clogging Model For Steam Generators. *International Journal for Uncertainty Quantification*, 15:27–45.

Jaber, E., Remy, E., Chabridon, V., Mougeot, M., and Lucor, D. (2025c). Fusion of heterogeneous data for robust degradation prognostics. *arXiv:2506.05882*.

Keller, M., Damblin, G., Pasanisi, A., Schumann, M., Barbillon, P., Ruggeri, F., and Parent, E. (2022). Validation of a computer code for the energy consumption of a building, with application to optimal electric bill pricing. *Econometrics*, 10(4).

Le Maître, O. P. and Knio, O. M. (2010). *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics*. Scientific Computation. Springer.

Lefebvre, L., Segond, M., Spaggiari, R., Le Gratiet, L., Deri, E., Iooss, B., and Damblin, G. (2023). Improving the Predictivity of a Steam Generator Clogging Numerical Model by Global Sensitivity Analysis and Bayesian Calibration Techniques. *Nuclear Science and Engineering*, 197(8):2136–2149.

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.

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.

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

Vachtsevanos, G., Lewis, F., Roemer, M., Hess, A., and Wu, B. (2006). *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. John Wiley & Sons.

Vovk, V., Gammerman, A., and Shafer, G. (2005). *Algorithmic Learning in a Random World*. Springer, New York.