VALUE-OF-INFORMATION-BASED OPTIMIZATION OF GREY-BOX MODELS FOR COMPUTATIONAL RISK ASSESSMENT OF CYBER-PHYSICAL SYSTEMS

Juan-Pablo Futalef[1], Francesco Di Maio[1,*], Pierre Beaurepaire[2], Enrico Zio[1,3]

[1]Energy Department, Politecnico Di Milano, Italy

[2] Université Clermont Auvergne, Clermont Auvergne INP, CNRS, Institut Pascal, France

[3]Centre de Recherche sur les Risques et les Crises, Mines Paris - PSL, France

Abstract: The complexity of Cyber-Physical Systems (CPSs) requires that their safety be evaluated by Computational Risk Assessment (CRA). The very large computational challenge can be addressed by Grey-Box Models (GBMs), which integrate first-principles White-Box Models (WBMs) and data-driven Black-Box Models (BBMs): different WBMs or BBMs can be used to model each CPS subsystem, which leads to a combinatorial number of possible GBM alternatives that come with different trade-offs between computational burden and physical fidelity. In this work, we propose a Value-of-Information (VoI)-based procedure to identify the best GBM for CPS CRA. We show the procedure by application to an Integrated Power and Telecommunication (IP&TLC) infrastructure: VoI-driven GBM alternatives are evaluated and those that reduce the computational burden while keeping accurate in computing the risk metrics of interest are identified.

Keywords: Cyber-Physical Systems, Computational Risk Assessment (CRA), Grey-Box Modelling, Hierarchical Modelling, Value-of-Information (VoI).

[*]Corresponding author Francesco Di Maio (francesco.dimaio@polimi.it).

Address: Via La Masa 32, 20156, Milan, Italy.

1 INTRODUCTION

Cyber-Physical Systems (CPSs) integrate cyber and physical subsystems to streamline operations across critical domains like energy, transportation and manufacturing (Lin et al., 2017; Lv et al., 2021). Their safety can be evaluated by Computational Risk Assessment (CRA), which relies on accidental scenarios simulation and post-processing (Di Maio & Zio, 2018). CRA is challenged by the high computational burden of the high-fidelity first-principles CPS models used (Cassottana et al., 2023; Zio, 2018). Advanced sampling methods have been introduced to accelerate accident scenarios simulation, but these may not be sufficient in practice due to the curse of dimensionality and the elevated complexity of the systems (Beck & Zuev, 2017; Futalef et al., 2025; Turati et al., 2016).

Modelling of CPS can resort to:

- *White-Box Models* (WBMs), also known as *physics-based* models; these are high-fidelity, first-principles models based on a deep understanding of the system dynamics to formulate the set of analytical equations that describe the underlying CPS behavior (de Paula Ferreira et al., 2020; Guo et al., 2021). While highly interpretable, WBMs are computationally burdensome for large CPSs (Colabianchi et al., 2021; Zografopoulos et al., 2021). Model simplifications, e.g., linearizing or reducing model granularity, can reduce the computational burden, but at the expense of their accuracy (Zio, 2018; Zografopoulos et al., 2021).
- *Black-Box Models* (BBMs), also known as data-driven or surrogate models; these are empirical models built and calibrated (trained) on data (e.g., monitoring data from sensors or data provided by high fidelity simulations) to approximate input-output relationships in terms of, for example, system response (Mao et al., 2024) and control logic (Pinto et al., 2021) (the interested reader may also refer to the review papers (Deb et al., 2017) and (Zaparoli Cunha et al., 2023)); once trained, BBMs often allow a significant reduction of computation times, but their construction and training require large

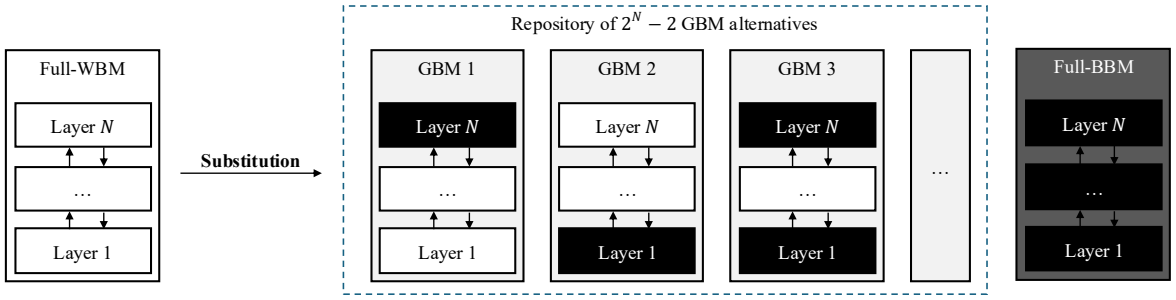

Figure 1 Hierarchical CPS model (left) and GBM alternatives (right)

amounts of data, they lack interpretability (hence "black box") and calibration errors cannot be avoided (Hastie et al., 2009; Zeigler et al., 2019)."

- *Grey-Box Models* (GBM); these models leverage both WBMs and BBMs in a unified framework, thus benefiting from the advantages of both WBMs (e.g., interpretability) and BBMs (e.g., speed of calculation) (Giselle Fernández-Godino et al., 2019; Thelen et al., 2022).

In this work, we construct GBMs of CPS by combining WBMs and BBMs of the different CPS subsystems (Figure 1). The CPS model is a hierarchy of $N \geq 2$ layers, each corresponding to a CPS subsystem made of one or many components (Futalef et al., 2022, 2025), and there is a combinatorial number of $2^N - 2$ possible GBM alternative configurations. Each alternative configuration of WBMs and BBMs offers a unique trade-off between computational burden and physical fidelity, and identifying the best alternative can be a challenge: a GBM configuration with a large majority of WBMs may lead to a prohibitive computational burden, whereas an excessive use of BBMs in the GBM may introduce approximation errors that degrade the accuracy of systemic risk metrics and performance indicators estimates.

In this paper, we propose a procedure to identify the GBM (i.e., the combination of WBMs and BBMs) that optimally balances computational burden and fidelity for the CRA of CPS. The procedure makes use

of the Value-of-Information (VoI) (Fenwick et al., 2020; Hoseyni et al., 2021; Malings & Pozzi, 2018) to efficiently quantify the expected trade-off gain (or loss) obtained by adopting a particular GBM relative to a Ground-Truth (GT) dataset of CPS responses generated using a full-WBM. The GBM alternative that maximizes expected performance optimally balances computational burden and fidelity.

To efficiently explore the GBM alternatives, we propose an adaptive selection mechanism that, based on VoI, allocates computational resources into refining the estimates of both computational load and fidelity of the sole alternatives that offer the best trade-offs. First, we split the GT data into batches of representative scenarios of normal and accident CPS responses. Then, at each iteration, the procedure (i) computes the VoI of each alternative from prior knowledge, (ii) selects probabilistically an alternative with large VoI, (iii) collects evidence of its computational burden and fidelity by simulating a batch of scenarios, and (iv) updates posterior estimates efficiently through conjugate priors (Robert, 2007). The procedure ends when the VoI increments are small or all batches are used, returning the GBM that maximizes expected trade-off.

We demonstrate the VoI-driven approach on an Integrated Power and Telecommunication (IP&TLC) system (Di Maio et al., 2022) that integrates three subsystems: a Power Grid (PG), a Telecommunication Network (TLCN) and a Control Center (CC). This configuration yields six GBM alternatives in addition to the original full-WBM and full-BBM. Fidelity of the configuration is measured in terms of its capability to estimate Energy Not Supplied (ENS) as risk metric with respect to the full-WBM. An exhaustive search shows that an optimal GBM alternative exists that outperforms the original full-WBM, capable of lowering the computational burden by more than 94% and retaining the same fidelity. The VoI-based selection framework identifies the same optimal GBM in <5% of the computational time needed for the exhaustive search.

The structure of the paper is organized as follows. Section 2 presents the concept of GBM for CPS evaluation. Section 3 introduces the VoI-based GBM selection procedure and the proposed solution method. Section 4 presents the realistic case study based on the IP&TLCN. Finally, Section 5 concludes the research findings and proposes future research directions.

2 CYBER-PHYSICAL SYSTEMS GREY-BOX MODELING

Let us consider a CPS made of $N \geq 2$ subsystems that interact in time under a fixed-step discrete-time setting: each instant k corresponds to a continuous time $t_k = \Delta t \cdot k + t_0$, where $\Delta t > 0$ is the fixed time step and t_0 the initial time. At each k-th instant, the j-th CPS subsystem is characterized by its state vector $\vec{x}_j(k)$, $j = 1, \dots, N$, with initial condition $\vec{x}_j(0) = \vec{x}_j^0$, whose entries can be continuous and discrete variables: continuous variables are physical quantities represented by the vector $\vec{\lambda}_j(k)$, whereas discrete variables are binary component states (e.g., 1 for "up" or 0 for "down"), represented by the vector $\vec{\pi}_j(k)$:

$$\vec{x}_j(k) = \begin{pmatrix} \vec{\lambda}_j(k) \\ \vec{\pi}_j(k) \end{pmatrix} \tag{1}$$

The evolution in time of $\vec{x}_j(k)$ is governed by a nonlinear one-step state transition function f_j:

$$\vec{x}_j(k+1) = f_j\left(\vec{x}_j(k), \vec{u}_j(k), \vec{e}_j(k)\right) \tag{2}$$

where $\vec{u}_j(k)$ are the subsystem-level controlled inputs (e.g., devices settings) and $\vec{e}_j(k)$ the uncontrolled external stimuli (e.g., demands, disturbances, environmental conditions). Optionally, noise can be injected in the transition to account for epistemic uncertainty, e.g., unmodeled dynamics and/or imperfect parameters settings.

The CPS state vector $\vec{x}(k)$ is such that:

$$\vec{x}(k) = \begin{pmatrix} \vec{x}_1(k) \\ \vec{x}_2(k) \\ \vdots \\ \vec{x}_N(k) \end{pmatrix} \tag{3}$$

and its evaluation in time follows a state transition function $F(\cdot)$ that accounts for the interactions between the N subsystems:

$$\vec{x}(k+1) = F\left(\vec{x}(k), \vec{u}(k), \vec{e}(k)\right), \tag{4}$$

where $\vec{u}(k)$ and $\vec{e}(k)$ are the CPS-level controlled variables and uncontrolled external stimuli, respectively. The simulated trajectories obtained recursively solving Eq. (4) over n^{miss} instants produce a scenario $Y = \left(y_0, y_1, y_2, \dots, y_{n^{miss}}\right)$, where each $y_k = \{t_k, \vec{x}(k), \vec{u}(k), \vec{e}(k)\}$ stores the process variable at instant k.

To simulate the CPS dynamics with Eq. (4), one can resort to well-known off-the-shelf suites like Matlab (MathWorks, 2025) and Modelica (Modelica Association, 2025), or custom co-simulation approaches (Hansen et al., 2024). In this work, we resort to the CPS-oriented co-simulation approach in (Futalef et al., 2025) (see Appendix A for a summary of key concepts) as, with respect to others, it offers a standardized way to handle exchanges of information between subsystems, whether they are physical (e.g., energy, mass) or digital (e.g., measurements, control signals), thereby enabling modularity by relieving the analyst from redefining the coordination logic whenever a subsystem model changes.

Modularity enables implementing each f_j as either a WBM or a BBM. In practice, several WBMs and BBMs can be considered for the same subsystem (e.g., by training different BBMs), each one with a different fidelity; this, however, leads to an unmanageable combinatorial explosion of the number of modeling alternatives to ensemble into GBMs. In this work, to keep the number of GBMs controlled, we assume only single WBMs and the best BBMs (in terms of training performance) to be available for each subsystem; this assumption needs to be verified to be conservative for the practice of reliability assessment. Thus, the GBM repository consists of 2^N model alternatives $\mathcal{F} = \left\{F_1, F_2, \dots, F_{2^N}\right\}$, where each F_i implements a strategy for estimating $F(\cdot)$ in Eq. (4). A plan vector $\vec{s}_i = \left(\vec{s}_i^{(1)}, \vec{s}_i^{(2)}, \dots, \vec{s}_i^{(N)}\right)$ encodes the model types of each j-th sub-system in model alternative F_i:

$$\vec{s}_i^{(j)} = \begin{cases} 0, & f_j \text{ is a WBM} \\ 1, & f_j \text{ is a BBM} \end{cases} \tag{5}$$

Under this convention, the all-zeros configuration defines a full-WBM (i.e., each f_j is a WBM), the all-ones configuration defines a full-BBM (i.e., each f_j is a BBM), and the remaining $2^N - 2$

configurations define GBMs that combine WBMs and BBMs in one or more subsystems, as shown in Figure 1.

As explained earlier, a good combination of WBMs and BBMs can be advantageous since the presence of BBMs, well trained on monitoring data, can speed up the simulation of some expensive WBMs, reducing the overall computational burden of the global CPS model. Nevertheless, selecting a suitable GBM introduces two technical challenges: (i) there is a combinatorial explosion of possible GBM alternatives and (ii) the trade-off between the gains in computational efficiency and the reduction of model fidelity when certain components are modeled by BBMs is, a priori, unknown. To tackle this issue, an optimization problem must be defined and solved, as we shall see in what follows.

3 GREY-BOX MODEL SELECTION FOR CPS CRA

3.1 PROBLEM SETTING

The performance of each model alternative $F_i \in \mathcal{F}$ is quantified by two losses: computational load L_i^1 (see Section 3.3) and fidelity L_i^2 (see Section 3.4) with respect to a Ground-Truth (GT) dataset of CPS scenarios $\mathcal{D}_{GT} = \{Y_1, Y_2, \dots, Y_{n^{GT}}\}$ where, in our case, each scenario $Y_l \in \mathcal{D}_{GT}$ is generated using a full-WBM $F_\emptyset$. The losses for $F_\emptyset$, denoted $L_\emptyset^1$ and $L_\emptyset^2$, are known a priori. Instead, the losses for the alternatives $F_i \in \mathcal{F} \backslash \{F_\emptyset\}$ are unknown. In practice, L_i^1 and L_i^2 are:

i. uncertain, as their values depend on the quality and quantity of information available, which can only be obtained from simulations;

ii. conflicting, since the inclusion of BBMs can introduce approximation errors and epistemic uncertainties that propagate and interact at systemic level, yielding solutions that, although computationally efficient, come with lower fidelity.

To handle uncertainty, we assume that $L_i^1 \sim \mathcal{N}\left(\mu_i^{L_1}, (\sigma_i^{L_1})^2\right)$ and $L_i^2 \sim \mathcal{N}\left(\mu_i^{L_1}, (\sigma_i^{L_1})^2\right)$ so that each F_i is associated with mean and variance distributions parameters $\vec{\theta}_i^{L_1} = \left(\mu_i^{L_1}, (\sigma_i^{L_1})^2\right)$ and $\vec{\theta}_i^{L_2} = \left(\mu_i^{L_1}, (\sigma_i^{L_1})^2\right)$ (see the following Sections 3.3 and 3.4). To handle the conflicting behavior of the objectives, we define an aggregated loss $L_i \triangleq L\left(\vec{\theta}_i^{L_1}, \vec{\theta}_i^{L_2}\right)$ (see Eq. (8) in following Section 3.2) that quantifies the trade-off between L_i^1 and L_i^2 according to their normal distribution parameters.

The goal is to identify the model alternative $F^* \in \mathcal{F}$ that optimally balances computational load L_i^1 and fidelity L_i^2 by minimizing L_i. Since L_i^1 and L_i^2 are unknown initially (except for $L_\emptyset^1$ and $L_\emptyset^2$), an efficient way to accurately estimating them is needed. For this, we (i) enable collecting evidence of computational burden and fidelity from batches of simulations to update their distribution parameters $\vec{\theta}_i^{L_1}$ and $\vec{\theta}_i^{L_2}$ (e.g., through normal-normal conjugacy), and (ii) frame the selection procedure as a Value-of-Information (VoI) decision-making problem (Howard, 1966) to guide the exploration.

In general, VoI is defined as the difference between the expected cost of an optimal decision with prior information (i.e., some reference optimal F_r^*) and the expected cost of an optimal decision after collecting new information (i.e., z_i of model alternative F_i):

$$\mathrm{VoI}(F_i) = \mathbb{E}\{L_r^*\} - \mathbb{E}_{z_i}\{L_i^*\} \tag{6}$$

In our case, VoI measures the expected loss decrease when collecting evidence of model alternative F_i, enabling targeted refinement towards the alternatives that are more likely to improve the optimal decision. Then, since a lower L_i implies larger VoI, the selection problem is reframed as identifying the model alternative F^* that maximizes VoI, subject to $\mathrm{VoI}(F_i) > 0$ (ensuring improvement), over the repository $\mathcal{F}$ of all possible model alternatives:

$$\begin{aligned} F^* = \arg\max_{F_i \in \mathcal{F}} & \mathrm{VoI}(F_i) \\ s.t. \quad & \mathrm{VoI}(F_i) > 0 \end{aligned} \tag{7}$$

VoI is a concept rooted in Bayesian statistical decision theory that quantifies the net benefit of acquiring (often costly) new information before committing to a decision. Recent applications of VoI include sensor placement (Hoseyni et al., 2021; Malings & Pozzi, 2018), data acquisition (Alawad & Kraemer, 2022; Liu et al., 2022) and resource allocation (Jarwan et al., 2022; Tong et al., 2024).

3.2 VOI-BASED PROCEDURE FOR GBM SELECTION

The performance of GBMs (in terms of both computational load and fidelity) is known only after running (potentially costly) simulations; thus, solving Eq. (7) by screening all model alternatives (i.e., simulating each $Y_l \in \mathcal{D}_{GT}$ with each F_i to compute $\mathrm{VoI}(F_i)$) results impractical. Adaptive selection procedures, e.g., racing for lazy-learner selection (Maron & Moore, 1997) and its variants for algorithm tuning (López-Ibáñez et al., 2016), have addressed the unknown error of models using Bayesian updates at runtime to remove inferior models; however, as in traditional selection, these adaptive methods only focus on fidelity. When computational burden is also considered as an explicit objective, as in our case, such methods become impractical.

In this work, we propose the adaptive selection procedure in Algorithm 1 to efficiently find the optimal GBM F^*. The procedure guides the exploration and refines the estimates of computational burden and fidelity of the sole alternatives that offer the best trade-offs. First, $\mathcal{D}_{GT}$ is partitioned into batches $\mathcal{B}_{GT}$ of representative scenarios using the `GTBatches` procedure (Figure 2) by (i) identifying accident scenarios in $\mathcal{D}_{GT}$ using some relevant risk metric $r(\cdot)$ (see Section 3.4), (ii) randomly splitting the normal scenarios into $n^B = \lceil n^{GT}/n^{batch} \rceil$ pre-batches and (iii) randomly inserting enough accident scenarios into the batches so that at least $p_{\%}^{acc}\%$ of the scenarios in each batch are accidental.

Since the mean and variance of L_i^1 and L_i^2 are unknown for each F_i, we place a Normal-Inverse-Gamma (NIG) prior on each loss (see Appendix B). A generic NIG prior is made of four parameters $\vec{\beta} = \left(\beta^{(1)}, \beta^{(2)}, \beta^{(3)}, \beta^{(4)}\right)$ that parametrize the uncertainty on the mean and the variance. At each iteration

Algorithm 1 Iterative VoI-driven GBM selection

Inputs: model repository $\mathcal{F}$; GT dataset $\mathcal{D}_{GT}$; risk metric $r(\cdot)$; batch size n^{batch}; batch accidental scenario percentage $p_{\%}^{acc}$; ranking skips n^{out}; NIG prior values $\{\vec{\beta}_{i,0}^{L_1}, \vec{\beta}_{i,0}^{L_2}\}_{i=1:2^{N-1}}$; predictive data samples M.

Outputs: Optimal GBM F^*

1. Partition GT data: $\mathcal{B}_{GT} = \text{GTBatches}\left(\mathcal{D}_{GT}, n^{batch}, p_{\%}^{acc}, r(\cdot)\right)$; (Figure 2)
2. Initialize $\kappa \leftarrow 0$
3. For each $F_i \in \mathcal{F}\backslash\{F_\emptyset\}$, initialize NIG prior parameters:
   a. $\vec{\beta}_{i,\kappa}^{L_1} \leftarrow \vec{\beta}_{i,0}^{L_1}$;
   b. $\vec{\beta}_{i,\kappa}^{L_2} \leftarrow \vec{\beta}_{i,0}^{L_2}$;
4. while $\kappa \leq n^B$:
   a. For each $F_i \in \mathcal{F}\backslash\{F_\emptyset\}$:
      i. $\vec{\hat{\theta}}_{i,\kappa}^{L_j} \leftarrow$ NIGtoNormal $\left(\vec{\beta}_{i,\kappa}^{L_j}\right)$, for $j = 1,2$; (Appendix B)
      ii. $L_i \leftarrow L\left(\vec{\theta}_{i,\kappa}^{L_1}, \vec{\theta}_{i,\kappa}^{L_2}\right)$; (Eq.(8))
   b. $\text{VoI}_\kappa(F_i) \leftarrow L_i^* - L_r^*$; for each $F_i \in \mathcal{F}\backslash\{F_\emptyset\}$ (Eq. (6))
   c. Select F_r from $\mathcal{F}$ with probability $\propto \text{VoI}_\kappa(F_i)$;
   d. Simulate batch b_κ using F_r to obtain evidence $z_r^{L_1}$ and $z_r^{L_2}$; (Sections 3.3 and 3.4)
   e. Update NIG priors for F_r:
      i. $\vec{\beta}_{i,\kappa+1}^{L_j} \leftarrow$ NIGUpdate $\left(\vec{\beta}_{i,\kappa}^{L_j}, z_i^{L_j}\right)$, for $j = 1,2$; (Appendix B)
   f. Temporarily remove F_r from $\mathcal{F}$ for n^{out} iterations;
   g. $\kappa \leftarrow \kappa + 1$;
5. Return $F^* = \arg\max_i \text{VoI}(F_i)$

κ, we associate each alternative F_i with the prior NIG parameters $\vec{\beta}_{i,\kappa}^{L_1}$ and $\vec{\beta}_{i,\kappa}^{L_2}$ for L_i^1 and L_i^2, respectively. It is worth mentioning that the initial setting of NIG priors may greatly impact the search of the best GBM. While the adaptive search is designed to refine these estimates throughout the iterations, to accelerate early convergence towards reasonable settings, the analyst is suggested to provide an educated guess for the priors using all available information from standalone BBM training performances (e.g., by estimating their computational burden reduction and expected fidelity loss) and eventually set the variance of the NIG priors parameters large enough to reflect high initial uncertainty. Then, to update these parameters, we collect evidence $z_i^{L_1}$ for the computational burden (see Section 3.3) and $z_i^{L_2}$ for fidelity (see Section 3.4) by simulating a batch from $\mathcal{B}_{GT}$ with some alternative F_i; then, the NIG priors are recursively updated using

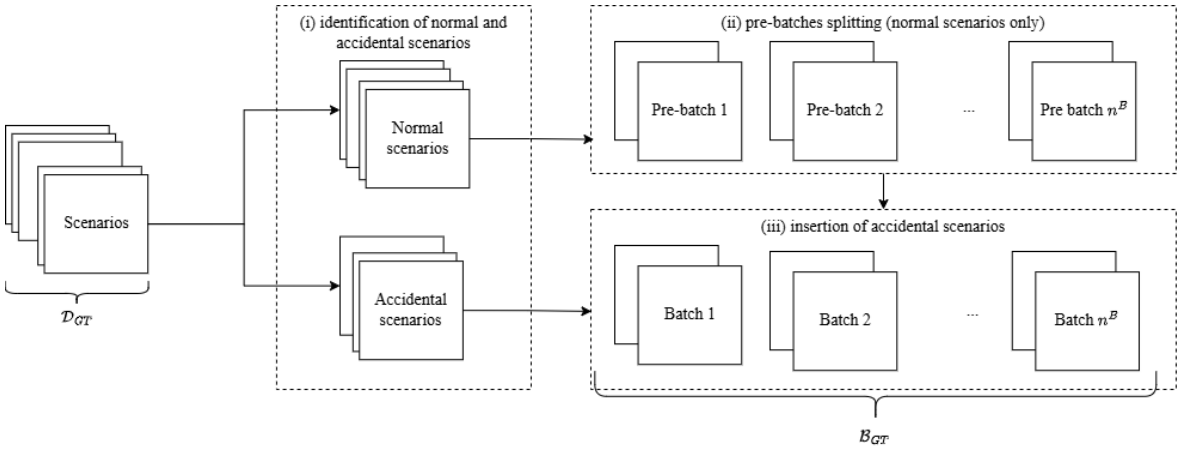

Figure 2 Illustration of the `GTBatches` procedure to construct batches representative of normal and accident conditions

the method $\vec{\beta}_{i,\kappa+1}^{L_j}$=NIGUpdate$\left(\vec{\beta}_{i,\kappa}^{L_j}, z_i^{L_j}\right)$, thereby producing updated priors for the next iteration. Since the expected values of the normal parameters $\vec{\theta}_i^{L_1}$ and $\vec{\theta}_i^{L_2}$ can be estimated from the NIG parameters using the method $\vec{\theta}_i^{L_j} = \texttt{NIGtoNormal}\left(\vec{\beta}_i^{L_j}\right)$, the NIG updates produce refinements on the normal posteriors.

Considering the above parametrization, we quantify the trade-off between some generic L_i^1 and L_i^2 via the aggregated loss L_i in Eq. (8) that accounts for (i) the conflicting nature of computational burden and fidelity and (ii) the uncertainty in the estimates:

$$L_i \triangleq L\left(\vec{\theta}_i^{L_1}, \vec{\theta}_i^{L_2}\right) = \omega_1\lambda_1\left(\mu_i^{L_1}\right) + \omega_2\lambda_2\left(\mu_i^{L_2}\right) + \sqrt{\lambda_1\left(\sigma_i^{L_1}\right)^2 + \lambda_2\left(\sigma_i^{L_2}\right)^2}, \tag{8}$$

where $\lambda_1(\cdot), \lambda_2(\cdot) > 0$ are scaling functions, and $\omega_1, \omega_2 \in [0,1]$ with $\omega_1 + \omega_2 = 1$ are weights reflecting the relative importance of computational load and fidelity. Penalizing the standard deviations is intended to reduce uncertainty at early iterations, followed by refinements of the mean values.

To avoid sub-modularity (in our context, it means that the marginal benefit obtained by including BBMs into the GBM configuration can diminish rather than increase (Hoseyni et al., 2019; Malings & Pozzi,

2019)), at each iteration κ, Algorithm 1 selects the next model to refine with a probability proportional to its current VoI, thereby allocating more computational effort to the most promising alternatives. To enable exploration of more alternatives, once an alternative is selected, the procedure removes it for $n^{out} > 0$ iterations. As a result, by selectively refining only the most promising alternatives, we rapidly converge to the GBM alternatives with largest VoI (thus, avoiding purely greedy updates from getting stuck in suboptimal solutions (Malings & Pozzi, 2019; Memarzadeh & Pozzi, 2016).

3.3 COMPUTATIONAL BURDEN

Computational burden L_i^1 measures the ratio of execution time per simulation time needed by F_i to solve Eq. (4). It is defined as the slope m_i in the (approximately) linear relation

$$t_k \approx \underbrace{m_i}_{L_i^1} \cdot t_k^{CPU} \tag{9}$$

where, at each k-th instant, t_k^{CPU} is the cumulative execution time and t_k the simulated time horizon. It is assumed that $m_i \sim \mathcal{N}\left(\mu_i^{L_1}, (\sigma_i^{L_1})^2\right)$, thus associated with the normal parameters $\vec{\theta}_i^{L_1}$ of the computational burden.

For a given scenario $Y_l \in \mathcal{D}_{GT}$, we can obtain an estimation $\widehat{m}_{i,l}$ of m_i by (i) simulating Y_l with F_i, (ii) collecting the points $\left\{\left(t_k, t_k^{CPU}\right)\right\}_{k=0:n_l^{miss}}$ and (iii) computing the least-squares slope:

$$\widehat{m}_{i,l} = \frac{\sum_{k=0}^{n_l^{miss}} t_k t_k^{CPU}}{\sum_{k}^{n_l^{miss}} (t_k)^2} \tag{10}$$

Repeating this procedure over a batch of n^{batch} scenarios yields the evidence $z_i^{L_1} = \left\{\widehat{m}_i^1, \widehat{m}_i^2, \dots, \widehat{m}_i^{n^{batch}}\right\}$, which can be used to update the NIG $\vec{\beta}_i^{L_1}$ and, thus, the normal $\vec{\theta}_i^{L_1}$ parameters.

3.4 FIDELITY

Fidelity L_i^2 is measured in terms of Lack-of-Fit (LoF) ϵ_i, which is defined as the Kolmogorov-Smirnov (KS) statistic (Gibbons & Chakraborti, 2020) between the probability distributions of the risk values

computed from $\mathcal{D}_{GT}$ and by simulation with F_i. Risk is quantified along some scenario $Y_l = \left(y_0, y_1, y_2, \dots, y_{n_l^{miss}}\right)$ through some risk metric $r(y_k)$; this metric is aggregated for the scenario (e.g., via summation, averaging or other methods) to obtain the scenario risk value: $r_{l,i}$ if Y_l was simulated using model alternative F_i or, instead, $r_{l,\emptyset}$ if Y_l was extracted from $\mathcal{D}_{GT}$. This metric can be used to label accident scenarios, for instance, using a traditional threshold approach (e.g., when $r_{l,i} \geq 0$).

Ideally, $r_{l,i} = r_{l,\emptyset}$ if F_i models well the CPS. In practice, $r_{l,i} \neq r_{l,\emptyset}$ because of (i) uncertainty and (ii) errors introduced by BBMs. From a statistical point of view, $r_{l,i}$ and $r_{l,\emptyset}$ are samples from two different distributions, p_i^{risk} and $p_\emptyset^{risk}$, respectively. We can measure empirically the discrepancy between these distributions through the KS statistic by, first, extracting some batch from $\mathcal{B}_{GT}$ and compute the aggregated scenario risk values $\{r_{l,\emptyset}\}_{l=1:n^{\text{batch}}}$; then, those scenarios are simulated using F_i and the corresponding aggregated risk metrics $\{r_{l,i}\}_{l=1:n^{\text{batch}}}$ are computed. By obtaining the empirical cumulative distributions $\hat{P}_i^{risk}$ and $\hat{P}_\emptyset^{risk}$ of the corresponding aggregated risk metrics, the KS statistic of the batch is:

$$\hat{\epsilon}_i = \delta\left(\hat{P}_i^{risk}, \hat{P}_\emptyset^{risk}\right) \tag{11}$$

where $\delta(\cdot)$ is the maximum distance between $\hat{P}_i^{risk}$ and $\hat{P}_\emptyset^{risk}$ (a perfect fit yield $\hat{\epsilon}_i = 0$, whereas $\hat{\epsilon}_i = 1$ is a complete mismatch). This unique measurement represents the collected evidence of fidelity $z_i^{L_2} = \{\hat{\epsilon}_i\}$, which can, then, be used to update the NIG $\vec{\beta}_i^{L_2}$ and, thus, normal $\vec{\theta}_i^{L_2}$ parameters.

4 Case Study – Integrated Power and Telecommunication Network

4.1 Model Details and GBM library

We consider an Integrated Power and Telecommunication Network (IP&TLCN) system (Futalef et al., 2025) comprising three subsystems, as illustrated in Figure 3. At each time t_k, the Power Grid (PG) must supply a set of customers with power. For this, Smart Monitoring Devices (SMDs) measure and transmit electrical quantities through the Telecommunication Network (TLCN) to the Control Center (CC); in turn,

the CC computes and sends back control actions to regulate the PG by accounting for load changes or network reconfigurations (e.g., due to unavailable components). It is worth noting that, although the IP&TLCN is made of only three layers, the numerous cyber-physical interdependencies (e.g., non-linear power flows, time delays, non-convex optimal control logic, among others) yield properties that are typical of CPSs, making it a representative case study with enough complexity and realism to justify the need of a GBM selection method for enabling CRA.

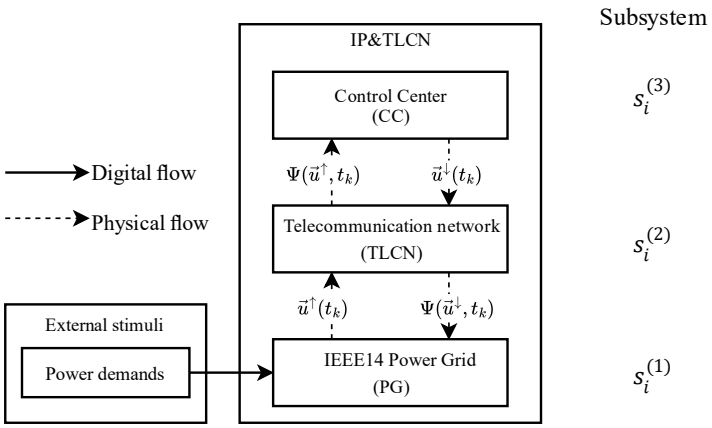

Figure 3 Hierarchical representation of the IP&TLCN system (Futalef et al., 2025)

Using the hierarchical co-simulation approach described in Appendix A , we build a noisy full-WBM $F_{\emptyset}$ for the IP&TLCN (see Appendix C for details). We use Energy Not Supplied (ENS) as a risk metric, which measures the mismatch between requested and delivered power to customers at each time instant; the risk aggregation method is a numerical integration, so that a scenario is labelled as accidental when the aggregated ENS is greater than zero.

Table 1 Specification of the different model alternatives in the repository $\mathcal{F}$

Model alternative	$\vec{s}_i = \left(s_i^{(1)}, s_i^{(2)}, s_i^{(3)}\right)$	Model name
F_0	$\vec{s}_0 = (0,0,0)$	Full-WBM
F_1	$\vec{s}_1 = (0,0,1)$	GBM1
F_2	$\vec{s}_2 = (1,0,0)$	GBM2
F_3	$\vec{s}_3 = (1,0,1)$	GBM3
F_4	$\vec{s}_4 = (0,1,0)$	GBM4
F_5	$\vec{s}_5 = (0,1,1)$	GBM5
F_6	$\vec{s}_6 = (1,1,0)$	GBM6
F_7	$\vec{s}_7 = (1,1,1)$	Full-BBM

The three subsystems yield a repository $\mathcal{F}$ with a total of $2^3 = 8$ model alternatives, including the full-WBM $F_\emptyset$, a full-BBM and six possible GBMs (as shown in Table 1). The BBMs are selected according to established solutions found in the literature and, then, tailored to the case study:

- for CC and PG, we consider fully connected feedforward Neural Networks (NNs) for learning the Power Flow (PF) and Optimal PF (OPF) mappings as in (Fioretto et al., 2020), with the resulting training performances shown in Table 2;
- for the TLCN, we consider a binary Gradient Boosting Classifier (GBC) for learning the End-to-End (E2E) communication between cyber devices as in (Futalef et al., 2024), with the resulting training performance shown in Table 3.

Training has been performed using the generated GT data $\mathcal{D}_{GT}$, with $n_{GT} = 3{,}607$ scenarios simulated with F_0 according to (Futalef et al., 2025). Each scenario spans over a range of $n^{miss} = 3$ to $n^{miss} = 384$ simulation instants with $\Delta t = 15$ [min] (i.e., mission times between 0.75 to 96 hours). From these generated data, the full-WBM obtains a computational effort $L_\emptyset^1 = 4.64\text{e}-4$ and a fidelity $L_\emptyset^2 = 0$ (comparing the exact data implies perfect fidelity).

Table 2 Architecture and training performance obtained by the BBM surrogates for the PG and CC subsystems

			Final RMSE loss		
Subsystem	Surrogate type	Architecture	Train	Validation	Test
CC	NN	$81-500-7$	3.07e-04	3.08e-04	3.42e-04
PG	NN	$63-500-111$	2.72e-4	2.78e-4	2.82e-4

– Architectures are listed as [# input] – [# hidden] – [# output]
– Activation functions: ReLU for hidden, linear for output
– RMSE: Root Mean Square Error

Table 3 Parameters and training performance obtained by the BBM surrogate for the TLCN subsystem

				ROC-AUC score	
Subsystem	Surrogate type	Estimators number	Maximum tree depth	Training	Test
TLCN	GBC	80	5	1.0	0.889

– AUC-ROC: Area Under the Receiver Operating Characteristic Curve

In Section 4.1, we first verify that an optimal GBM exists by running exhaustive simulations. To compute the loss function in Eq. (8), we set $\omega_1 = \omega_2 = 0.5$ (i.e., trade-off where computational load and fidelity have equal importance) and consider different scaling factors λ_1 and λ_2 to study their effects on the loss outcomes: Max. normalization, WBM values normalization and Min-Max normalization, obtained from the simulation outcomes. Later, in Section 4.2, we test our VoI-driven approach with the same parameters. All experiments run on a computer with the following specifications: WSL2 on Windows 11, CPU Intel i7 8750H at 2.2 GHz, GPU Nvidia GeForce GTX 1060 with CUDA enabled and 32 GB RAM memory at 2,667 MHz.

4.2 EXHAUSTIVE GBM SELECTION

To verify the existence of an optimal GBM alternative that balances computational load and fidelity, we exhaustively simulate every scenario in $\mathcal{D}_{GT}$ for each alternative described in Table 1. Even in this relatively limited case, the exhaustive approach required ~33.25 hours of sequential execution and ~14.37 hours in parallel, which highlights the need for an efficient exploration method. For each alternative F_i, we compute

L_i^1 (Eq. (10)) and L_i^2 (Eq. (11)), which are listed in Table 4; while each computational burden obtains positive variance (arising from the many simulations), the KS values are deterministic because we build the empirical distributions using all available data, thus, yielding variance zero for all cases. Using these values, we compute each L_i (Eq. (8)) and the actual VoI (Eq. (6)) considering the following scaling approaches:

Scaling 1. A maximum-based approach, where $\lambda_j(x) = x/\max\left(\mu_0^{L_j}, \mu_1^{L_j}, \dots, \mu_7^{L_j}\right)$, for $j = 1,2$.

Scaling 2. Normalization for computational load using the WBM values, i.e., $\lambda_1(x) = x/\mu_\emptyset^{L_1}$; no scaling is performed for the fidelity, since it is already between 0 and 1.

Scaling 3. A min-max normalization with:

$$\lambda_j(x) = \frac{\left(x - \min\left(\mu_0^{L_j}, \mu_1^{L_j}, \dots, \mu_7^{L_j}\right)\right)}{\max\left(\mu_0^{L_j}, \mu_1^{L_j}, \dots, \mu_7^{L_j}\right) - \min\left(\mu_0^{L_j}, \mu_1^{L_j}, \dots, \mu_7^{L_j}\right)}$$

for $j = 1,2$.

Table 4 Obtained losses, scaled losses and tradeoff gain computation for each model alternative

Model	$\mu_i^{L_1}$ ($\times 10^{-6}$)	$\sigma_i^{L_1}$ ($\times 10^{-6}$)	$\mu_i^{L_2}$ ($\times 10^{-1}$)	$\sigma_i^{L_2}$ ($\times 10$)	$\mathbb{E}\{L_\emptyset\} - \mathbb{E}\{L_i\}$					
					Scaling 1 ($\times 10^{-2}$)	Rank	Scaling 2 ($\times 10^{-2}$)	Rank	Scaling 3 ($\times 10^{-2}$)	Rank
Full-WBM	464.38	46.15	0.00	0.00	0.00	3	0.00	4	0.00	3
GBM1	26.64	4.75	0.00	0.00	**41.44**	1	**47.13**	1	**42.05**	1
GBM2	405.75	38.65	9.43	0.00	-43.56	7	-40.85	7	-43.48	7
GBM3	7.71	1.78	9.55	0.00	-6.47	5	1.44	3	-5.83	4
GBM4	528.19	34.2	0.00	0.00	-6.04	4	-6.87	6	-6.13	5
GBM5	77.05	16.33	0.00	0.00	36.67	2	41.70	2	37.21	2
GBM6	473.97	26.34	9.37	0.00	-49.72	8	-47.91	8	-49.74	8
Full-BBM	19.77	1.56	9.6	0.00	-7.91	6	-0.14	5	-7.29	6

Legend: underline is ≥ 0; **bold** is the largest

GBM1 consistently performs better than the other alternatives regardless of the scaling approach, yielding the largest loss reduction and, thus, making it the optimal choice: it reduces simulation times by

94.26% with respect to the full-WBM with the same fidelity. This occurs because the PG remains a WBM, thus preserving the underlying physics behavior that affects the state variables from which the ENS is derived. Although other alternatives are faster than GBM1 (e.g., GBM3 and the Full-BBM), their fidelity is lower, making them suboptimal with respect to GBM1. This fidelity loss may be due to the propagation of error when calculating the systemic risk metrics (see, for example, combinations GBM2, GBM3, GBM6 and the Full-BBM which adopt the PG BBM, whose physical state variables are crucial for computing the ENS).

The combinations that include the TLCN BBM (i.e., GBM4, GBM5, GBM6 and the Full-BBM) result in an increase of the computational burden. This occurs because the chosen GBC implementation, which is based on GPU acceleration, is mainly optimized for processing large amounts of data inputs simultaneously (in batches), yet less efficient when processing inputs one-by-one (as it is within the co-simulation environment here adopted) (Chen & Guestrin, 2016; Nabavinejad et al., 2022). While reducing the GBC complexity (e.g., by tuning specific hyperparameters to constrain model growth) may limit this drawback, it is here shown that the VoI-guided approach overcomes this issue thanks to adaptively driving the search towards the most beneficial combinations, naturally avoiding configurations where BBMs might introduce unexpected runtime overheads.

As an example, in Figure 4 the calculated ENS is plotted for a scenario of partial blackout and the errors obtained for the derived ENS using the constructed model alternatives F_i: while most alternatives exhibit non-zero error, alternatives GBM1, GBM4 and GBM5 show close approximations thanks to the use of a WBM in the power grid subsystem.

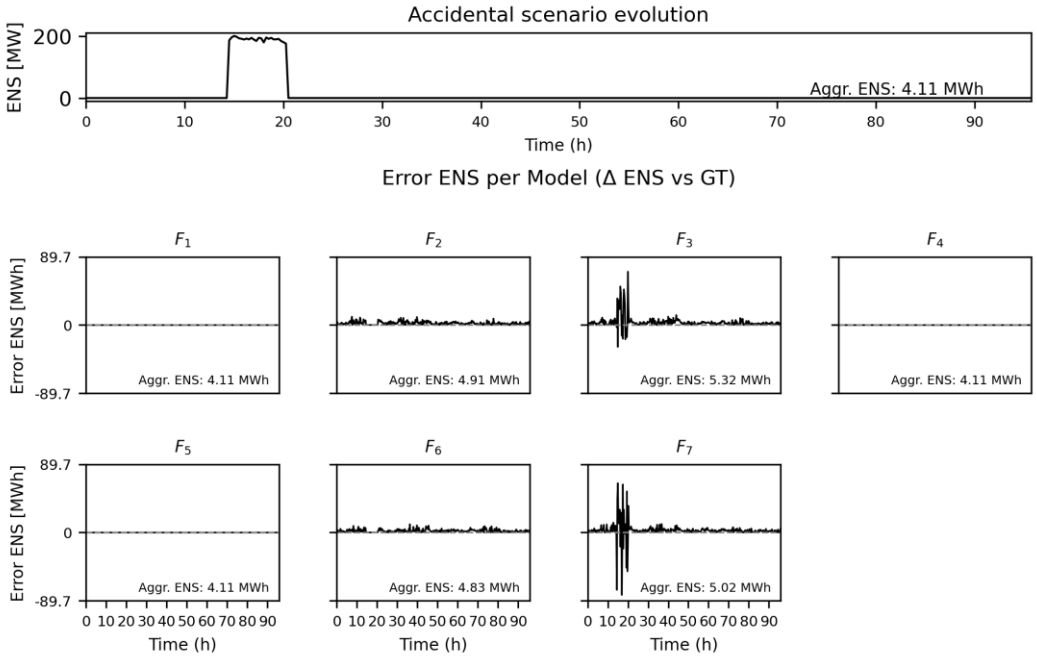

Figure 4 Prediction error and computed aggregated ENS using each model alternative for an accidental scenario

4.3 VOI-BASED SELECTION PROCEDURE

We now apply the VoI-based procedure (Algorithm 1) to explore the GBM alternatives. The algorithm parameters are adjusted as follows: $n^{batch} = 8$ (batch size), $n^{out} = 5$ (ranking skips) and $p_{\%}^{acc} = 20\%$. Considering a practical case in which only $\mathcal{D}_{GT}$ and $F_{\emptyset}$ are available, we use the scaling approach 2 (i.e., normalizing using the full-WBM values). As discussed in Section 3.2, we derive the NIG priors (Table 5) from the known performances of the Full-WBM and the individual BBM training outcomes. For the computational burden, each initial mean guess ($\beta_{1,0}^{(1)}$) is obtained by proportionally scaling the average computational burden of the original Full-WBM by the percentage of average execution time reduction offered by the BBMs in the configuration; for the fidelity, each initial mean guess ($\beta_{2,0}^{(1)}$) is obtained by linearly increasing the expected LoF based on the fraction of BBMs in the configuration, so that the Full-BBM yields a high baseline error with a LoF of 0.5. In both cases, the precision parameters ($\beta_{1,0}^{(2)}$ and $\beta_{2,0}^{(2)}$) are set based on a pre-defined epistemic uncertainty tolerance. Regarding the variance components, the

shape parameters ($\beta_{1,0}^{(3)}$ and $\beta_{2,0}^{(3)}$) are set to 5.0 and 2.5 to reflect high uncertainty and the scale parameters ($\beta_{1,0}^{(4)}$ and $\beta_{2,0}^{(4)}$) are set so that the expected initial variance matches a prior guess obtained by scaling the original Full-WBM variance (for fidelity, a negligible value of 10^{-4} is used to ensure numerical stability, since its value is zero).

Table 5 Prior NIG parameters

	L_i^1				L_i^2			
	$\beta_{1,0}^{(1)}$	$\beta_{1,0}^{(2)}$	$\beta_{1,0}^{(3)}$	$\beta_{1,0}^{(4)}$	$\beta_{2,0}^{(1)}$	$\beta_{2,0}^{(2)}$	$\beta_{2,0}^{(3)}$	$\beta_{2,0}^{(4)}$
Model	$(\times 10^{-4})$	$-$	$-$	$(\times 10^{-10})$	$(\times 10^{-1})$	$(\times 10^{-1})$	$-$	$(\times 10^{-4})$
GBM1	2.093	1.100	5.000	5.019	1.705	2.401	2.500	1.090
GBM2	4.096	1.100	5.000	19.212	1.705	2.401	2.500	1.090
GBM3	1.638	1.100	5.000	3.074	3.352	1.067	2.500	4.214
GBM4	3.732	1.100	5.000	15.949	1.705	2.401	2.500	1.090
GBM5	1.274	1.100	5.000	1.860	3.352	1.067	2.500	4.214
GBM6	3.277	1.100	5.000	12.296	3.352	1.067	2.500	4.214
Full-BBM	0.819	1.100	5.000	0.768	5.000	0.600	2.500	9.375

The procedure completes in ~1.9 hours, over 85% faster than the exhaustive method. Figure 5 illustrates the parameter evolutions, where the narrowing bands indicate a reduction in standard deviation across iterations, while converging to a mean value (as encouraged by the loss function in Eq. (8)). The VoI of each alternative evolves accordingly in Figure 6, displaying notorious early high values for most alternatives, which later remains high for F_1, F_4 and F_5, which correspond to the top options for Normalization 2. Table 6 shows the final ranking obtained using this method, whose order notoriously

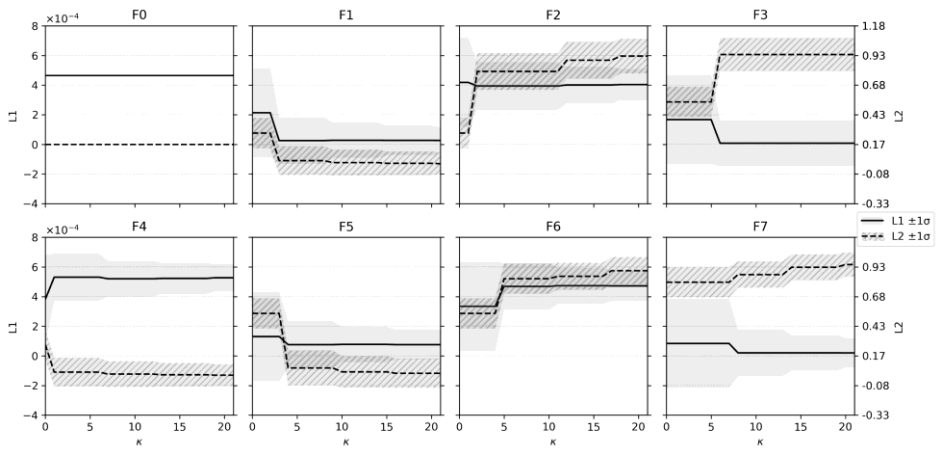

Figure 5 Evolution of the estimations of $L_1(\vec{s}_i)$ and $L_2(\vec{s}_i)$ for each plan $\vec{s}_i$

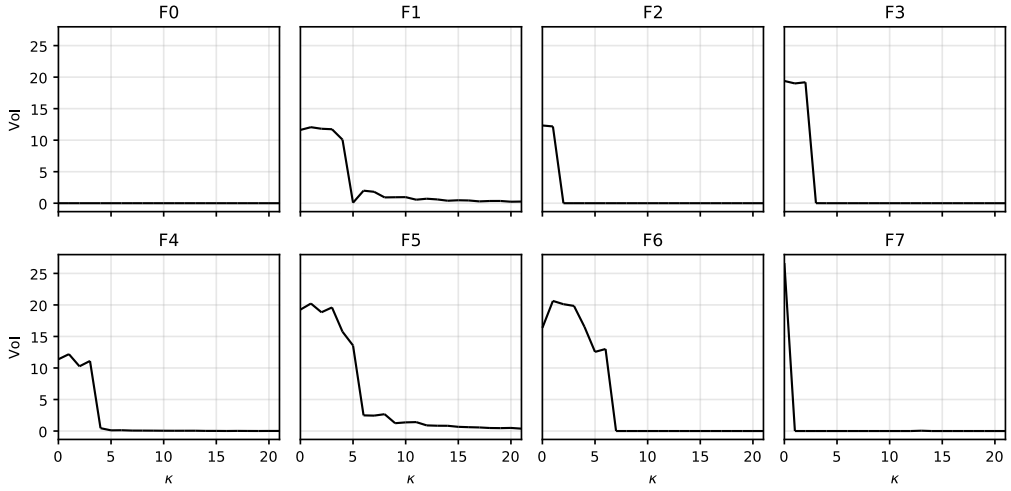

Figure 6 Evolution of VoI estimates for each model alternative

matches several of the alternatives in the exhaustive approach. Notably, F_1 is, again, identified as the best option, with a positive loss gain higher than the other alternatives.

To further analyze the performance of the VoI-driven approach, we implement a greedy heuristic that mimics the straightforward practice of iteratively replacing the most computationally burdensome WBM (among those modeling the subsystems of the IP&TLCN) with the best corresponding "off-the-shelf" BBM. At each iteration, the tradeoff gain of the model alternative is evaluated using $\mathcal{D}_{GT}$ and the Scaling 2

Table 6 Final ranking obtained using the VoI-driven method

Rank	Model name	Description	$\mathbb{E}\{L_0\} - \mathbb{E}\{L_i\}$ $(\times 10^{-1})$
1	GBM1	(0, 0, 1)	8.21
2	GBM5	(0, 1, 1)	7.82
3	Full-WBM	(0, 0, 0)	0.00
4	GBM3	(1, 0, 1)	-12.69
5	GBM4	(0, 1, 0)	-625.55
6	Full-BBM	(1, 1, 1)	-640.68
7	GBM2	(1, 0, 0)	-645.47
8	GBM6	(1, 1, 0)	-647.23

approach (see Section 4.2), until the Full-BBM configuration is reached. This greedy heuristic completes in ~2.1 hours on the same computational resources, i.e., ~10.5% slower than the VoI approach.

A comparison of rankings is provided in Table 7 that shows that, whereas the greedy heuristic method correctly identifies the two best alternatives, it overlooks GBM3, which is instead considered by the VoI-driven approach, meaning that it falls into the suboptimal solutions Full-WBM and Full-BBM (ranked 4th and 5th by the exhaustive search, respectively). This is because the greedy heuristic is solely driven by computational burden and does not account for fidelity, as it is, instead, in the VoI-driven tradeoff gain search of the optimal GBM. For example, GBM3 is overlooked by the greedy heuristic ranking because its reduction in computational burden is negligible, due to the adoption of the computational demanding BBM of the TLCN, with respect to the overall IP&TLCN fidelity gain. In contrast, the VoI-guided approach strategically explores the model repository $\mathcal{F}$, resulting in a complete ranking of alternatives (including GBM3) that, although approximate, offers a full overview of the benefits (and drawbacks) of all alternatives in the repository.

Table 7 Comparison of obtained rankings for each method

Ranking	Exhaustive	VoI-driven	Greedy Heuristic
1	GBM1	GBM1	GBM1
2	GBM5	GBM5	GBM5

3	GBM3	Full-WBM	Full-WBM
4	Full-WBM	GBM3	Full-BBM
5	Full-BBM	GBM4	-
6	GBM4	Full-BBM	-
7	GBM2	GBM2	-
8	GBM6	GBM6	-

5 CONCLUSIONS

Computational Risk Assessment (CRA) of Cyber-Physical Systems (CPSs) by Grey-Box Models (GBMs) requires to optimally combine White-Box Models (WBMs) and Black-Box Models (BBMs) of CPS sub-systems to balance the trade-off between computational burden and fidelity. Since such trade-off are unknown a priori and can only be uncovered by running simulations, we frame the GBM optimization as a Value-of-Information (VoI) decision-making problem and propose an efficient procedure of solution. VoI is iteratively computed using prior knowledge while resorting to efficient Bayesian updates from data, using conjugate priors. As a result, we avoid the need to exhaustively screen all GBM configurations across all data.

We illustrate the VoI-driven GBM optimization method by application to a case study regarding an Integrated Power and Telecommunication Network (IP&TLCN), where the performance is assessed using Energy Not Supplied (ENS) as risk metric. For validation of the procedure, we first by exhaustive search that an optimal GBM indeed exists that outperforms the original WBM; then, we show that the proposed VoI-based procedure effectively finds the optimal GBM at a fraction of the computational cost.

Future research will include developing further the proposed VoI-driven approach for i) applications to CPSs with more subsystems than the IP&TLCN, and ii) allowing the GBM selection from a repository $\mathcal{F}$ of multi-states fidelity WBMs and BBMs. A detailed study of the impact of the GBM structure (i.e., the configuration vector) on the global performance of the optimality search (e.g., through global sensitivity analysis) to characterize and refine the computational burden and fidelity estimates (e.g., dropping

normality assumptions) can prove useful for such endeavors and, also, enable accelerated identification of the optimal GBM for real-time applications.

Appendix A THE CO-SIMULATION ENVIRONMENT FOR HIERARCHICAL CPS

INFRASTRUCTURES

The CPS simulation relies on a co-simulator to coordinate the interaction across the N CPS subsystems, thus, avoiding strict dependence on individual models for each CPS subsystem. Following (Futalef et al., 2022), we achieve this task by: (i) organizing the CPS into layers, where each layer contains a subsystem that controls that in the layer immediately below; (ii) associating each subsystem with a model that implements the one-step transition function (f_j in Eq. (2)); (iii) defining the physical and cyber exchanges, corresponding to flows of bottom-up (e.g., measurements) and top-down (e.g., computed control inputs) quantities that emerge from the state vectors of each subsystem; and (iv) simulating the dynamics of the CPS using Algorithm 2, which yields a detailed simulation of the CPS response at the end of each time window (t_k, t_{k+1}), with $k \in \{0,1,2,\dots\}$.

Algorithm 2	Procedure for the recursive evaluation of the one-step state transition function $F(\cdot)$ in the CPS state-space model (Eq. (4)) considering the hierarchical approach in (Futalef et al., 2022, 2025)

Inputs: CPS state $\vec{x}(k)$

Output: updated CPS state $\vec{x}(k+1)$

1. For each j-th subsystem: sample external stimuli $\vec{e}_j$ from assigned PDFs
2. Update the binary statuses $\vec{\pi}_j$ (e.g., Markov chain)
3. Simulate the (imperfect) bottom-up transmission layer by layer
4. Use the transmitted data to compute the CPS control inputs $\vec{u}_j$ (top-down data)
5. Simulate the (imperfect) top-down transmission layer by layer
6. Use the transmitted data to evaluate: $x_j(k+1) \leftarrow f_j\left(\vec{x}_j(k), \vec{u}_j(k), \vec{e}_j(k)\right)$ (Eq. (2))
7. Return $\vec{x}(k+1) = (x_1(k+1)^\top, x_2(k+1)^\top, \dots, x_N(k+1)^\top)^\top$

Appendix B CONJUGATE PRIORS FOR THE NORMAL DISTRIBUTION WITH UNKNOWN MEAN AND VARIANCE

Let $x \sim \mathcal{N}(\mu, \sigma^2)$ with both mean μ and variance σ^2 unknown (e.g., computational burden and fidelity). Here, a convenient conjugate prior is the Normal–Inverse-Gamma (NIG) (Robert, 2007), parameterized by four parameters $\vec{\beta} = \left(\beta^{(1)}, \beta^{(2)}, \beta^{(3)}, \beta^{(4)}\right)$ such that:

$$\mu \mid \sigma^2 \sim \mathcal{N}(\beta^{(1)}, \sigma^2/\beta^{(2)}),$$

$$\sigma^2 \sim \text{Inv} - \text{Gamma}\left(\beta^{(3)}, \beta^{(4)}\right)$$

where, $b^{(1)}$ remains as the prior mean guess with $\beta^{(2)} > 0$ acting as an equivalent sampling size for μ, while $b^{(3)} > 1$ and $b^{(4)} > 0$ encode the prior belief on σ^2.

The expected values of the original normal distribution parameters μ and σ^2 can be estimated from the NIG prior parameters using the marginal normal prior for μ; these yield:

$$\left.\begin{aligned} \mu &= \beta^{(1)} \\ \sigma^2 &= \frac{\beta^{(4)}}{\beta^{(3)} - 1} \end{aligned}\right\} \qquad \texttt{(NIGtoNormal)}$$

To update the NIG parameters from evidence of n i.i.d. measurements $\{z_1, z_2, \ldots, z_n\}$, first, let:

$$\bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i$$

$$S = \sum_{i=1}^{n} (z_i - \bar{z})^2$$

The conjugate update is performed to obtain the posterior parameters $\vec{\beta}'$:

$$\left.\begin{aligned} \beta'^{(2)} &= \beta^{(2)} + n, \\ \beta'^{(1)} &= \frac{\beta^{(2)}\beta^{(1)} + n\,\bar{z}}{\beta'^{(2)}} \\ \beta'^{(3)} &= \beta^{(3)} + \frac{n}{2}, \\ \beta'^{(4)} &= \beta^{(4)} + \frac{S}{2} + \frac{\beta^{(2)} n}{2\beta'^{(2)}}\left(\bar{z} - \beta^{(1)}\right)^2 \end{aligned}\right\} \qquad \texttt{(NIGUpdate)}$$

Hence, the posterior normal parameters are recovered using $\mu', \sigma' = \texttt{NIGtoNormal}(\vec{\beta}')$.

For the expected posterior analysis in VoI, one must integrate over the NIG parameters space to obtain the marginal distribution of unobserved data $\tilde{z}$. Following (Raiffa, 1961), this results in the predictive distribution being the Student-t.

Appendix C THE WBM OF THE INTEGRATED POWER AND TELECOMMUNICATION NETWORK SYSTEM

C.1 IEEE 14 POWER GRID CASE

The IEEE14 Power Grid (PG) (Christie, 1993) (Figure 7) is composed of 14 buses, 20 transmission lines and 3 transformers. Power sources are 2 generators of active and reactive power, and 3 synchronous condensers of reactive power. Power demands are 11 loads (customers) of active and reactive power.

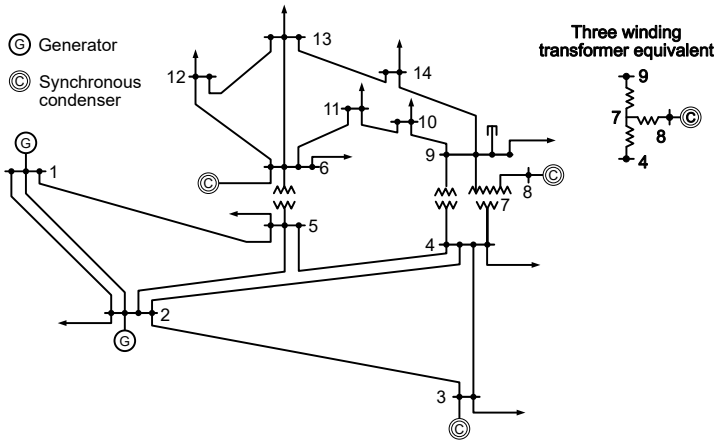

Figure 7 Electrical schematic of the IEEE14 case with its numbered buses

External stimuli. The vectors $\vec{p}_d(k)$ and $\vec{q}_d(k)$ track the active and reactive power demands of customers, respectively. Then, the external stimuli are:

$$\vec{e}_{PG}(k) = \begin{pmatrix} \vec{p}_d(k) \\ \vec{q}_d(k) \end{pmatrix} \in \mathbb{R}^{22} \tag{12}$$

The values of $\vec{p}_d(k)$ and $\vec{q}_d(k)$ are taken from (Di Maio et al., 2022), consisting of historical open data provided by (Terna, 2023), sampled every $\Delta t = 15$ [min]. For generalization purposes, we process the Terna data by applying min-max normalization and, then, multiplying these results by the nominal active and reactive power load demands in the original IEEE14 case.

Controlled inputs. The PG can be controlled by setting the voltage angles $\vec{v}_{\angle,\text{gen}}(k)$, active power generation $\vec{p}_{\text{g,gen}}(k)$ and voltage magnitudes $\vec{v}_{\text{m,gen}}(k)$ of generators, and the voltages magnitude $\vec{v}_{\text{m,sc}}(k)$ setpoints of the synchronous condensers. Then, the controlled inputs are

$$\vec{u}_{\text{PG}}(k) = \begin{pmatrix} \vec{v}_{\angle,gen}(k) \\ \vec{p}_{g,gen}(k) \\ \vec{v}_{m,gen}(k) \\ \vec{v}_{m,sc}(k) \end{pmatrix} \in \mathbb{R}^9 \tag{13}$$

State variables. The tracked continuous variables $\vec{\lambda}_{PG}(k)$ are: $\vec{p}_d(k)$ and $\vec{q}_d(k)$, i.e., the actual active and reactive power received by the customers, $\vec{p}_g(k)$ and $\vec{q}_g(k)$, i.e., the actual active and reactive power generations, $\vec{v}_m(k)$ and $\vec{v}_\angle(k)$, i.e., the voltage magnitudes and angles at each bus:

$$\vec{\lambda}_{\text{PG}}(k) = \begin{pmatrix} \vec{p}_d(k) \\ \vec{q}_d(k) \\ \vec{p}_g(k) \\ \vec{q}_g(k) \\ \vec{v}_m(k) \\ \vec{v}_\angle(k) \end{pmatrix} \in \mathbb{R}^{60} \tag{14}$$

The tracked discrete variables $\vec{\pi}_{PG}(k)$ are binary vectors $\vec{\pi}_{\text{Gen}}(k)$, $\vec{\pi}_{\text{Line}}(k)$, and $\vec{\pi}_{\text{Trafo}}(k)$, corresponding to the statues (1/0 for ON/OFF) of generators, transmission lines and transformers, respectively:

$$\vec{\pi}_{PG} = \begin{pmatrix} \vec{\pi}_{\text{Gen}}(k) \\ \vec{\pi}_{\text{Line}}(k) \\ \vec{\pi}_{\text{Trafo}}(k) \end{pmatrix} \in \{0,1\}^{28} \tag{15}$$

Generators, transmission lines and transformers can fail and be repaired (see (Futalef et al., 2025) for failure/repair rates), making $\vec{\pi}_{PG}(k)$ follow a binary Markov chain with initial condition $\vec{\pi}_{PG}(0) = \vec{1}$ (all components are up).

Following the λ-π representation of Eq. (1), the state vector of the power grid $\vec{x}_{PG}(k)$ is:

$$\vec{x}_{PG}(k) = \begin{pmatrix} \vec{\lambda}_{\mathrm{PG}}(k) \\ \vec{\pi}_{\mathrm{PG}}(k) \end{pmatrix} \in \mathbb{R}^{88} \tag{16}$$

One-step transition function. The simulation of the PG is achieved by (i) stepping the Markov chain and (ii) solving a power flow instance.

C.2 TELECOMMUNICATION NETWORK

The TLCN is an undirected weighted graph $G = (V, E, t_d)$, whose vertices V are the TLCN nodes and edges E are the TLCN links, according to Figure 8. Each edge $e \in E$ is associated a weight function $t_d: E \times [0, \infty) \mapsto [0, \infty)$ corresponding to a time-evolving time delay $t_d\left(e, t_{\mathrm{day}}(k)\right)$, with $t_{\mathrm{day}}(k) \geq 0$ the time of the day at simulation instant k. The time delays follow a reflected Wiener process with a standard deviation σ and boundaries $t^- > 0$ and $t^+ > t^-$. The TLCN enables End-to-End (E2E) communication between cyber devices (i.e., sensors/actuators in the PG and the CC), which is, however,

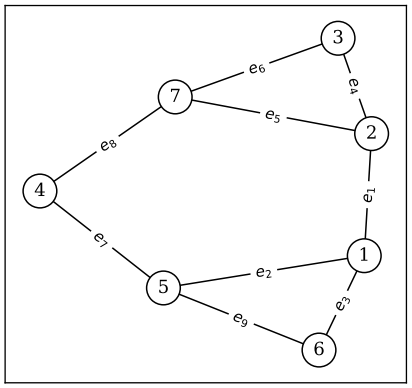

Figure 8 Topology of the TLCN with the corresponding nodes and edges labels

affected by the evolving time delays and the structure of the TLCN network itself (see (Futalef et al., 2024) for specific details).

External stimuli. Without loss of generality, the TLCN is not affected by external stimuli; thus:

$$\vec{e}_{\mathrm{TLCN}}(k) = \emptyset \tag{17}$$

Controlled inputs. The bottom-up packet $\vec{u}^{\uparrow}_{\mathrm{TLCN}}(k)$ are the PG external stimuli $\vec{e}_{PG}(k)$ and the binary statuses $\vec{\pi}_{\mathrm{PG}}(k)$ in the PG:

$$\vec{u}^{\uparrow}_{\mathrm{TLCN}}(k) = \begin{pmatrix} \vec{e}_{PG}(k) \\ \vec{\pi}_{PG}(k) \end{pmatrix} \tag{18}$$

The top-down packet $\vec{u}^{\downarrow}_{\mathrm{TLCN}}(k)$ are the PG control settings (e.g., computed by the control center in the next Appendix C.3):

$$\vec{u}^{\downarrow}_{\mathrm{TLCN}}(k) = \begin{pmatrix} \vec{v}_{\angle,gen}(k) \\ \vec{p}_{g,gen}(k) \\ \vec{v}_{m,gen}(k) \\ \vec{v}_{m,sc}(k) \end{pmatrix} \tag{19}$$

The controlled inputs vector $\vec{u}_{TLCN}(k)$ is the stack of both packets:

$$\vec{u}_{TLCN}(k) = \begin{pmatrix} \vec{u}^{\uparrow}_{\mathrm{TLCN}}(k) \\ \vec{u}^{\downarrow}_{\mathrm{TLCN}}(k) \end{pmatrix} \tag{20}$$

State variables. We track the following variables: the time delays across TLCN links $\vec{t}_d\ (t_k)$, the binary statuses of TLC nodes $\vec{\pi}_{\mathrm{node}}(t_k)$ and links $\vec{\pi}_{\mathrm{link}}(t_k)$; by denoting the bottom-up(top-down) flow variables with superscript ↑(↓), then the state variables are:

$$\vec{x}_{\text{TLCN}}(k) = \begin{pmatrix} \vec{x}^{\uparrow}_{\text{TLCN}}(k) \\ \vec{x}^{\downarrow}_{\text{TLCN}}(k) \end{pmatrix} = \begin{pmatrix} \vec{t}^{\uparrow}_{d}(t_k) \\ \vec{\pi}^{\uparrow}_{\text{node}}(k) \\ \vec{\pi}^{\uparrow}_{\text{link}}(k) \\ \vec{t}^{\downarrow}_{d}(t_k) \\ \vec{\pi}^{\downarrow}_{\text{node}}(k) \\ \vec{\pi}^{\downarrow}_{\text{link}}(k) \end{pmatrix} \tag{21}$$

At any instant k, any node in V and edge in E might become due to sudden disconnections of nodes/edges or underlying TLCN protocol (see (Futalef et al., 2024) for specific values).

One-step transition function. The simulation of the TLCN is achieved by (i) simulating the stochastic disconnection of TLCN nodes and links, (ii) stepping the reflected Weiner process and (iii) emulating the E2E communication between cyber devices.

C.3 CONTROL CENTER

The Control Center (CC) is a cyber device that computes optimal control inputs for the PG by solving an Alternating Current Optimal Power Flow (AC-OPF) instance according to the received measurements.

External stimuli. Without loss of generality, the CC is not affected by external stimuli; thus:

$$\vec{e}_{\text{CC}}(k) = \emptyset \tag{22}$$

Controlled inputs. Following Appendix C.1, these inputs correspond to the transmitted packet $\vec{u}^{\uparrow}_{\text{TLCN}}(k)$, i.e., the PG measurements, denoted $\Psi\left(\vec{u}^{\uparrow}_{\text{TLCN}}(k), \vec{x}_{TLCN}(k)\right)$, where $\Psi(\cdot)$ is a transmission function that emulates the E2E communication (see (Futalef et al., 2024)) according to the TLCN state vector $\vec{x}_{TLCN}(k)$:

$$\vec{u}_{CC}(k) = \Psi\left(\vec{u}^{\uparrow}_{\text{TLCN}}(k), \vec{x}_{TLCN}(k)\right) \tag{23}$$

State variables. The state vector is like that of the PG, although these are obtained from the AC-OPF simulation (and not the power flow simulation) according to the received measurements, denoted with the subscript "OPF":

$$\vec{x}_{CC}(k) = \begin{pmatrix} \vec{\lambda}_{\mathrm{PG,OPF}}(k) \\ \vec{\pi}_{\mathrm{PG,OPF}}(k) \end{pmatrix}$$

One-step transition function. The simulation of the CC is achieved by creating and solving an instance of the AC-OPF using the obtained measurements; if the AC-OPF does not converge, the CC state is set to zero, i.e., request all generators to shut down.

C.4 CPS DYNAMICS

The CPS process variables are the concatenation of the subsystem process variables (Eqs. (12) to (24)):

$$\begin{aligned} \vec{e}(k) &= \left(\vec{e}_{PG}^{\,\mathsf{T}}(k), \vec{e}_{TLCN}^{\,\mathsf{T}}(k), \vec{e}_{CC}^{\,\mathsf{T}}(k)\right)^{\mathsf{T}} \\ \vec{u}(k) &= \left(\vec{u}_{PG}^{\,\mathsf{T}}(k), \vec{u}_{TLCN}^{\,\mathsf{T}}(k), \vec{u}_{CC}^{\,\mathsf{T}}(k)\right)^{\mathsf{T}} \\ \vec{x}(k) &= \left(\vec{x}_{PG}^{\,\mathsf{T}}(k), \vec{x}_{TLCN}^{\,\mathsf{T}}(k), \vec{x}_{CC}^{\,\mathsf{T}}(k)\right)^{\mathsf{T}} \end{aligned} \tag{24}$$

REFERENCES

Alawad, F., & Kraemer, F. A. (2022). Value of Information in Wireless Sensor Network Applications and the IoT: A Review. *IEEE Sensors Journal*, *22*(10), 9228–9245. https://doi.org/10.1109/JSEN.2022.3165946

Beck, J. L., & Zuev, K. M. (2017). Rare-event simulation. In *Handbook of Uncertainty Quantification* (pp. 1075–1100). https://doi.org/10.1007/978-3-319-12385-1_24

Cassottana, B., Roomi, M. M., Mashima, D., & Sansavini, G. (2023). Resilience analysis of cyber-physical systems: A review of models and methods. *Risk Analysis*, *43*(11), 2359–2379. https://doi.org/10.1111/risa.14089

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16*, 785–794. https://doi.org/10.1145/2939672.2939785

Christie, R. D. (1993). *14 Bus Power Flow Test Case*. http://labs.ece.uw.edu/pstca/pf14/pg_tca14bus.htm

Colabianchi, S., Costantino, F., Di Gravio, G., Nonino, F., & Patriarca, R. (2021). Discussing resilience in the context of cyber physical systems. *Computers & Industrial Engineering*, *160*, 107534. https://doi.org/10.1016/j.cie.2021.107534

de Paula Ferreira, W., Armellini, F., & De Santa-Eulalia, L. A. (2020). Simulation in industry 4.0: A state-of-the-art review. *Computers and Industrial Engineering*, *149*. https://doi.org/10.1016/j.cie.2020.106868

Deb, C., Zhang, F., Yang, J., Lee, S. E., & Shah, K. W. (2017). A review on time series forecasting techniques for building energy consumption. *Renewable and Sustainable Energy Reviews*, *74*(7373), 902–924. https://doi.org/10.1016/j.rser.2017.02.085

Di Maio, F., Stincardini, A., & Zio, E. (2022). Identification of Vulnerabilities in Integrated Power-Telecommunication Infrastructures; A Simulation-based Approach. *Proceedings of the 32nd European Safety and Reliability Conference*, 2749–2756. https://doi.org/10.3850/978-981-18-5183-4_S20-04-425-cd

Di Maio, F., & Zio, E. (2018). Dynamic Accident Scenario Generation, Modeling and Post-Processing for the Integrated Deterministic and Probabilistic Safety Analysis of Nuclear Power Plants. In *Advanced Concepts in Nuclear Energy Risk Assessment and Management: Volume 1* (World Scientific, pp. 477–504). https://doi.org/10.1142/9789813225619_0013

Fenwick, E., Steuten, L., Knies, S., Ghabri, S., Basu, A., Murray, J. F., Koffijberg, H. (Erik), Strong, M., Sanders Schmidler, G. D., & Rothery, C. (2020). Value of Information Analysis for Research Decisions—An Introduction: Report 1 of the ISPOR Value of Information Analysis Emerging Good Practices Task Force. *Value in Health*, *23*(2), 139–150. https://doi.org/https://doi.org/10.1016/j.jval.2020.01.001

Fioretto, F., Mak, T. W. K., & van Hentenryck, P. (2020). Predicting AC optimal power flows: Combining deep learning and lagrangian dual methods. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, 630–637. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85102868479&partnerID=40&md5=52e8d059402fad96c85c304215aaec2b

Futalef, J.-P., Di Maio, F., & Zio, E. (2022). Grey-Box Models for Cyber-Physical Systems Reliability, Safety and Resilience Assessment. *Proceedings of the 32nd European Safety and Reliability Conference*, 2757–2764. https://doi.org/10.3850/978-981-18-5183-4_S20-05-564-cd

Futalef, J.-P., Di Maio, F., & Zio, E. (2024). A Classification Problem Formulation for the Reliability Assessment of End-to-End Communication Among Cyber Devices of Cyber-Physical Systems. *2024 8th International Conference on System Reliability and Safety (ICSRS)*, 629–633. https://doi.org/10.1109/ICSRS63046.2024.10927501

Futalef, J.-P., Di Maio, F., & Zio, E. (2025). A dynamic importance function for accidental scenarios generation by RESTART in the computational risk assessment of cyber-physical infrastructures. *Reliability Engineering & System Safety*, *253*, 110538. https://doi.org/10.1016/j.ress.2024.110538

Gibbons, J. D., & Chakraborti, S. (2020). Tests of Goodness of Fit. In *Nonparametric Statistical Inference* (6th Edition, pp. 107–166). https://doi.org/10.1201/9781315110479

Giselle Fernández-Godino, M., Park, C., Kim, N. H., & Haftka, R. T. (2019). Issues in Deciding Whether to Use Multifidelity Surrogates. *AIAA Journal*, *57*(5), 2039–2054. https://doi.org/10.2514/1.J057750

Guo, L., Yang, B., Ye, J., Chen, H., Li, F., Song, W., Du, L., & Guan, L. (2021). Systematic Assessment of Cyber-Physical Security of Energy Management System for Connected and Automated Electric Vehicles. *IEEE Transactions on Industrial Informatics*, *17*(5), 3335–3347. https://doi.org/10.1109/TII.2020.3011821

Hansen, S. T., Thule, C., Gomes, C., Lausdahl, K. G., Madsen, F. P., Abbiati, G., & Larsen, P. G. (2024). Co-simulation at different levels of expertise with Maestro2. *Journal of Systems and Software*, *209*, 111905. https://doi.org/10.1016/J.JSS.2023.111905

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer New York. https://doi.org/10.1007/978-0-387-84858-7

Hoseyni, S. M., Di Maio, F., & Zio, E. (2019). VoI-Based Optimal Sensors Positioning and the Sub-Modularity Issue. *2019 4th International Conference on System Reliability and Safety (ICSRS)*, 148–152. https://doi.org/10.1109/ICSRS48664.2019.8987650

Hoseyni, S. M., Di Maio, F., & Zio, E. (2021). Optimal sensor positioning on pressurized equipment based on Value of Information. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, *235*(4), 533–544. https://doi.org/10.1177/1748006X21989661

Howard, R. (1966). Information Value Theory. *IEEE Transactions on Systems Science and Cybernetics*, *2*(1), 22–26. https://doi.org/10.1109/TSSC.1966.300074

Jarwan, A., Sabbah, A., & Ibnkahla, M. (2022). Information-Oriented Traffic Management for Energy-Efficient and Loss-Resilient IoT Systems. *IEEE Internet of Things Journal*, *9*(10), 7388–7403. https://doi.org/10.1109/JIOT.2021.3132925

Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., & Zhao, W. (2017). A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications. *IEEE Internet of Things Journal*, *4*(5), 1125–1142. https://doi.org/10.1109/JIOT.2017.2683200

Liu, Z., Meng, X., Liu, Y., Yang, Y., & Wang, Y. (2022). AUV-Aided Hybrid Data Collection Scheme Based on Value of Information for Internet of Underwater Things. *IEEE Internet of Things Journal*, *9*(9), 6944–6955. https://doi.org/10.1109/JIOT.2021.3115800

López-Ibáñez, M., Dubois-Lacoste, J., Pérez Cáceres, L., Birattari, M., & Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, *3*, 43–58. https://doi.org/10.1016/j.orp.2016.09.002

Lv, Z., Han, Y., Singh, A. K., Manogaran, G., & Lv, H. (2021). Trustworthiness in Industrial IoT Systems Based on Artificial Intelligence. *IEEE Transactions on Industrial Informatics*, *17*(2), 1496–1504. https://doi.org/10.1109/TII.2020.2994747

Malings, C., & Pozzi, M. (2018). Value-of-information in spatio-temporal systems: Sensor placement and scheduling. *Reliability Engineering & System Safety*, *172*, 45–57. https://doi.org/10.1016/j.ress.2017.11.019

Malings, C., & Pozzi, M. (2019). Submodularity issues in value-of-information-based sensor placement. *Reliability Engineering & System Safety*, *183*, 93–103. https://doi.org/10.1016/J.RESS.2018.11.010

Mao, S., Chen, B., Malki, M., Chen, F., Morales, M., Ma, Z., & Mehana, M. (2024). Efficient prediction of hydrogen storage performance in depleted gas reservoirs using machine learning. *Applied Energy*, *361*, 122914. https://doi.org/10.1016/j.apenergy.2024.122914

MathWorks. (2025). *MATLAB*. https://www.mathworks.com/products/matlab.html

Memarzadeh, M., & Pozzi, M. (2016). Value of information in sequential decision making: Component inspection, permanent monitoring and system-level scheduling. *Reliability Engineering & System Safety*, *154*, 137–151. https://doi.org/10.1016/j.ress.2016.05.014

Modelica Association. (2025). *Modelica*. https://modelica.org/

Nabavinejad, S. M., Reda, S., & Ebrahimi, M. (2022). Coordinated Batching and DVFS for DNN Inference on GPU Accelerators. *IEEE Transactions on Parallel and Distributed Systems*, *33*(10), 2496–2508. https://doi.org/10.1109/TPDS.2022.3144614

Pinto, G., Deltetto, D., & Capozzoli, A. (2021). Data-driven district energy management with surrogate models and deep reinforcement learning. *Applied Energy*, *304*, 117642. https://doi.org/10.1016/j.apenergy.2021.117642

Robert, C. P. (2007). From Prior Information to Prior Distributions. In C. P. Robert (Ed.), *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation* (pp. 105–163). Springer New York. https://doi.org/10.1007/0-387-71599-1_3

Terna. (2023). *Transparency Report: The Dashboard*. https://www.terna.it/en/electric-system/transparency-report

Thelen, A., Zhang, X., Fink, O., Lu, Y., Ghosh, S., Youn, B. D., Todd, M. D., Mahadevan, S., Hu, C., & Hu, Z. (2022). A comprehensive review of digital twin — part 1: modeling and twinning enabling technologies. *Structural and Multidisciplinary Optimization*, *65*(12). https://doi.org/10.1007/s00158-022-03425-4

Tong, X., Huang, Y., Chang, B., & Chen, Z. (2024). A Novel Information Update Policy for Real-Time Wireless Feedback Control in IIoT. *IEEE Internet of Things Journal*, *11*(8), 13152–13166. https://doi.org/10.1109/JIOT.2023.3332042

Turati, P., Pedroni, N., & Zio, E. (2016). Advanced RESTART method for the estimation of the probability of failure of highly reliable hybrid dynamic systems. *Reliability Engineering & System Safety*, *154*, 117–126. https://doi.org/10.1016/j.ress.2016.04.020

Zaparoli Cunha, B., Droz, C., Zine, A.-M., Foulard, S., & Ichchou, M. (2023). A review of machine learning methods applied to structural dynamics and vibroacoustic. *Mechanical Systems and Signal Processing*, *200*, 110535. https://doi.org/10.1016/j.ymssp.2023.110535

Zeigler, B. P., Muzy, A., & Kofman, E. (2019). *Theory of Modeling and Simulation* (3rd ed.). Elsevier. https://doi.org/10.1016/C2016-0-03987-6

Zio, E. (2018). The future of risk assessment. *Reliability Engineering & System Safety*, *177*, 176–190. https://doi.org/10.1016/j.ress.2018.04.020

Zografopoulos, I., Ospina, J., Liu, X., & Konstantinou, C. (2021). Cyber-Physical Energy Systems Security: Threat Modeling, Risk Assessment, Resources, Metrics, and Case Studies. *IEEE Access*, *9*, 29775–29818. https://doi.org/10.1109/ACCESS.2021.3058403

List of Acronyms

BBM	Black-Box Model	KS	Kolmogorov-Smirnov
CC	Control Center	LoF	Lack-of-Fit
CPS	Cyber-Physical System	NIG	Normal-Inverse-Gamma
CRA	Computational Risk Assessment	NN	Neural Network
ENS	Energy Not Supplied	PG	Power Grid
GBC	Gradient Boosting Classifier	SMD	Smart Monitoring Device
GBM	Grey-Box Model	TLCN	Telecommunication Network
GT	Ground-Truth	VoI	Value-of-Information
IP&TLCN	Integrated Power and Telecommunication Network	WBM	White-Box Model

List of Symbols

N	Num. of CPS subsystems	k	Discrete simulation instant
t_k	Continuous simulation time at instant k	Δt	Simulation time step
$\vec{x}_j(k)$	State vector of j-th CPS subsystem at instant k	$\vec{\lambda}_j(k)$	Continuous process variables of j-th CPS subsystem
$\vec{\pi}_j(k)$	Discrete process variables of j-th CPS subsystem	$\vec{u}_j(k)$	Controlled inputs of j-th CPS subsystem
$\vec{e}_j(k)$	Uncontrolled inputs of j-th CPS subsystem	$f_j(\cdot)$	State transition function of j-th CPS subsystem
$\vec{x}(k)$	CPS state vector	$\vec{u}(k)$	CPS control inputs
$\vec{e}(k)$	CPS uncontrolled inputs	$F(\cdot)$	CPS state transition function
y_k	CPS process variables at instant k	n^{miss}	Simulation mission instants
Y_l	l-th scenario	$\mathcal{F}$	Model repository
F_i	i-th model alternative in $\mathcal{F}$	$\vec{s}_i$	Plan vector of model alternative F_i
L_i^1	Computational load of model alternative F_i	L_i	Aggregated loss of model alternative F_i
$\mathcal{D}_{GT}$	Ground-truth dataset	L_i^2	Fidelity of model alternative F_i
$\emptyset$	Used for reference values (e.g., model, losses)	n^{GT}	Number of scenarios in $\mathcal{D}_{GT}$
$\vec{\theta}_i^{L_2}$	Normal parameters for fidelity	$\vec{\theta}_i^{L_1}$	Normal parameters for computational load
$\vec{\beta}_{i,\kappa}^{L_2}$	NIG parameters for fidelity	$\vec{\beta}_{i,\kappa}^{L_1}$	NIG parameters for computational load
$L(\cdot)$	Aggregated loss function	F^*	Optimal GBM alternative
n^{batch}	Batch size	$r(\cdot)$	Risk metric
$p_{\%}^{acc}$	Accidental scenario percentage in each batch	n^{out}	Iteration skips

$\lambda_1(\cdot), \lambda_2(\cdot)$	Scaling functions for computational load and fidelity	n^B	Number of pre-batches
$z_i^{L_1}$	Computational load evidence collected using F_i	ω_1, ω_2	Weights for computational load and fidelity
t_k^{CPU}	Cumulative execution time until instant k	$z_i^{L_2}$	Fidelity evidence collected using F_i
$\hat{\epsilon}_i$	LoF obtained from simulating a batch using F_i	$\widehat{m}_{i,l}$	Least-squares sloped obtained from simulation scenario Y_l using F_i
$r(\cdot)$	Risk metric	$\mathcal{r}_{l,i}$	Aggregated risk metric for scenario Y_l using F_i
$\hat{P}_i^{risk}$	Empirical cumulative distribution of a batch using F_i	$\delta(P_1, P_2)$	KS statistic between cumulative distributions P_1 and P_2