A DYNAMIC IMPORTANCE FUNCTION FOR ACCIDENTAL SCENARIOS GENERATION BY RESTART IN THE COMPUTATIONAL RISK ASSESSMENT OF CYBER-PHYSICAL INFRASTRUCTURES

Juan-Pablo Futalef[1], Francesco Di Maio[1,*], Enrico Zio[1,2]
[1]Energy Department, Politecnico Di Milano, Italy
[2]Centre de Recherche sur les Risques et les Crises, Mines Paris - PSL, France

Abstract – The Computational Risk Assessment (CRA) of Cyber-Physical Systems (CPSs) calls for the analysis of accidental scenarios emerging from the complexities and interdependencies typical of CPSs. Generating these scenarios via crude Monte Carlo Simulation (MCS) is impractical due to the high computational demand of simulation codes of CPSs, considering the combinatorial number of possible scenarios. In this paper, we tailor the use of Repetitive Simulation Trials After Reaching Thresholds (RESTART), a rare-event simulation method of literature, to efficiently generate relevant accidental scenarios. The tailored RESTART is guided by a dynamic Importance Function (IF) originally introduced here to dynamically characterize the relevance of the scenarios with reference to the current topology of the CPS and the susceptibility of its components. Two case studies of increasing complexity are considered: a single power grid and a CPS consisting of an Integrated Power and Telecommunication (IP&TLC) infrastructure. Results show that RESTART mines out more relevant scenarios than crude MCS for a number of different IFs based on vulnerability metrics of literature, and thus particularly efficiently when the novel IF is adopted.

***Keywords*:** Cyber-physical system, Risk assessment, Rare event simulation, RESTART, Black-box modelling, Grey-box modelling, Integrated Power and Telecommunication infrastructure

1 INTRODUCTION

Cyber-Physical Systems (CPSs) integrate cyber and physical networks of components to enhance the performance and robustness of lifelines, such as energy, communication, transportation, and water supply (Cassottana et al. 2023; Zio 2018). However, the integration of the cyber and physical elements creates complex interdependencies that can shadow hazards and threats: this renders risk assessment for these systems a challenge (Cassottana et al., 2023; Kröger & Zio, 2011).

With the continuous rise of computational power, Computational Risk Assessment (CRA) has established itself as an approach to generate and analyze multiple scenarios of response of complex systems and infrastructures (Di Maio & Zio, 2018; Sezen et al., 2019). With regards to CPS risk assessment, CRA can be embraced to generate and analyze multiple CPS responses to disruptive events of cyber-attacks and/or physical failures. The set of scenarios generated is to be post-processed to identify causes and criticalities for the loss of CPS functionality that would be difficult to discover by expert judgment or by traditional risk assessment methods, given the complexity of the CPS and the interdependent dynamics therein (Di Maio & Zio, 2018).

Given the random character of disruptions to CPSs, Monte Carlo Simulation (MCS) (Zio, 2013) stands as the straightforward way to generate scenarios. However, the large number of scenarios required to comprehensively analyze the CPS response under the combinatorial number of disrupted elements in the system makes MCS impractical. This is even more so when the simulation of the response of the CPS to the combinations of disrupted elements is done with a high-fidelity White-Box Model (WBM), which is computationally burdensome, and those combinations which make the CPS fail have a small probability of occurrence, i.e. it is a rare-event problem (Au & Patelli, 2016; Di Maio & Zio,

[*]Corresponding author Francesco Di Maio (francesco.dimaio@polimi.it). Address: Via La Masa 32, 20156, Milan, Italy.

Table 1 Summary of rare events simulation methods

Family	Method	Curse of dimensionality	Dynamic processes	Multimodal PDFs	Convergence	Reference
Crude Monte Carlo Simulation (MCS)	Crude MCS	●●●	Yes	●●●	●○○	(Zio, 2013)
Importance Sampling (IS)	Classical IS	●○○	Yes	●●○	●●○	(Kroese et al., 2011)
	State-dependent IS	●○○	Yes	●●○	●●●	(Rubino & Tuffin, 2009)
	Cross Entropy (CE) Method	●○○	Yes	●●○	●●●	(Boer et al., 2005)
Splitting	Multilevel splitting	●●○	Yes	●●●	●●●	(Glasserman et al., 1999)
	Subset Simulation (SS)	●●○	Yes	●●●	●●●	(Au & Beck, 2001; Pedroni & Zio, 2017)
	Repetitive Simulation Trials After Reaching Thresholds (RESTART)	●●○	Yes	●●●	●●●	(Turati et al., 2016; M. Villén-Altamirano & Villén-Altamirano, 2011)
Line Sampling (LS)	Traditional	●●○	No	●●○	●●○	(Dang et al., 2023; Zio & Pedroni, 2010)
	Heuristic	●●○	No	●●○	●●●	(Dang et al., 2023)

Legend: ineffective (•○○), sufficient (••○), or effective (•••)

2018). With respect to the former issue of computationally demanding simulations, it is known that data-driven Black-Box Models (BBMs) of Artificial Intelligence (AI) can be trained to give the system responses with a good level of accuracy (if enough data are available for training) and do so in a lower computational time than WBMs; one disadvantage, however is that BBMs lack interpretation of their output with respect to the input and the physics of the problem (Peherstorfer et al., 2018). To alleviate this last issue, Grey-Box Models (GBMs) have been introduced to effectively merge WBMs and BBMs for exploiting the benefits of both, i.e. to reduce the computational burden and retain interpretability, at least to a certain extent (Futalef et al., 2022; Giselle Fernández-Godino et al., 2019; Kapteyn et al., 2022).

For the issue of rare event simulation, common to several applications of physics and engineering (Au & Patelli, 2016; Cadini et al., 2017; Turati et al., 2016), methods have been developed to efficiently sample small probability events that trigger scenarios of interest (Beck & Zuev, 2017; Kroese et al., 2011). A non-exhaustive list of rare event simulation methods is provided in Table 1; the methods listed are qualitatively compared with respect to their ability to handle:

- *Curse of dimensionality*;
- *Dynamic processes*;
- *Multimodal Probability Density Functions (PDFs)*
- *Probability estimation convergence*.

Line Sampling (LS) methods (both traditional and heuristic) are designed for dealing with steady-state processes (mainly in the structural reliability domain), which makes it difficult to use for CPS applications. As for Importance Sampling (IS) (i.e., classical, state-dependent and Cross-Entropy (CE)) methods, their performance drops with the curse of dimensionality. Splitting methods, such as multi-level splitting, Subset Simulation (SS) coupled with Metropolis-Hastings sampling or IS, and the Repetitive Simulation Trials After Reaching Thresholds (RESTART), stand out for the purposes of simulating CPS responses to disturbed states, due to their capability of dealing with the curse of

dimensionality, dynamic processes and multimodal PDFs, while converging relatively fast to the rare event probability estimate. Among these, RESTART has been chosen because already successfully applied for the study of interdependent and hybrid dynamic systems, for example in (Janghoon Kim et al., 2013; J. Villén-Altamirano, 2014; Turati et al., 2016), that very much resonate with the peculiarities of CPSs, and because does not require for the identification of minimal cut sets to be used, that would result impractical when dealing with CPSs (Di Maio et al., 2023). Still, RESTART needs a specific definition of the splitting process for the CPS response simulation.

The goal of this work is to tailor RESTART for efficiently guiding the exploration of accidental scenarios in a CPS, initiated by disrupted states of some of its elements. We achieve this by resorting to a novel dynamic IF that quantifies the CPS vulnerability during each simulated scenario by accounting simultaneously for (a) the centrality of CPS components exposed to the hazards/threats that can cause their disruption/failure and (b) their time-dependent susceptibility to disruption/failure. RESTART is wrapped around a general stochastic discrete-time state-space model of the CPS to track its underlying process variables and the disrupted states of its components. The introduced IF will be shown to allow RESTART to effectively allocate computational time for simulating scenarios contributing to reaching the CPS failed state, i.e., by the use of a metric of exploration efficiency. A comparison to traditional MCS and to RESTART guided by other IFs based on vulnerability metrics of literature shows that the method proposed in this paper indeed yields better exploration efficiency.

Centrality metrics are built into the IF for guiding RESTART because they provide a quantitative indication of the importance of the CPS components with reference to the CPS network topology (Devineni et al., 2020; Eusgeld et al., 2009; Piccinelli et al., 2017; Zio et al., 2012). For the purposes of the present work, the centrality metric used is the Current-Flow Betweenness Centrality (CF-BC), which ranks the components with respect to their topological relevance in fulfilling the system function (Newman, 2005). The original IF defined in this work also accounts for the susceptibility to failure of the components of the CPS by considering the dynamically evolving stress on the components throughout the scenario evolution to measure to what extent the component is endangered to failure while exposed to a hazard/threat. By coupling centrality and susceptibility in the IF of RESTART, the algorithm can be guided to generate relevant combinations of failures of central and susceptible components.

The performance of the method developed is benchmarked on two case studies of increasing system complexity: the IEE14 power grid (Christie, 1993) and an Integrated-Power and Telecommunication (IP&TLC) that couples the IEEE14 power grid with a Telecommunication Network (TLCN) (Di Maio et al., 2022). The results of both cases show that the proposed methods outperform others of literature in terms of effectiveness of relevant scenarios exploration and computational savings.

The remainder of the paper is organized as follows. Section 2 recalls the RESTART method, introduces the tailoring of RESTART to generate relevant scenarios for CPSs, and defines the novel IF. Section 3 presents the case studies and discusses the main findings. Finally, Section 4 concludes the work and suggests future research.

2 SCENARIO GENERATION BY RESTART

CPSs are designed and manufactured to be reliable, hence their failures are rare events. To efficiently perform the CRA of CPSs, scenarios that lead them to failure must be efficiently simulated. The simulation can be, in general terms, based on stochastic process modeling, e.g., discrete-time state-space models.

2.1 CPS STATE-SPACE MODEL

In practice, to each instant k corresponds a time $t_k = \Delta t \cdot k + t_0$, with Δt being a fixed time step and t_0 the initial time. At any discrete instant $k = \{0,1,2,\dots\}$, the CPS state vector is $\mathbf{X}(k) \in \Omega \subset \mathbb{R}^{n_x}$, $1 \leq n_x$, with initial condition $\mathbf{X}(0) = \mathbf{X}_0$. This vector contains the dynamic process variables that characterize the state of all the N interacting components in the CPS. In general terms, $\mathbf{X}(k)$ might comprise continuous and discrete variables: in our case, continuous variables are physical quantities represented by the vector $\mathbf{\Lambda}(k)$ and discrete variables are binary component states, i.e., 1 ("up") or 0 ("down"), represented by the vector $\mathbf{\Pi}(k)$, so that:

$$\mathbf{X}(k) = \begin{pmatrix} \mathbf{\Lambda}(k) \\ \mathbf{\Pi}(k) \end{pmatrix} \tag{1}$$

For any component $j \in \{1,2,\dots,N\}$, its multidimensional state vector $\mathbf{X}_j(k) \in \mathbb{R}^{n_{x,j}}$, $1 \leq n_{x,j} \leq n_x$, is the portion of $\mathbf{X}(k)$ that contains the dynamic process variables of that component.

The CPS state variables evolve in time following the one-step transition process in Eq. (2), where $\boldsymbol{F}(\cdot)$ is a nonlinear state-transition function that considers the N components interdependencies, $\mathbf{U}(k) \in \mathbb{R}^{n_u}$ represents the controlled variables (e.g., power generators settings in the power grid, data packets in the telecommunication network), whereas $\mathbf{E}(k) \in \mathbb{R}^{n_e}$ are the uncontrolled external stimuli (e.g., power demands); for each j-th component, we denote its controlled and uncontrolled variables as $\mathbf{U}_j(k) \in \mathbb{R}^{n_{u,j}}$ and $\mathbf{E}_j(k) \in \mathbb{R}^{n_{e,j}}$, respectively, where $n_{u,j}, n_{e,j} \geq 0$:

$$\mathbf{X}(k+1) = \mathbf{F}\big(\mathbf{X}(k), \mathbf{U}(k), \mathbf{E}(k)\big) \tag{2}$$

The simulation of the CPS uses a co-simulation approach to account for the interactions among the N components, while avoiding dependence on the underlying models of those components. Each component is characterized by its own independent model that operates in its corresponding state-space domain. To enable the co-simulation of these diverse models, we implement the hierarchical approach described in (Futalef et al., 2022), which provides a master algorithm coordinating the co-simulation process and integrating the different models. This approach requires proceeding as follows (1) organizing the CPS into layers: each layer contains the components that control those in the layer immediately below it; (2) defining physical and cyber bottom-up and top-down flows (e.g., bottom layers measurements flow up until they reach a central controller (bottom-up), where control inputs are computed and, then, sent to the bottom layer components (top-down)); (3) simulating at each time k the dynamical response of the CPS using Algorithm 1. This procedure yields a detailed simulation of the CPS response at the end of each time window (t_k, t_{k+1}), $k \in \{0,1,2,\dots\}$.

Algorithm 1	Procedure for the recursive evaluation of the one-step state transition function $\mathbf{F}(\cdot)$ in the CPS state-space model (Eq. (2)) considering the hierarchical approach in (Futalef et al., 2022)

Inputs: last state $\mathbf{X}(k)$ and CPS hierarchical model

Output: updated next state $\mathbf{X}(k+1)$

1 Sample the external stimuli values E(k) from assigned PDFs
2 Update the status of components Π(k)
3 Collect measurements from and states of components (bottom-up data)
4 Simulate the (imperfect) bottom-transmission
5 Use the transmitted data to compute the CPS control inputs U(k) (top-down data)
6 Simulate the (imperfect) top-down transmission
7 Use the transmitted data to evaluate $\mathbf{F}\big(\mathbf{X}(k), \mathbf{U}(k), \mathbf{E}(k)\big)$ (Eq. (2))
8 Return X(k+1)

2.2 RESTART FOR CPS ACCIDENTAL SCENARIO GENERATION

For the general CPS discrete-time state-space model introduced in the previous Section, we tailor RESTART to generate the relevant scenarios of interest. This is achieved by integrating the co-simulation framework with RESTART via the state-space model (as illustrated in Figure 1), allowing us to identify the most relevant sequences of $\mathbf{X}(k)$ that can lead the CPS to failure. Considering this, RESTART proceeds as follows:

1) Define $\{\phi(\mathbf{X}(k))\}$ that is a scalar map $\phi: \Omega \rightarrow \mathbb{R}$ called *Importance Function* (IF), whose range is divided into $m+1$ regions defined by several thresholds $T_i \in \mathbb{R}$, $i \in \{1, \ldots, m\}$, such that $T_{i'} > T_i$, if and only if $i' > i$. Such thresholds define m nested regions of increasing relevance $C_i := \{\mathbf{X} \in \Omega | \phi(\mathbf{X}(k)) \geq T_i\}$; by denoting $C_0 = \Omega$, then, $C_0 \supset C_1 \ldots \supset C_m$, and the intermediate regions $\Delta C_i := C_i - C_{i+1}$ for $i \in \{0, \ldots, m-1\}$, and $\Delta C_m = C_m$.
2) Run a main trial, which is a crude MCS of $\mathbf{X}(k)$, track $\phi(\mathbf{X}(k))$ and record the instant(s) it up-crosses the threshold T_1; label the up-crossing event(s) as U_1 and the occurrence instants(s) as k_{U_1}, and save the corresponding state(s) $\mathbf{X}(k_{U_1})$.
3) For each U_1, perform $R_1 - 1$, $R_1 \in \mathbb{N}$, simulations called *retrials* starting from k_{U_1} with initial condition $\mathbf{X}(k_{U_1})$ and finishing at some end-of-simulation condition (e.g., mission time, occurrence of a fixed number of events) or when $\phi(\mathbf{X}(k))$ down-crosses T_1; label the latter event as D_1.

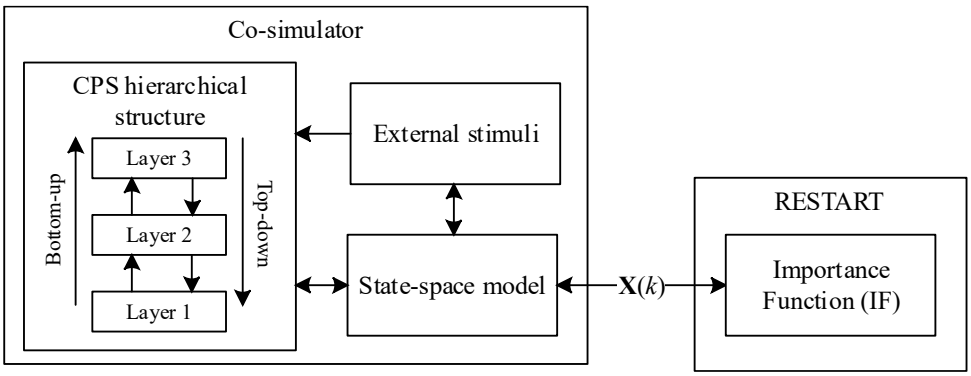


Figure 1 Integration of the master co-simulator (Futalef et al., 2022) with RESTART

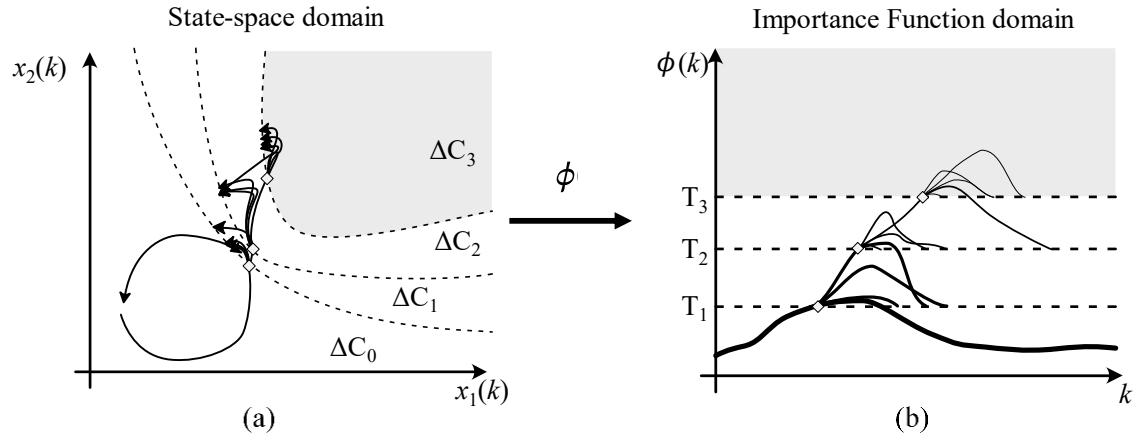


Figure 2 Visualization of RESTART for a two-dimensional state: in (a), the process evolves in the state-space domain, which is mapped simultaneously into the IF domain in (b), where new retrials emerge when the IF surpasses some threshold

4) If during any of the R_1 retrials $\phi(\mathbf{X}(k))$ up-crosses the threshold $T_2 > T_1$, steps 2) and 3) are repeated for labelled event(s) U_2, instants k_{U_2} and states $\mathbf{X}(k_{U_2})$ are saved, and $R_2 - 1$, $R_2 \in \mathbb{N}$, new retrials are simulated for each U_2, until the end-of-simulation condition is reached or T_2 is down-crossed.
5) Repeat steps 2) to 5) for all remaining thresholds and retrials, or until the end-of-simulation condition is reached, i.e., simulate $R_i - 1$ new retrials, $R_i \in \mathbb{N}$, when $\phi(\mathbf{X}(k))$ up-crosses any threshold T_i. If $\phi(\cdot)$ goes above several thresholds at once, say $\{T_i, T_{i+1}, \dots, T_{i+b}\}$ or b thresholds above T_i at once, then run $\prod_{l=i}^{i+b}(R_l - 1)$ retrials (i.e. a number equal to the product of the retrials that should be run for each threshold(s)).

Figure 2a shows an example of the application of RESTART to a two-dimensional process $\mathbf{X}(k) = (x_1(k), x_2(k))^{\mathsf{T}}$, whereas Figure 2b shows the corresponding mapping $\phi(\mathbf{X}(k))$. The rationale of RESTART is that it is more likely for a state $\mathbf{X}(k)$ to reach an intermediate region when it transitions from a state in a region of immediately lower importance, rather than jumping abruptly across several regions. Therefore, by oversampling states within high-importance regions, the likelihood of observing states of even greater importance increases. This goal is primarily achieved if $\phi(\cdot)$ effectively highlights relevant rare states, ultimately enabling RESTART to allocate the computational effort to simulating low probability events.

As highlighted in (M. Villén-Altamirano & Villén-Altamirano, 2006), the definition of the IF in step 1) is most critical for a successful implementation of RESTART. An effective IF should:

- Assign low importance values to high probability states;
- Provide similar importance values to states nearby a threshold;
- Reduce importance skipping, i.e., sudden IF jumps across several regions.

Typical IFs adopted for the analysis of various complex systems are briefly discussed hereafter. In (Janghoon Kim et al., 2013), the number of line outages is taken as IF to guide the simulation, but it is limited to dealing with static states and conditions of power grid components. In (J. Villén-Altamirano, 2014) and (Turati, Pedroni, and Zio (2016)), the cut sets of a system, found from its structure function, are used as IF to guide the sampling by RESTART; however, cut sets identification in large CPS is difficult, if not impossible (Di Maio et al., 2017, 2023). CPS vulnerability metrics, such as Energy Not Supplied (ENS) or System Average Interruption Frequency Index (SAIFI), can be used as IFs given that they offer a synthetic characterization of disruptive CPS scenarios (Di Maio et al., 2022); however, the cumulated information that they provide is strictly conditioned to the occurrence of the rare failure

events so that if no failure occurs, such metrics do not provide any useful information (e.g., they are zero throughout the simulation) and RESTART would act as crude MCS.

In this work, we introduce an original IF specific for the CRA of CPSs. Based on the centrality and susceptibility metrics of all the components of the CPS, we compute throughout the simulation its *Dynamic Network-level Vulnerability* (DNV) to measure the level of exposition to failure of the CPS and allocate the computational effort primarily to the system vulnerabilities to probabilistically favor the occurrence of CPS failures. For more detailed insights and examples on centrality and susceptibility metrics, please refer to Appendix A.

As mentioned in the Introduction, *centrality* is a concept used in network analysis to describe the influence that the single elements of a network have on its performance (Eusgeld et al., 2009). Since CPS are flow networks (e.g. flows of electricity, packets or substances among the multiple components), a suitable centrality metric is the Current Flow Betweenness Centrality (CF-BC) (Newman, 2005). This metric characterizes the importance of a component based on the average topology-induced flow passing through it, defined as follows. Let $G = (V, E)$ be a graph of the CPS, with V being the set of the n_V vertices (e.g., components) and E the n_E edges (e.g., interconnections) of the network, so that the total number of network elements is $n_V + n_E$. Given the sets of sources $S \subseteq V$ and targets $T \subseteq V$, the CF-BC of a j-th node is:

$$\mathbb{c}_j := \frac{\sum_{\{s<t\}} I_j^{(s-t)}}{(1/2)n_V(n_V-1)}, \tag{3}$$

where $I_j^{(s-t)}$ is the flow due to a source-target pair of components $s-t$, $s \in S$ and $t \in T$. We normalize all the centrality values by the maximum obtained centrality to ensure $\mathbb{c}_j \in [0,1]$, for any component j. Notice that this metric provides us with a ranking of the components that is source-target dependent; thus, by choosing the $s-t$ pairs, we can tune the exploration to focus on the accidental scenarios for a particular set of components.

The other quantity used to build the IF is the *susceptibility* of a component, which quantifies at any instant k its usage level (or, vice versa, its resourcefulness) with respect to design specifications and ratings, i.e., the stress when exposed to hazards/threats. Thus, a value of susceptibility equal to zero indicates "no usage" or "up" (i.e., resourceful component), whereas a value equal to one corresponds to "full usage" or "down" (i.e., resourceless component). Since each j-th component is characterized by its state vector $\mathbf{X}_j(k)$, we define its susceptibility $\mathbb{s}_j(k) \in [0,1]$ by considering that the l-th state variable in $\mathbf{X}_j(k)$ has a maximum factory rating $x_j^{+(l)}$ that cannot be exceeded (the definition for minimum ratings is dually equivalent):

$$\mathbb{s}_j(k) := \begin{cases} \dfrac{X_j^{(l)}(k)}{x_j^{+(l)}} & \text{if } j \text{ is operating;} \\ 1 & \text{otherwise.} \end{cases} \tag{4}$$

In line with (M. Villén-Altamirano & Villén-Altamirano, 2006), we combine the centrality in Eq. (3) with the susceptibility in Eq. (4) to provide a *weighted susceptibility* (Eq. (5)) that emphasizes the role of components that contribute to the vulnerability of the system when they are susceptible (closer to the limit $x_j^{+(l)}$):

$$\mathbb{v}_j(k) := \mathbb{c}_j \cdot \mathbb{s}_j(k). \tag{5}$$

We, then, obtain the DNV metric $\mathbb{V}(k)$ as the sum of the weighted susceptibilities of all components in the network:

$$\begin{aligned} \mathbb{V}(k) &= \sum_{j\in\{1,\dots,n_N+n_E\}} \mathbb{c}_j \cdot \mathbb{s}_j(k) \\ &= \sum_{j\in\{1,\dots,n_N+n_E\}} \mathbb{v}_j(k) \end{aligned} \tag{6}$$

In practice, the dynamic network-level vulnerability $\mathbb{V}(k)$ of the CPS can be obtained as shown in Figure 3 by:

1) modeling the CPS by means of the state vector $\mathbf{X}(k)$ and its dynamics as in Eq. (2);
2) mapping the CPS into a graph $G = (V, E)$, where vertices V are components (e.g., bar buses, generators, electrical loads, TLC gateways, controller units), and the edges E are interconnections (e.g., transmission lines, transformers, fiber-optic cables);
3) calculating the CF-BC for all the elements in G with Eq. (3), where the $(s - t)$ pairs are defined by the sources S that inject flows (e.g., generators, control center) and the targets T that receive flows (e.g., customers);
4) computing, throughout the simulation, the susceptibility (Eq. (4)) and the weighted susceptibility (Eq. (5)) of each component, to obtain the DNLV (Eq. (6)).

2.3 PERFORMANCE METRIC

For any scenario generation method $\mathcal{M}$ that produces n_{sim} simulation points in a dataset $\mathcal{D}_{\mathcal{M}} = \{\mathbf{X}_{(1)}, \dots, \mathbf{X}_{(n_{\text{sim}})}\}$, we can determine the exploration efficiency $\xi_{\phi}(T)$ of an IF ϕ; this metric computes the ratio between the number points in $\mathcal{D}_{\mathcal{M}}$ that fall within a generic importance region ΔC (defined by a generic threshold T in the range of ϕ) over the total number points n_{sim}:

$$\xi_{\phi}(T) = \frac{\sum_{n=1}^{n_{\text{sim}}} [\phi(\mathbf{X}_{(n)}) > T]}{n_{\text{sim}}}, \tag{7}$$

where the Iverson bracket $[\cdot]$ counts one when $\phi\left(\mathbf{X}_{(n)}(k)\right) > T$ is true or zero otherwise. The ξ_{ϕ} is then representative of the conditional probability associated to the region from T onward.

An effective IF ϕ, alongside its thresholds $\{T_0, T_1, \dots, T_m\}$, can be recognized through the values of ξ_{ϕ}, calculated over some given (generated) dataset $\mathcal{D}_{\mathcal{M}}$. A consistent decrease in ξ_{ϕ}, as thresholds increase, indicates a balanced probability of transitioning between importance zones: in RESTART, this pattern suggests a reduced skipping and smooth threshold crossings, allowing ϕ, in combination with the strategic oversampling of importance regions, to produce a balanced acceleration of relevant rare events occurrences while encouraging exploration of the state-space domain. In practice, a quasi-optimal thresholds setting by fixed number yields a balanced probability of transitioning between importance zones, i.e., equal conditional probabilities of transitioning from each region C_i to C_{i+1}. In other words, being m the number of thresholds, a balanced probability of transitioning between importance zones occurs when ξ_{ϕ} decreases by $1/(m + 1)$ for each threshold increase (M. Villén-Altamirano & Villén-Altamirano, 2011). Considering an alternative IF ϕ' with different thresholds $\{T_0', T_1', T_2', \dots\}$ (defined in the range of ϕ'), we can claim its superiority if $\xi_{\phi'}(T_i') > \xi_{\phi}(T_i)$ over $\mathcal{D}_{\mathcal{M}}$, for any $i \in \{0,1,2, \dots\}$: this means that ϕ' detects events of low probability (high importance) more often than ϕ by enabling the oversampling of those events not detected by ϕ.

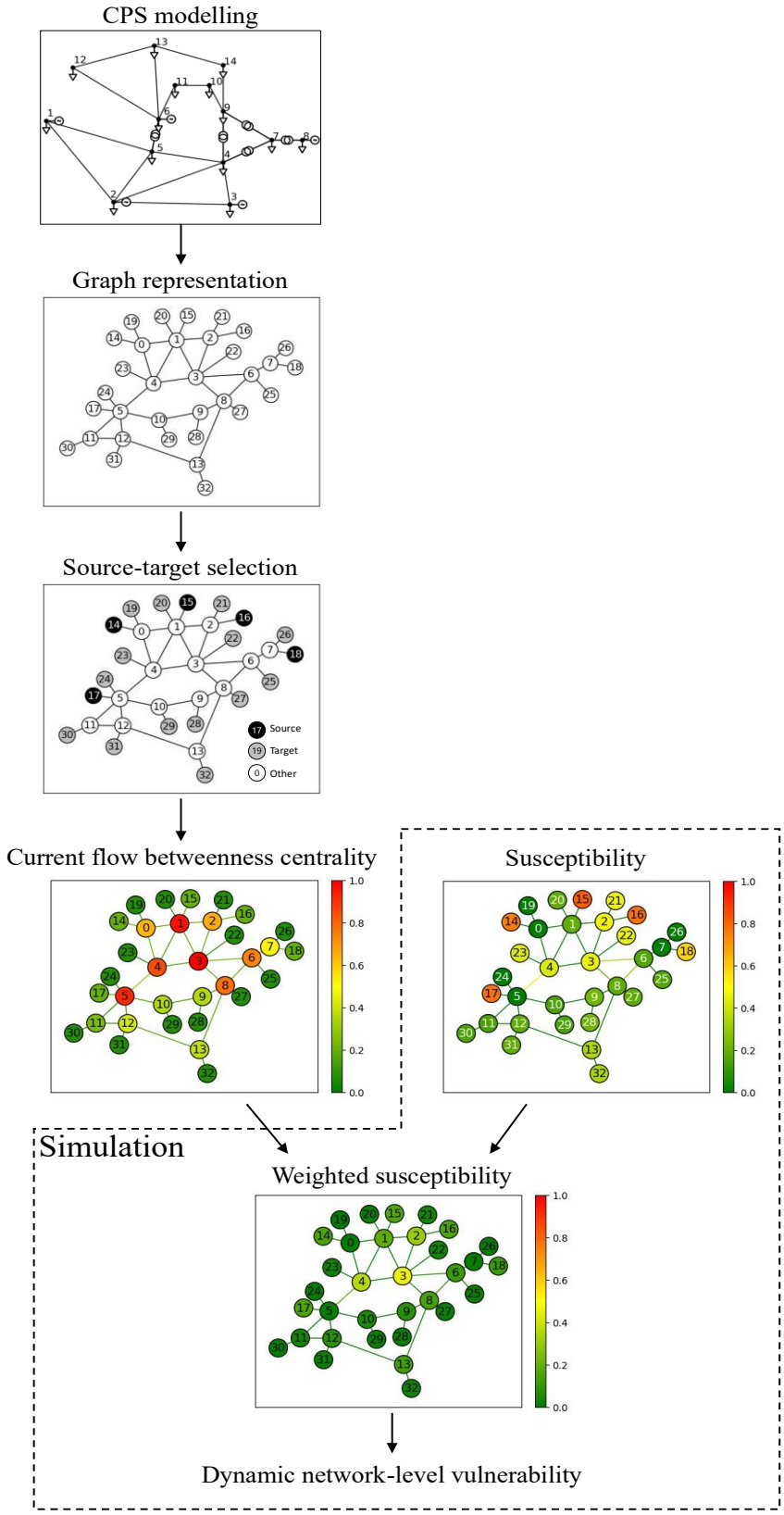


Figure 3 Steps for the calculation of the DNLV of a generic network

3 CASE STUDIES

This Section presents two case studies. In Section 3.1, we consider the IEEE14 power grid case (Christie, 1993) with a centralized controller. For the CRA of this system, RESTART is guided by the DNLV and traditional CPS vulnerability metrics (Power Generation Mismatch (PGM), number of down components, and Energy Not Supplied (ENS)) and the simulation performance is compared with a crude MCS. In Section 3.2, we consider an IP&TLC CPS (Di Maio et al., 2022), where data transmission between the power grid and the centralized controller is affected by the TLC; in this case, we compare the performance of crude MCS and RESTART guided by the DNLV. In both case studies, models and procedures are implemented in Python, where Pandapower (Thurner et al., 2018) is used for power flow calculations and NetworkX (Hagberg et al., 2008) for graph calculations. Simulations are performed on Windows 11 on an Intel i7 9750H computer at 2.2 GHz.

3.1 IEEE14 POWER GRID

The IEEE14 power grid (Christie, 1993) is composed of 14 buses, denoted by $\mathcal{B}$, which interconnect multiple electrical components: $\mathcal{C}$ the 11 electrical loads (customers) that demand active and reactive power, $\mathcal{G}$ the 2 controllable generators that input active and reactive power plus 3 synchronous condensers modelled as generators with zero active power inputs, $\mathcal{L}$ the 20 transmission lines that interconnect the buses, and $\mathcal{T}$ the 3 transformers that separate high and low voltage zones (Figure 4a).

Each component of the power grid is equipped with a Smart Monitoring Device (SMD) (e.g., a Phasor Measurement Unit (PMU)) that monitors, at each instant k, the component availability and collects electrical measurements. The SMDs send these values to a Control Center (CC) that computes control inputs for the generators by solving an Alternating Current Optimal Power Flow (AC-OPF) problem with the parameters described in (MATPOWER, 2014); such control inputs are, then, passed to the power grid generators. Without loss of generality, we use a Power Flow (PF) model to obtain the detailed power grid response, enabling the simulation of the hierarchical closed-loop control shown in Figure 4b. For the PF and the OPF, the slack bus is the available generator with the highest active power rating at instant k.

The vectors $\overline{\mathbf{p}}_d(k) = \left(\bar{p}_{d,1}(k), \dots, \bar{p}_{d,11}(k)\right)^{\mathsf{T}}$ and $\overline{\mathbf{q}}_d(k) = \left(\bar{q}_{d,1}(k), \dots, \bar{q}_{d,11}(k)\right)^{\mathsf{T}}$ group the active and reactive power demands of all the customers at any instant k, respectively, defining the demands of each customer $j \in \mathcal{C}$. The power grid uncontrolled inputs vector (i.e., external stimuli) $\mathbf{E}_{\mathrm{PG}}(k)$ is, thus, the concatenation of the active and reactive power demand vectors:

$$\mathbf{E}_{\mathrm{PG}}(k) = \begin{pmatrix} \overline{\mathbf{p}}_d(k) \\ \overline{\mathbf{q}}_d(k) \end{pmatrix} \in \mathbb{R}^{22} \tag{8}$$

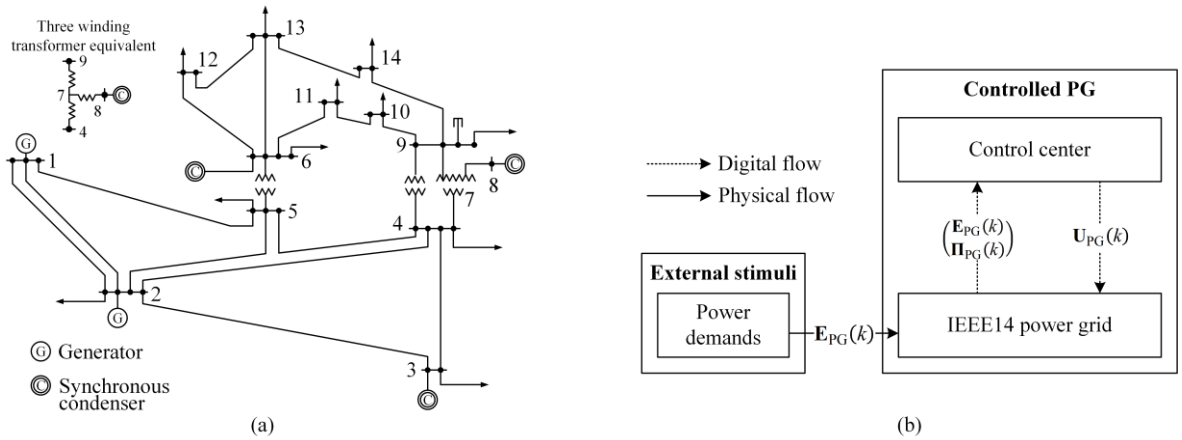


Figure 4 Visualization of the controlled IEEE14 power grid case. In (a), the power grid schematic diagram and in (b) the hierarchical control scheme

Table 2 Power grid failure and repair rates (Di Maio et al., 2022)

Component	$\lambda_{\text{Rep}}\ [h^{-1}]$	$\lambda_{\text{Fail}}\ [h^{-1}]$
Generator	24	3e-3
Transmission Line	12	3.5e-3
Transformer	4	19e-3

The values of $\bar{\mathbf{p}}_d(k)$ and $\bar{\mathbf{q}}_d(k)$ are taken from (Di Maio et al., 2022), consisting of 30 days of historical open data provided by (Terna, 2023), sampled every $\Delta t = 15\ [\text{min}]$. For generalization purposes, we process the Terna data by applying min-max normalization and, then, multiplying the results by the nominal active and reactive power load demands of each customer in the IEEE14 case.

The binary statuses of components are tracked at any instant k by the vectors $\boldsymbol{\pi}_{\text{Gen}}(k)$, $\boldsymbol{\pi}_{\text{Line}}(k)$, and $\boldsymbol{\pi}_{\text{Trafo}}(k)$ for generators, transmission lines and transformers, respectively. We place them in the binary status vector $\boldsymbol{\Pi}_{\text{PG}}(k)$ (Eq. (9)), for which we consider that generators $\mathcal{G}$, transmission lines $\mathcal{L}$, and transformers $\mathcal{T}$ can fail and be repaired with the exponential rates in Table 2, making $\boldsymbol{\Pi}_{\text{PG}}(k)$ follow a binary Markov chain with an initial condition $\boldsymbol{\Pi}_{\text{PG}}(0) = \mathbf{1}$ (all components are up):

$$\boldsymbol{\Pi}_{\text{PG}}(k) = \begin{pmatrix} \boldsymbol{\pi}_{\text{Gen}}(k) \\ \boldsymbol{\pi}_{\text{Line}}(k) \\ \boldsymbol{\pi}_{\text{Trafo}}(k) \end{pmatrix} \in \{0,1\}^{28} \tag{9}$$

The CC receives at each instant k both $\mathbf{E}_{\text{PG}}(k)$ and $\boldsymbol{\Pi}_{\text{PG}}(k)$ to create and solve an instance of the AC-OPF, returning in output the power grid controlled inputs $\mathbf{U}_{\text{PG}}(k)$, i.e., the voltage angles $\mathbf{v}_{\angle,\text{gen}}(k)$, active power generation $\mathbf{p}_{\text{g,gen}}(k)$ and voltage magnitudes $\mathbf{v}_{\text{m,gen}}(k)$ of controlled generators, and the voltages magnitude $\mathbf{v}_{\text{m,sc}}(k)$ setpoints of the synchronous condensers:

$$\mathbf{U}_{\text{PG}}(k) = \begin{pmatrix} \mathbf{v}_{\angle,\text{gen}}(k) \\ \mathbf{p}_{\text{g,gen}}(k) \\ \mathbf{v}_{\text{m,gen}}(k) \\ \mathbf{v}_{\text{m,sc}}(k) \end{pmatrix} \in \mathbb{R}^{9} \tag{10}$$

If the AC-OPF does not converge, we set $\mathbf{U}_{\text{PG}}(k) = \mathbf{0}$ (i.e., request all generators to shut down).

The control inputs are fed to each generator and, then, the PF is run to obtain the detailed response of the physical quantities in the power grid $\boldsymbol{\Lambda}_{\text{PG}}(k)$, made of, but not limited to: $\mathbf{p}_{\boldsymbol{d}}(k)$ and $\mathbf{q}_{\boldsymbol{d}}(k)$, i.e., the actual active and reactive power received by the customers, $\mathbf{p}_{\boldsymbol{g}}(k)$ and $\mathbf{q}_{\boldsymbol{g}}(k)$, i.e., the actual active and reactive power generations, $\mathbf{v}_{\boldsymbol{m}}(k)$ and $\mathbf{v}_{\angle}(k)$, i.e., the voltage magnitudes and angles at each bus:

$$\boldsymbol{\Lambda}_{\text{PG}}(k) = \begin{pmatrix} \mathbf{p}_d(k) \\ \mathbf{q}_d(k) \\ \mathbf{p}_g(k) \\ \mathbf{q}_g(k) \\ \mathbf{v}_m(k) \\ \mathbf{v}_{\angle}(k) \end{pmatrix} \in \mathbb{R}^{60} \tag{11}$$

The binary statuses $\boldsymbol{\Pi}_{\text{PG}}(k)$ (Eq. (9)) and physical quantities $\boldsymbol{\Lambda}_{\text{PG}}(k)$ (Eq. (11)) are appended within the state vector of the power grid $\mathbf{X}_{\mathbf{PG}}(k)$ (see the **Λ-Π** representation of Eq. (1)):

$$\mathbf{X}_{\mathbf{PG}}(k) = \begin{pmatrix} \boldsymbol{\Lambda}_{\text{PG}}(k) \\ \boldsymbol{\Pi}_{\text{PG}}(k) \end{pmatrix} \in \mathbb{R}^{88} \tag{12}$$

Table 3 Threshold values considered for each IF

IF	Thresholds
ϕ_1	$\{15.55, 62.22, 140.00, 161.37, 225.51, 332.40, 347.95, 394.62\}$
ϕ_2	$\{1,2,3,4,5,6,7,8\}$
ϕ_3	$\{0.04, 0.09, 0.13, 0.18, 0.22, 0.27, 0.31, 0.36\}$
ϕ_4	$\{1.99, 2.15, 2.64, 3.46, 4.60, 6.07, 7.86, 9.98\}$

RESTART is setup with a fixed number of retrials $R_i = 4$, for each threshold T_i, $i \in \{1,2,\dots\}$; the number of main trials R_0 is not set, letting the scenarios generation run until $\mathrm{T_{exec}} = 6[\mathrm{h}]$ is reached considering a mission time representative of a week of operation $\tau_f = 100[h]$ (i.e., enough to obtain at least one hundred crude MCS scenarios for comparison). In the following, we list the IFs that are used for the comparison of the results; for consistency, we develop $m = 8$ thresholds for all IFs (Table 3).

The *Power Generation Mismatch (PGM)* (ϕ_1) computes the difference between the apparent power generated in a scenario and that of a reference normal conditions scenario. Let $\mathbf{S}(k)$ and $\bar{\mathbf{S}}(k)$ be the vectors of actual and reference complex power generations at instant k, respectively. Then, $\phi_1(k)$ is the sum of magnitudes of the apparent power differences among all generators ($\phi_1(k) = 0$ if the actual system condition is equal to the reference one (i.e., normal conditions); $\phi_1(k) > 0$, otherwise):

$$\phi_1(k) = \sum_{j=1}^{5} \left| \left|\mathbf{S}^{(j)}(k)\right| - \left|\bar{\mathbf{S}}^{(j)}(k)\right| \right| \tag{13}$$

Thresholds (Table 3, 1st row) are defined by, first, sorting in ascending order the maximum active power ratings of the PV generators (i.e., $\{140.0, 332.4\}$[MW]): each of these ratings are one threshold. Next, we consider a temporary last threshold equal to the sum of the ratings (i.e., 472.4 [MW]); in-between these thresholds, we insert two more thresholds quadratically distanced from the one previously added. Finally, we remove the temporary last threshold.

The *number of down components* **(ϕ_2)** counts the number of unavailable components in the power grid at each instant k:

$$\phi_2(k) = 28 - \sum_{j=1}^{28} \mathbf{\Pi}_{\mathrm{PG}}^{(j)}(k) \tag{14}$$

Thresholds (Table 3, 2nd row) are set equal to integer values starting from zero, which are the values that $\phi_2(k)$ can take.

The *ENS percentage* (ϕ_3) computes at each instant k the accumulated portion of active power not delivered to each customer $j \in \mathcal{C}$:

$$\phi_3(k) = \frac{1}{11} \sum_{j=1}^{11} \left(1 - \frac{p_{d,j}(k)}{\bar{p}_{d,j}(k)}\right) \tag{15}$$

Thresholds (Table 3, 3rd row) are set considering that if a single customer does not receive any active power, ϕ_3 increments by $1/11 \approx 0.091$; for a half demand lost, the increment is $1/22 \approx 0.045$: we use this value to create the thresholds in an incremental manner.

The *Dynamic Network-Level Vulnerability* (ϕ_4) is that defined in Section 2 (see Eq. (6)). We compute the CF-BC considering that sources are the controllable generators in $\mathcal{G}$ (i.e., we drop synchronous condensers) and targets are all the customers in $\mathcal{C}$. Thresholds (Table 3, 4th row) are constructed by first obtaining the sum of all centralities, i.e., 11.091. For the first threshold, we consider that all components operate normally around 18% of their capacity (i.e., a base susceptibility of 0.18), yielding an average DNLV of 1.99: this is the first threshold. We repeat the procedure considering components operating at

Table 4 Performance comparison between MCS and RESTART for the IEEE14 case

	1	2	3	4	5	6	7	8	9	10
			ϕ_1		ϕ_2		ϕ_3		ϕ_4	
	Method	n_{sim}	$\xi_{\phi_1}(T_1^{\phi_1})$	$\xi_{\phi_1}(T_2^{\phi_1})$	$\xi_{\phi_2}(T_1^{\phi_2})$	$\xi_{\phi_2}(T_2^{\phi_2})$	$\xi_{\phi_3}(T_1^{\phi_3})$	$\xi_{\phi_3}(T_2^{\phi_3})$	$\xi_{\phi_4}(T_1^{\phi_4})$	$\xi_{\phi_4}(T_2^{\phi_4})$
1	MCS	40648	0.0129	0.008	0.0653	0.0059	0.0	0.0	**0.4667**	**0.1974**
2	ϕ_1	30280	<u>0.8327</u>	<u>0.8327</u>	0.8349	0.0375	0.0	0.0	**0.9216**	**0.8801**
3	ϕ_2	32287	0.0662	0.0478	<u>0.3254</u>	<u>0.0548</u>	0.0	0.0	**0.5246**	**0.3257**
4	ϕ_3	32193	0.0246	0.0148	0.0854	0.0056	<u>0.0</u>	<u>0.0</u>	**0.4810**	**0.2136**
5	ϕ_4	38218	0.5717	0.5486	0.604	0.0467	0.0	0.0	**<u>0.9683</u>**	**<u>0.8921</u>**

90% of their capacity, yielding a DNLV of 9.98: this is the last threshold. The remaining thresholds are inserted in between, distanced quadratically from the one previously inserted.

We compute the exploration efficiency ξ_ϕ of each ϕ (see Section 2.3) over the generated datasets to compare the performance of the scenario generation methods, considering, without loss of generality, the two lowest thresholds $T_i^\phi = \{T_1^\phi, T_2^\phi\}$ associated to each ϕ (i.e., the two highest probability regions). In Table 4, each row summarizes the results for a specific scenario generation method, along with the total number of simulation points produced by this procedure (n_{sim}). In the columns, we report (1) the exploration efficiency of the IF used in that procedure (underlined), and (2) the exploration efficiency of the other IFs with their corresponding thresholds calculated over the simulation points of that procedure, highlighting in bold those with the highest exploration efficiency. This comparison allows us to evaluate the effectiveness of the different IFs sample relevant configurations and generate correspondingly relevant scenarios.

When MCS is used (Table 4, row 1), $\xi_\phi(T_1^\phi)$ is noticeably larger than $\xi_\phi(T_2^\phi)$ for any ϕ, implying that MCS always allocates more computational resources than DNV into simulating high probability scenarios below the first threshold, which are irrelevant, i.e., normal conditions or failure events that do not alter the power grid normal conditions. This is evident for the ENS ϕ_3, which is equal to zero (row 1, columns 7 and 8), meaning that the simulations never explore scenarios in which customers are not supplied with energy. The noticeably larger values of $\xi_{\phi_4}(T_1^{\phi_4})$ and $\xi_{\phi_4}(T_2^{\phi_4})$ imply that the DNV detects low importance zones more often than MCS; indeed, more than half of the simulation points generated using MCS are in the lowest importance region of ϕ_4 (i.e., below the first threshold), suggesting that this IF is more effective for scenario exploration.

When RESTART is used, the calculated ξ_ϕ is larger than for MCS for all IFs, except the ENS ϕ_3, which behaves as MCS. These results suggest that RESTART effectively generates more relevant scenarios if the used IF is suitable for this purpose. In our case, all IFs satisfy the basic requirement of assigning low importance values to high probability events (i.e., events in which the CPS operates in normal conditions), yet some of them fail to reduce skipping or assign similar importance values to states near the thresholds, as explained by the obtained exploration efficiency values. Indeed, the IF with the highest ξ_ϕ in its own run is the DNV ϕ_4 (row 5, column 9), where the reduction from $\xi_{\phi_4}(T_1^{\phi_4})$ to $\xi_{\phi_4}(T_2^{\phi_4})$ of 0.08 (i.e., the difference between the values in columns 9 and 10, row 5) provides a reasonable setting of threshold values, enabling a balanced exploration of the importance regions (i.e., it approaches $1/9 = 0.\bar{1}$). On the other hand, ϕ_1, ϕ_2, and ϕ_3 show a reduced performance overall: their exploration efficiency is low when compared to the DNLV, even in their own runs, and the reduction between thresholds is either too large or too small. For ϕ_3, the zeros in columns 7 and 8 of Table 4 mean that this IF is not capable to detect higher importance zones above the first threshold, resulting useless for guiding RESTART in scenario exploration. Such reduced performances are explained by their shared inferior properties: these IFs react to failures in the system (i.e., component disconnections that yield a suboptimal system performance), which cause discontinuous jumps of (some) process

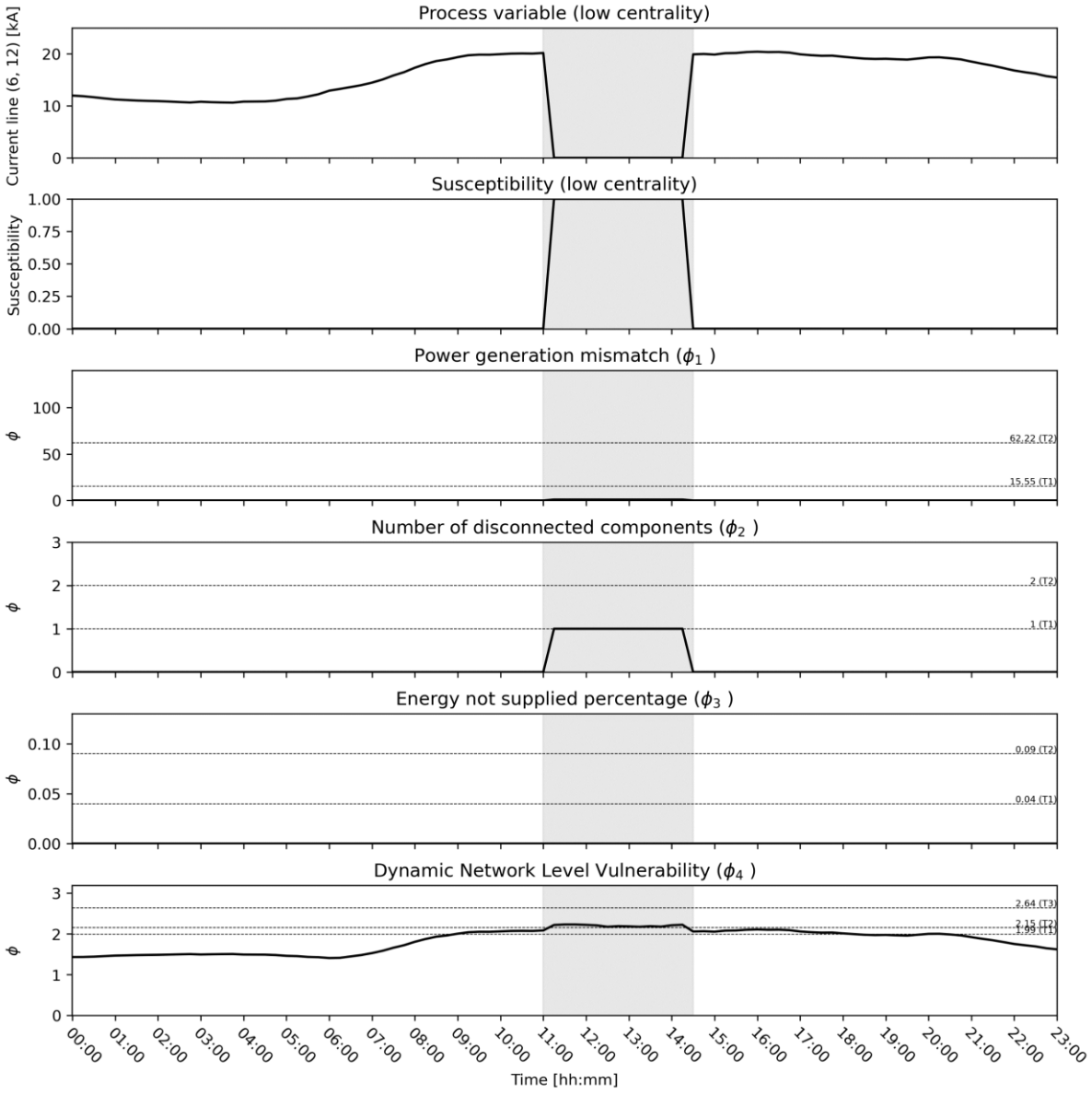


Figure 5 Example of disconnection of a low importance component (transmission line (6, 12))

variables and, thus, importance skipping. Since the susceptibility is weighted by the centrality in the DNV, the importance skipping is reduced for the low centrality components that are, in practice, the majority of the components in the network.

We complement the above discussion with a visualization of the evolution of the IFs across a simulation when components fail. Figure 5 presents a case in which a low centrality component (i.e., transmission line (6,12)), fails between 11:00 AM and 02:30 PM, but not sufficient to cause power dispatch issues. When the failure occurs, both ϕ_1 and ϕ_4 increase little due to the low importance of this component and ϕ_2 increases by one (i.e., one device down) as expected; ϕ_3, instead, does not provide any information about this scenario because all customers receive the demanded power. Figure 6 repeats the experiment for a higher centrality component (i.e., transmission line (1,5)), but still not enough to cause power dispatch issues. We observe a larger increase in ϕ_1 and ϕ_4; however, ϕ_2 increases by one just as before, providing no information about the differences between the two scenarios and ϕ_3, again, does not provide any information since the network is still satisfying the requested power demands. In both experiments, the ϕ_4 curves are smooth and always greater than zero, evolving accordingly with the state variables and increasing when components are more susceptible, suggesting an assignment of

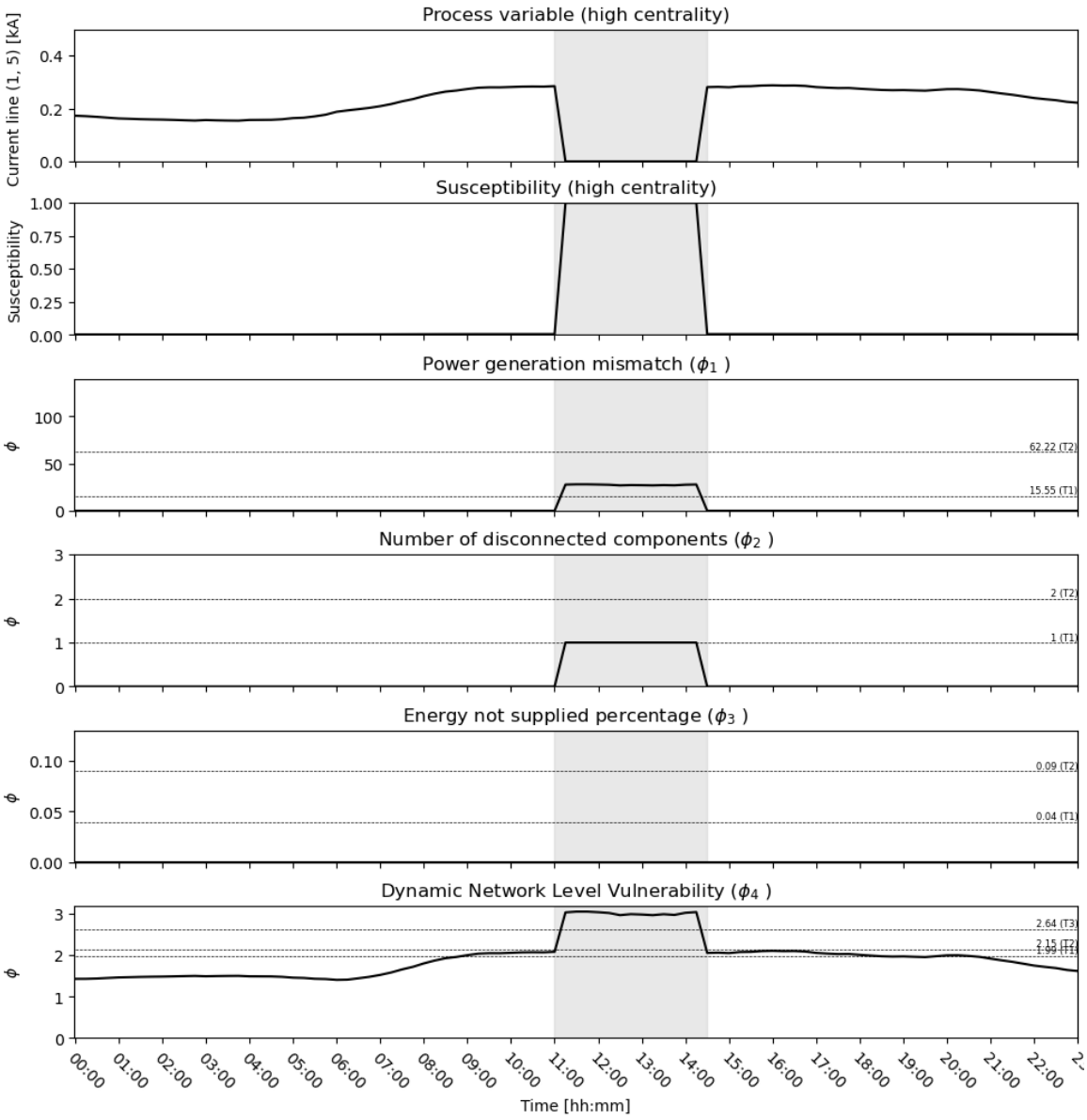


Figure 6 Example of disconnection of a high importance component (transmission line (1,5))

similar importance values to states near the thresholds. Such behavior is not observed in other IFs: as discussed above, ϕ_1, ϕ_2, and ϕ_3 do not provide any useful information when the system is operating in normal conditions, but only when a failure occurs.

3.2 INTEGRATED POWER AND TELECOMMUNICATION CPS

In this case study, we consider an Integrated Power and Telecommunication Infrastructure (IP&TLC) (Di Maio et al., 2022) that combines the IEEE14 power grid of Section 3.1 with the Telecommunication Network (TLCN) sketched in Figure 7a. The SMDs transmit the necessary data via the TLCN to the CC located at TLCN node 7, where the received measurements are processed to obtain the AC-OPF values to be sent back to the generators through the TLCN, enabling the hierarchical closed-loop control scheme shown in Figure 7b.

As in Section 3.1, the power grid is affected at any instant k by uncontrolled inputs, i.e., the evolving power demands $\mathbf{E}_{\mathrm{PG}}(k)$ (see Eq. (8)); the TLCN, instead, is not affected by external stimuli and, thus:

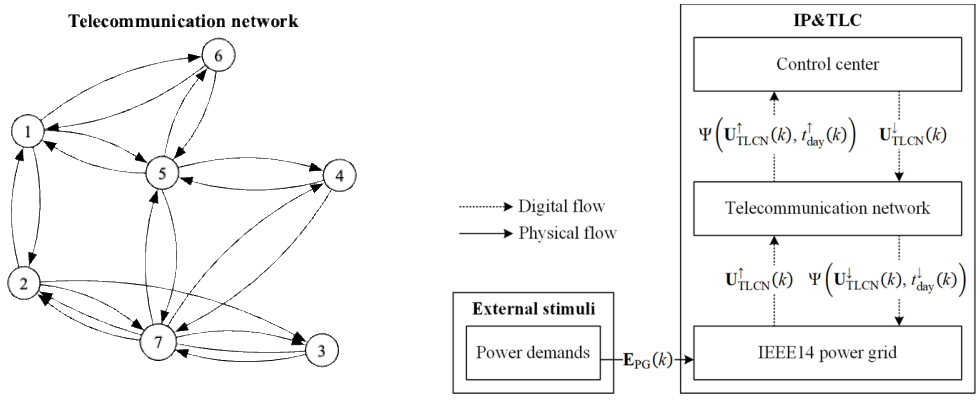
$$\mathbf{E}_{\mathrm{TLCN}}(k) = \emptyset \tag{16}$$

The power grid controlled inputs $\mathbf{U}_{\mathrm{PG}}(k)$ are the generators control settings (see Eq. (10)), and the TLCN controlled inputs $\mathbf{U}_{\mathbf{TLCN}}(k)$ contains the packets to be delivered: the bottom-up packet $\mathbf{U}^{\uparrow}_{\mathrm{TLCN}}(k)$ are the $\mathbf{E}_{\mathrm{PG}}(k)$ measurements and components status $\mathbf{\Pi}_{\mathrm{PG}}(k)$ (Eq. (9)) collected by the SMDs, whereas the top-down packets $\mathbf{U}^{\downarrow}_{\mathrm{TLCN}}(k)$ are the generators control settings as in the AC-OPF output:

$$\mathbf{U}^{\uparrow}_{\mathrm{TLCN}}(k) = \begin{pmatrix} \mathbf{E}_{\mathrm{PG}}(k) \\ \mathbf{\Pi}_{\mathrm{PG}}(k) \end{pmatrix} \in \mathbb{R}^{50}, \qquad \mathbf{U}^{\downarrow}_{\mathrm{TLCN}}(k) = \begin{pmatrix} \mathbf{v}_{\angle,\mathrm{gen}}(k) \\ \mathbf{p}_{\mathrm{g,gen}}(k) \\ \mathbf{v}_{\mathrm{m,gen}}(k) \\ \mathbf{v}_{\mathrm{m,sc}}(k) \end{pmatrix} \in \mathbb{R}^{10}$$
$$\mathbf{U}_{\mathbf{TLCN}}(k) = \begin{pmatrix} \mathbf{U}^{\uparrow}_{\mathrm{TLCN}}(k) \\ \mathbf{U}^{\downarrow}_{\mathrm{TLCN}}(k) \end{pmatrix} \tag{17}$$

An imperfect transmission along the TLCN might trigger packet losses, resulting in (1) incomplete data arriving at the CC, i.e., empty spaces in $\mathbf{U}^{\uparrow}_{\mathrm{TLCN}}(k)$, or (2) failure to deliver control settings, i.e., $\mathbf{U}^{\downarrow}_{\mathrm{TLCN}}(k)$ does not reach some generator(s). A transmission function Ψ (see Eq. (19) below) tests the success of these transmissions, which are influenced by the architecture and technology of the TLCN. We model such behaviors by adapting the approach in (Di Maio et al., 2022) as follows.

The TLCN is an undirected weighted graph $G = (V, E, t_d)$, where vertices V are the TLCN nodes, edges E are the TLCN links, and weights $t_d: E \times [0,\infty) \mapsto [0,\infty)$ assign to each edge $e \in E$ 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 instant k. These time delays follow a reflected Wiener process with a standard deviation σ and boundaries $t^- > 0$ and $t^+ > t^-$. Figure 8 exemplifies this behavior for a day of operation, considering $t^- = 30[\mathrm{ms}]$, $t^+ = 50[\mathrm{ms}]$, and $\sigma = 1$ [ms]. At any instant k, any node in V and edge in E might become unavailable with probability



(a) (b)

Figure 7 Description of the IP&TLCN: in (a), the topology of the TLCN; in b), the hierarchical scheme of the IP&TLCN, with the explicit bottom/up and top/down flows and uncontrollable inputs (external stimuli)

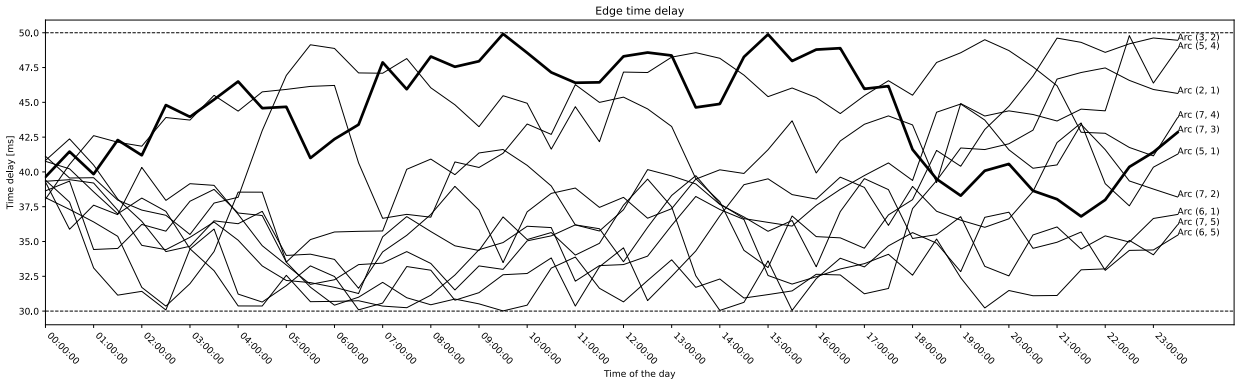


Figure 8 Visualization of the evolution of time delays

$p_d^v = 1\text{e} - 5$ and $p_d^e = 1\text{e} - 6$, respectively, due to sudden disconnections of nodes/edges or underlying TLCN protocol diversions; the binary status of nodes and links are represented by vectors $\boldsymbol{\pi}_{\text{node}}(k)$ and $\boldsymbol{\pi}_{\text{link}}(k)$, respectively.

For any origin-destination pair $(a, b) \in V \times V,\ a \neq b$, we compute a transmission path $\mathcal{P}_{ab} = (v_1, v_2, \dots, v_n),\ v_1 = a$ and $v_n = b$, using the TLCN adjacency matrix and the Dijkstra algorithm (Sedgewick, 2001). We mimic the process of (re)establishing network connections by caching these paths until they become disconnected (i.e., some TLC nodes or links become unavailable, yielding the established path unconnected): if that is the case, we recompute the path; however, if no path between (a, b) exists, then $\mathcal{P} = \emptyset$. Once the path is set, we compute its total time delay $T_d\left(\mathcal{P}_{ab}, t_{\text{day}}(k)\right)$ as:

$$T_d\left(\mathcal{P}_{ab}, t_{\text{day}}(k)\right) = \begin{cases} \sum_{i=1}^{n-1} t_d\left((v_i, v_{i+1}), t_{\text{day}}(k)\right) & \text{if P}\neq\emptyset \\ +\infty & \text{otherwise} \end{cases} \tag{18}$$

A transmission function $\Psi\left(\mathbf{d}, t_{\text{day}}\right)$ tests the transmission: if $T_d(\cdot) \leq t^{\max}$, for some timeout $t^{\max} > 0$, then the transmission is successful and the destination device at TLCN node b receives the packet $\mathbf{d}$; otherwise, the destination receives $\emptyset$ (empty packet):

$$\Psi\left(\mathbf{d}, t_{\text{day}}\right) = \begin{cases} \mathbf{d} & \text{if } T_d\left(\mathcal{P}_{ab}, t_{\text{day}}\right) \leq t^{\max}, \\ \emptyset & \text{otherwise} \end{cases} \tag{19}$$

Figure 9 exemplifies some cases in which the transmission can succeed or fail, considering $(a, b) = (6,7)$ and $t^{\max} = 80$ [ms].

The TLCN state vector $\mathbf{X}_{\text{TLCN}}(k)$ stores at any instant k the graph variables when doing the bottom-up ($\mathbf{X}_{\text{TLCN}}^{\uparrow}(k)$) and top-down ($\mathbf{X}_{\text{TLCN}}^{\downarrow}(k)$) transmissions: time delays $\mathbf{t}_d(k)$, and binary status of nodes $\boldsymbol{\pi}_{\text{node}}(k)$ and links $\boldsymbol{\pi}_{\text{link}}(k)$:

$$\mathbf{X}_{\text{TLCN}}^{\uparrow}(k) = \begin{pmatrix} \mathbf{t}_d^{\uparrow}(k) \\ \boldsymbol{\pi}_{\text{node}}^{\uparrow}(k) \\ \boldsymbol{\pi}_{\text{link}}^{\uparrow}(k) \end{pmatrix} \in \mathbb{R}^{27}, \qquad \mathbf{X}_{\text{TLCN}}^{\downarrow}(k) = \begin{pmatrix} \mathbf{t}_d^{\downarrow}(k) \\ \boldsymbol{\pi}_{\text{node}}^{\downarrow}(k) \\ \boldsymbol{\pi}_{\text{link}}^{\downarrow}(k) \end{pmatrix} \in \mathbb{R}^{27}$$
$$\mathbf{X}_{\text{TLCN}}(k) = \begin{pmatrix} \mathbf{X}_{\mathbf{TLCN}}^{\uparrow}(k) \\ \mathbf{X}_{\mathbf{TLCN}}^{\downarrow}(k) \end{pmatrix} \tag{20}$$

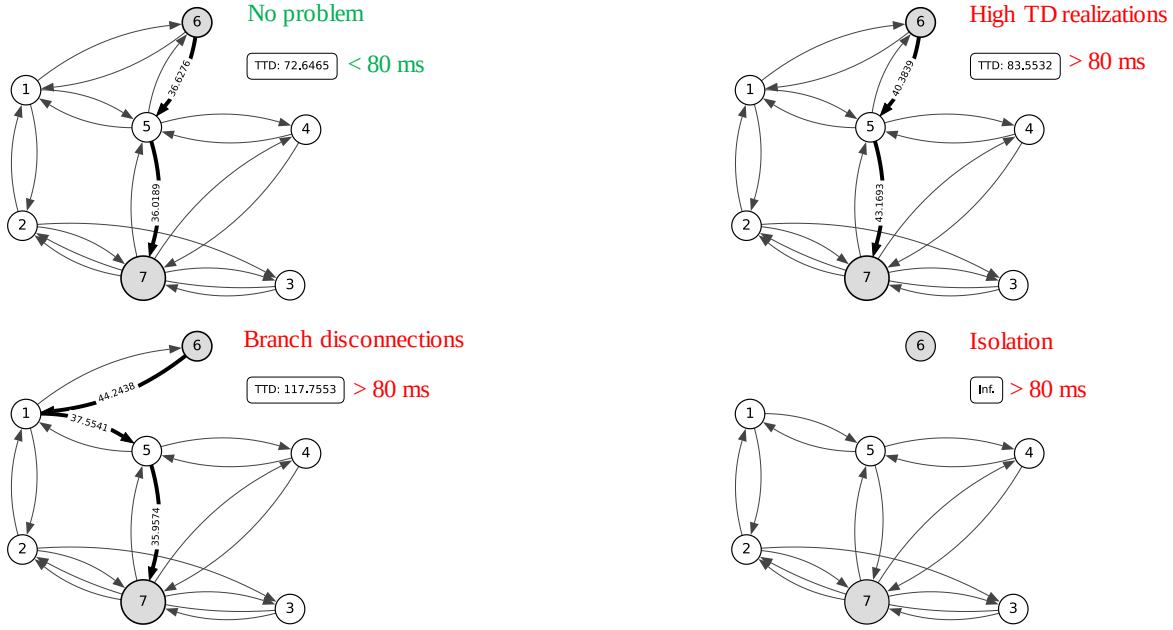


Figure 9 Different scenarios of successful and unsuccessful transmission between two nodes

We enable the use of RESTART for the IP&TLCN by concatenating the input and state vectors of the power grid and the TLCN (see Eqs. (8) and (16) for the external stimuli, Eqs. (10) and (17) for the controlled inputs, and Eqs. (12) and (20) for the state vector):

$$\begin{aligned}\mathbf{E}(k) &= \left(\mathbf{E}_{\mathrm{PG}}^{\top}(k), \mathbf{E}_{\mathrm{TLCN}}^{\top}(k)\right)^{\top},\\ \mathbf{U}(k) &= \left(\mathbf{U}_{\mathrm{PG}}^{\top}(k), \mathbf{U}_{\mathrm{TLCN}}^{\top}(k)\right)^{\top},\\ \mathbf{X}(k) &= \left(\mathbf{X}_{\mathrm{PG}}^{\top}(k), \mathbf{X}_{\mathrm{TLCN}}^{\top}(k)\right)^{\top},\end{aligned} \tag{21}$$

We compare MCS and RESTART guided by the DNLV. The TLCN considers $t^{-} = 10\ [ms]$ and $t^{+} = 40[ms]$ for all links, and timeouts of 80 [ms]. As in the previous case study, we consider a mission time of 100 [h] and a timestep $\Delta t = 15$ minutes. We increase the allocated computational time to 10 [h] to account for the greater model complexity of the IP&TLCN compared to the IEEE14 of the previous case study. RESTART parameters are set as in the previous section, with the thresholds of Table 5. In the DNLV, we assign source-targets for each network: for the power grid, we assign generators as sources and customers as targets (as in the previous case study), whereas, for the TLCN, we assign the SMDs as sources and the CC as target.

Table 6 compares the efficiency for the third and the fourth thresholds, using the DNLV. It can be seen that, as in the previous case study, RESTART is the method that best allocates computational resources for sampling high importance zones. The efficiency is lower than in the previous case, which suggests that the thresholds were placed conservatively (i.e., low).

In Figure 10, we visualize the impact of disconnecting a low centrality TLC link (i.e., (1, 5)), yielding a slight increase on the DNLV. On the other hand, Figure 11 shows the case of disconnection of a high centrality TLC link (i.e., (2, 7)), producing a larger DNLV increase and, thus, underlying the relevance of the scenario. These properties are similar to the ones of the previous case study, confirming that RESTART can effectively guide the scenario generation for the IP&TLCN using the DNLV.

Table 5 Threshold values considered for the IF in the IP&TLCN

IF	Thresholds
ϕ	$\{0.0, 2.44, 2.72, 3.57, 4.99, 6.97, 9.51, 12.62, 16.30\}$

Table 6 Performance comparison between MCS and RESTART for the IP&TLC case

	MCS		**RESTART**	
	$n_{\text{sim}} = 32180$		$n_{\text{sim}} = 29122$	
IF	$\xi(T_3)$	$\xi(T_4)$	$\xi(T_3)$	$\xi(T_4)$
ϕ	0.0124	0.001	**0.6867**	**0.4321**

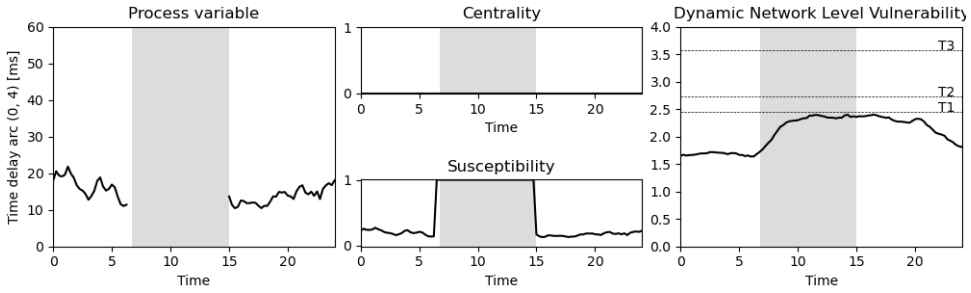


Figure 10 Disconnection of a low importance TLCN link and its impact in the DNLV

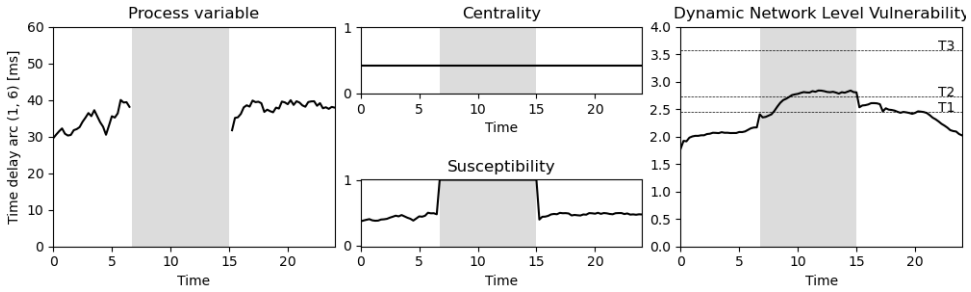


Figure 11 Disconnection of a high importance TLCN link and its impact in the DNLV

4 CONCLUSIONS

In this work, we have considered the rare event simulation method RESTART for generating relevant CPS scenarios within the context of CRA. We have tailored the method by introducing a novel dynamic IF that accounts for the relevance of the states of the components of the CPS to guide the simulation of the scenarios so as to favor conditions leading to CPS failure. The IF combines a flow centrality metric of the components in the CPS network with their susceptibility to failure. The results of the application of the proposed method to two case studies regarding the IEEE14 power grid and an IP&TLC network confirm that: (1) the exploration of RESTART depends strongly on the selected IF, (2) typical vulnerability metrics (PGM, number of down components, ENS percentage) fail to efficiently guide RESTART in generating relevant scenarios towards CPS failure, and (3) the novel IF proposed in this paper is shown effective with respect to the other approaches by means of an exploration efficiency metric. Since RESTART does not impose constraints on the dynamics of the system, this method can be extended to other applications without major adaptations. Future work will consist in post-processing the generated scenarios for performing the CRA of CPSs.

ACKNOWLEDGMENTS

We acknowledge the funding received from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement GREYDIENT No 955393.

APPENDIX A: CENTRALITY AND SUSCEPTIBILITY METRICS

For completion, this Appendix recalls the concept of vulnerability and provides brief details on centrality and susceptibility metrics, with some examples of application.

Vulnerability has been coined in the literature to highlight those components in a complex network that, when disrupted, cause noticeable disturbances to the overall system (Kröger & Zio, 2011). Currently, there is no consensus on the most appropriate approach to conduct a vulnerability assessment, although analytical and simulation-based methods have been highlighted as major approaches (Abedi et al., 2019; Devineni et al., 2020). Analytical solutions are faster and explainable, although they tackle simplified versions of the problem. Simulations, instead, are slower and require post-processing, though they can be more faithfully representative of the real system.

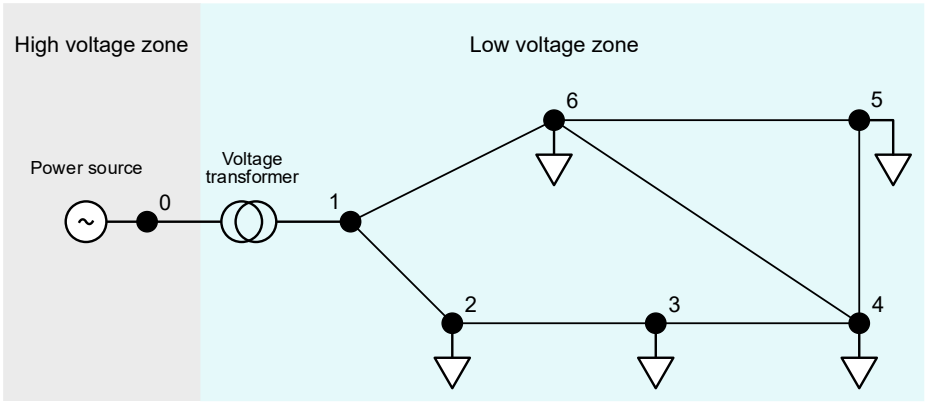
Centrality metrics are analytical approaches widely used for vulnerability assessment as they quantify the importance of elements in a network (vertices and links) according to a specific criterion (Eusgeld et al., 2009). Well-known centrality metrics include Degree Centrality, Betweenness Centrality (BC) and Closeness Centrality (CC). Specifically, BC is representative of the system function as it quantifies the average number of shortest paths passing through each node or link; however, since not all networks function based on shortest paths, new variants were introduced inspired by electrical network analysis, such as Current Flow BC (CF-BC) and Current Flow CC (CF-CC) (Newman, 2005).

While critical components are a primary focus, the combined failure of less critical components might be as disruptive as the former. By highlighting such less critical components, analytical methods can help guide detailed simulations and serve as informed exploration, thereby saving computational resources. For example, simulating scenarios where a critical component rarely fails might waste computational resources; instead, focusing on less critical components with higher failure chances can reveal unexpected configurations that lead to system failure.

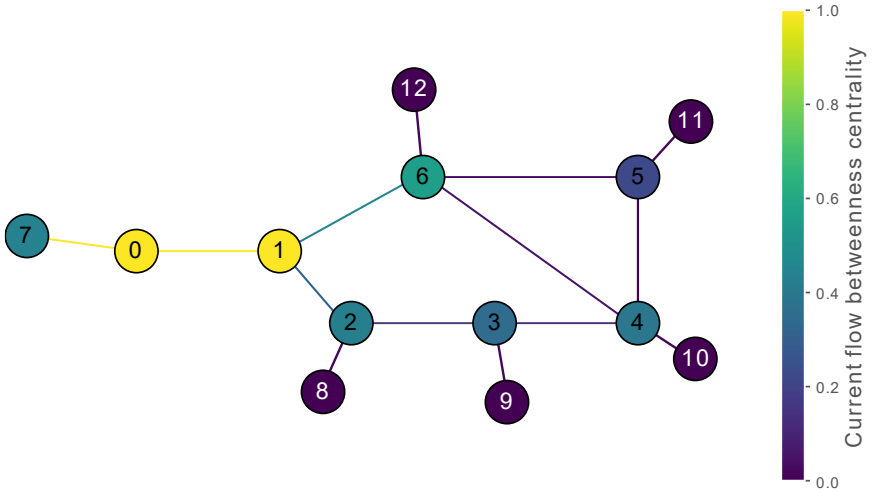
The susceptibility of a component is a quantification of its degree of exposition to a threat or hazard: the more(less) susceptible a component, the more(less) the chances it fails. Component stress can be adopted as a dynamic measure of susceptibility as it represents its usage with respect to adequate ranges of operation, such as those provided by manufacturers, regulations, or physical laws, along a dynamic scenario. Stress is versatile since it can be adopted in most (if not all) domains. For instance, in the electrical domain, any electrical device or machine operates within well-defined voltage and current

ranges. In the telecommunication domain, the communication latency among devices must be lower than a timeout setting. The same is true for other domains.

To illustrate the application of the metrics, consider the simple power grid example in Figure 12a, where a single power source supplies power to five loads, and the transformer separates between high and low voltage zones. From Figure 12a, evidently, interrupting the path between buses 0 and 1 will disconnect the power supply from the network. As a result, the power supply, the transformer, and buses 0 and 1 are the most critical components. In Figure 12b, we map the power grid into a graph and compute the CF-BC considering the power supply (node 7) as the source, and the loads (nodes 8 to 12) as the targets. The CF-BC effectively highlights the critical nodes and links identified above. However, we observe that the nodes that represent the buses have high centrality as well, ranging from ~0.55 to ~0.7; this is relevant since removing them will cause line disconnections that act as current paths. Node 6 also shows high importance given the two outgoing lines (6, 5) and (6,4) that split the flows.



(a) Simple power grid diagram



(b) Map of the simple power grid into a graph and the obtained CFBC considering source {7} and targets {8, 9, 10, 11, 12}

Figure 12 Demonstration of the mapping process of a simple power grid to its graph equivalent and the calculation of the CF-BC

REFERENCES

Abedi, A., Gaudard, L., & Romerio, F. (2019). Review of major approaches to analyze vulnerability in power system. *Reliability Engineering & System Safety*, *183*, 153–172. https://doi.org/10.1016/j.ress.2018.11.019

Au, S.-K., & Beck, J. L. (2001). Estimation of small failure probabilities in high dimensions by subset simulation. *Probabilistic Engineering Mechanics*, *16*(4), 263–277. https://doi.org/10.1016/s0266-8920(01)00019-4

Au, S.-K., & Patelli, E. (2016). Rare event simulation in finite-infinite dimensional space. *Reliability Engineering and System Safety*, *148*, 67–77. https://doi.org/10.1016/j.ress.2015.11.012

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

Boer, P.-T. de, Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A Tutorial on the Cross-Entropy Method. *Annals of Operations Research*, *134*(1), 19–67. https://doi.org/10.1007/s10479-005-5724-z

Cadini, F., Agliardi, G. L., & Zio, E. (2017). Estimation of rare event probabilities in power transmission networks subject to cascading failures. *Reliability Engineering and System Safety*, *158*, 9–20. https://doi.org/10.1016/j.ress.2016.09.009

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

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

Dang, C., Valdebenito, M. A., Faes, M. G. R., Song, J., Wei, P., & Beer, M. (2023). Structural reliability analysis by line sampling: A Bayesian active learning treatment. *Structural Safety*, *104*, 102351. https://doi.org/10.1016/j.strusafe.2023.102351

Devineni, P., Kay, B., Lu, H., Tabassum, A., Chintavali, S., & Lee, S. M. (2020). Toward Quantifying Vulnerabilities in Critical Infrastructure Systems. *2020 IEEE International Conference on Big Data (Big Data)*, 2884–2890. https://doi.org/10.1109/BigData50022.2020.9377730

Di Maio, F., Baronchelli, S., Vagnoli, M., & Zio, E. (2017). Determination of prime implicants by differential evolution for the dynamic reliability analysis of non-coherent nuclear systems. *Annals of Nuclear Energy*, *102*, 91–105. https://doi.org/https://doi.org/10.1016/j.anucene.2016.12.018

Di Maio, F., Pettorossi, C., & Zio, E. (2023). Entropy-driven Monte Carlo simulation method for approximating the survival signature of complex infrastructures. *Reliability Engineering & System Safety*, *231*, 108982. https://doi.org/10.1016/j.ress.2022.108982

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: Vol. Volume 1* (World Scientific, pp. 477–504). https://doi.org/10.1142/9789813225619_0013

Eusgeld, I., Kröger, W., Sansavini, G., Schläpfer, M., & Zio, E. (2009). The role of network theory and object-oriented modeling within a framework for the vulnerability analysis of critical

infrastructures. *Reliability Engineering & System Safety*, *94*(5), 954–963. https://doi.org/https://doi.org/10.1016/j.ress.2008.10.011

Futalef, J.-P., Maio, F. Di, & 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

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

Glasserman, P., Heidelberger, P., Shahabuddin, P., & Zajic, T. (1999). Multilevel Splitting for Estimating Rare Event Probabilities. *Operations Research*, *47*(4), 585–600. http://www.jstor.org/stable/223163

Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring Network Structure, Dynamics, and Function using NetworkX. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), *Proceedings of the 7th Python in Science Conference* (pp. 11–15).

Janghoon Kim, Bucklew, J. A., & Dobson, I. (2013). Splitting method for speedy simulation of cascading blackouts. *2013 IEEE Power & Energy Society General Meeting*, 1–1. https://doi.org/10.1109/PESMG.2013.6672598

Kapteyn, M. G., Knezevic, D. J., Huynh, D. B. P., Tran, M., & Willcox, K. E. (2022). Data-driven physics-based digital twins via a library of component-based reduced-order models. *International Journal for Numerical Methods in Engineering*, *123*(13), 2986–3003. https://doi.org/10.1002/nme.6423

Kroese, D. P., Taimre, T., & Botev, Z. I. (2011). Rare-Event Simulation. In *Handbook of Monte Carlo Methods* (pp. 381–419). John Wiley & Sons, Ltd. https://doi.org/10.1002/9781118014967.ch10

Kröger, W., & Zio, E. (2011). *Vulnerable Systems*. Springer London. https://doi.org/10.1007/978-0-85729-655-9

MATPOWER. (2014). *Power flow data for IEEE 14 bus test case.* https://matpower.org/docs/ref/matpower5.0/case14.html

Newman, M. E. J. (2005). A measure of betweenness centrality based on random walks. *Social Networks*, *27*(1), 39–54. https://doi.org/10.1016/j.socnet.2004.11.009

Pedroni, N., & Zio, E. (2017). An Adaptive Metamodel-Based Subset Importance Sampling approach for the assessment of the functional failure probability of a thermal-hydraulic passive system. *Applied Mathematical Modelling*, *48*, 269–288. https://doi.org/10.1016/j.apm.2017.04.003

Peherstorfer, B., Willcox, K., & Gunzburger, M. (2018). Survey of multifidelity methods in uncertainty propagation, inference, and optimization. *SIAM Review*, *60*(3), 550–591. https://doi.org/10.1137/16M1082469

Piccinelli, R., Sansavini, G., Lucchetti, R., & Zio, E. (2017). A General Framework for the Assessment of Power System Vulnerability to Malicious Attacks. *Risk Analysis*, *37*(11), 2182–2190. https://doi.org/https://doi.org/10.1111/risa.12781

Rubino, G., & Tuffin, B. (2009). *Rare Event Simulation using Monte Carlo Methods* (G. Rubino & B. Tuffin, Eds.). Wiley. https://doi.org/10.1002/9780470745403

Sedgewick, R. (2001). *Algorithms in C, Part 5: Graph Algorithms, Third Edition* (Third). Addison-Wesley Professional.

Sezen, H., Hur, J., Smith, C., Aldemir, T., & Denning, R. (2019). A computational risk assessment approach to the integration of seismic and flooding hazards with internal hazards. *Nuclear Engineering and Design*, *355*, 110341. https://doi.org/https://doi.org/10.1016/j.nucengdes.2019.110341

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

Thurner, L., Scheidler, A., Schafer, F., Menke, J.-H., Dollichon, J., Meier, F., Meinecke, S., & Braun, M. (2018). Pandapower—An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems. *IEEE Transactions on Power Systems*, *33*(6), 6510–6521. https://doi.org/10.1109/TPWRS.2018.2829021

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

Villén-Altamirano, J. (2014). Asymptotic optimality of RESTART estimators in highly dependable systems. *Reliability Engineering & System Safety*, *130*, 115–124. https://doi.org/10.1016/j.ress.2014.05.012

Villén-Altamirano, M., & Villén-Altamirano, J. (2006). On the efficiency of RESTART for multidimensional state systems. *ACM Transactions on Modeling and Computer Simulation*, *16*(3), 251–279. https://doi.org/10.1145/1147224.1147227

Villén-Altamirano, M., & Villén-Altamirano, J. (2011). The Rare Event Simulation Method RESTART: Efficiency Analysis and Guidelines for Its Application. In *Network Performance Engineering: A Handbook on Convergent Multi-Service Networks and Next Generation Internet* (pp. 509–547). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-02742-0_22

Zio, E. (2013). The Monte Carlo Simulation Method for System Reliability and Risk Analysis. In *Springer Series in Reliability Engineering*. Springer London. https://doi.org/10.1007/978-1-4471-4588-2

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

Zio, E., & Pedroni, N. (2010). An optimized Line Sampling method for the estimation of the failure probability of nuclear passive systems. *Reliability Engineering & System Safety*, *95*(12), 1300–1313. https://doi.org/https://doi.org/10.1016/j.ress.2010.06.007

Zio, E., Piccinelli, R., Delfanti, M., Olivieri, V., & Pozzi, M. (2012). Application of the load flow and random flow models for the analysis of power transmission networks. *Reliability Engineering and System Safety*, *103*, 102–109. https://doi.org/10.1016/j.ress.2012.02.005

List of acronyms

AI	Artificial Intelligence	LS	Line Sampling
BBM	Black-Box Model	MCS	Monte Carlo Simulation
CC	Control Center	OPF	Optimal Power Flow
CE	Cross-Entropy	PF	Power Flow
CPS	Cyber-Physical System	PGM	Power Generation Mismatch
CRA	Computational Risk Assessment	PMU	Phasor Measurement Unit
DNV	Dynamic Network-level Vulnerability	RESTART	Repetitive Simulation Trials After Reaching Thresholds
ENS	Energy Not Supplied	SAIFI	System Average Interruption Frequency Index
GBM	Grey-Box Model	SMD	Smart Monitoring Device
IF	Importance Function	SS	Subset Simulation
IP&TLC	Integrated Power and Telecommunication	TLCN	Telecommunication Network
IS	Importance Sampling	WBM	White-Box Model

List of symbols

k	Simulation instant	$\mathcal{C}$	Set of power grid customers
t_k	Simulation time at instant k	$\mathcal{G}$	Set of power grid generators
$t_{\mathrm{day}}(k)$	Time of the day at instant k	$\mathcal{L}$	Set of power grid lines
Δt	Simulation time step	$\mathcal{T}$	Set of power grid transformers
n_x	Number of state variables	$\mathbf{E}_{\mathrm{PG}}(k)$	Vector of power grid uncontrolled inputs
$n_{x,j}$	Number of state variables of the j-th component	$\mathbf{U}_{\mathrm{PG}}(k)$	Vector of power grid-controlled inputs
n_e	Number of uncontrolled inputs	$\mathbf{\Lambda}_{\mathrm{PG}}(k)$	Vector of power grid continuous variables
$n_{e,j}$	Number of uncontrolled inputs of the j-th component	$\mathbf{\Pi}_{\mathrm{PG}}(k)$	Vector of power grid discrete variables
n_u	Number of controlled inputs	$\mathbf{X}_{\mathbf{PG}}(k)$	Power grid state vector
$n_{u,j}$	Number of controlled inputs of the j-th component	$\boldsymbol{\pi}_{\mathrm{Gen}}(k)$	Vector of generators statuses
$\mathbf{X}(k)$	State vector	$\boldsymbol{\pi}_{\mathrm{Line}}(k)$	Vector of line statuses
$\mathbf{E}(k)$	Uncontrolled inputs	$\boldsymbol{\pi}_{\mathrm{Trafo}}(k)$	Vector of transformers statuses
$\mathbf{U}(k)$	Controlled inputs	$\mathbf{v}_{\angle,\mathrm{gen}}(k)$	Vector of generators voltages angles
$\mathbf{\Lambda}(k)$	Continuous process variables	$\mathbf{p}_{\mathrm{g,gen}}(k)$	Vector of generators active power setpoints
$\mathbf{\Pi}(k)$	Discrete process variables	$\mathbf{v}_{\mathrm{m,gen}}(k)$	Vector of generators voltages magnitudes
$\boldsymbol{F}(\cdot)$	State-transition function	$\mathbf{v}_{\mathrm{m,sc}}(k)$	Vector of synchronous condenser voltage magnitudes
$\mathbf{X}_j(k)$	State vector of the j-th component	$\mathbf{p}_d(k)$	Vector of delivered active power demands
$\mathbf{X}_{(n)}$	n-th state vector in an ordered dataset	$\mathbf{q}_d(k)$	Vector of delivered reactive power demands
$\boldsymbol{x}_j^{+(l)}$	Maximum rating of the l-th state variable of component j	$\mathbf{p}_g(k)$	Vector of generated active power
$\mathbf{E}_j(k)$	Uncontrolled inputs of the j-th component	$\mathbf{q}_g(k)$	Vector of generated reactive power
$\mathbf{U}_j(k)$	Controlled inputs of the j-th component	$\mathbf{v}_m(k)$	Vector of voltage magnitudes
Ω	State-space domain	$\mathbf{v}_\angle(k)$	Vector of voltage angles
$I_j^{(s-t)}$	Flow across component j induced by s and t	$\mathbf{S}(k)$	Vector of complex power generation
$\mathbb{C}_j$	Centrality of the j-th component	$\bar{\mathbf{S}}(k)$	Vector of complex power generation in normal conditions
$\mathbb{s}_j(k)$	Susceptibility of the j-th component	$\mathbf{E}_{\mathrm{TLCN}}(k)$	Vector of TLCN uncontrolled inputs
$\mathbb{v}_j(k)$	Weighted susceptibility of the j-th component	$\mathbf{U}_{\mathbf{TLCN}}(k)$	Vector of TLCN controlled inputs
$\mathbb{V}(k)$	Dynamic Network-level Vulnerability	$\mathbf{U}_{\mathrm{TLCN}}^{\uparrow}$	Bottom-up TLCN packet
$\phi(\cdot)$	Importance Function	$\mathbf{U}_{\mathrm{TLCN}}^{\downarrow}(k)$	Top-down TLCN packet

C_i	Importance region i	$\mathbf{X_{TLCN}}(k)$	TLCN state vector
T_i^{ϕ}	Threshold that defines C_i in the range of ϕ	$\Psi(\cdot)$	TLCN transmission function
ΔC_i	Importance region between C_i and C_{i-1}	$t_d(\cdot)$	Time-dependent TLC link time delay
R_i	Retrials associated with C_i	$\mathcal{P}_{ab}$	TLC path between a and b
$\xi_{\phi}(\cdot)$	Exploration efficiency of ϕ	$T_d(\cdot)$	Total time delay
$\mathcal{D}_{\mathcal{M}}$	Dataset generated using method $\mathcal{M}$	p_d^v	Disconnection probability of TLC node
m	Number of thresholds	p_d^e	Disconnection probability of TLC link
$\overline{\mathbf{p}}_d(k)$	Vector of active power demands	$\boldsymbol{\pi}_{\text{node}}(k)$	Vector of TLC node statuses
$\overline{\mathbf{q}}_d(k)$	Vector of reactive power demands	$\boldsymbol{\pi}_{\text{link}}(k)$	Vector of TLC link statuses
$\mathcal{B}$	Set of power grid buses		