
A Framework for Integrating Virtualized PAC into Availability Model of a Digital Substation: An Exploratory Adaptation of Software Aging and Rejuvenation Model

[Rizwan Rafique Syed](#) * and Hans Kristian Høidalen

Posted Date: 8 April 2026

doi: 10.20944/preprints202604.0602.v1

Keywords: digital substation automation system (DSAS); semi-Markov process (SMP); Markov chain (MC); substation communication network (SCN); software aging and rejuvenation (SAR); virtual protection automation and control (VPAC); virtual machine (VM)



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a [Creative Commons CC BY 4.0 license](#), which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

A Framework for Integrating Virtualized PAC into Availability Model of a Digital Substation: An Exploratory Adaptation of Software Aging and Rejuvenation Model

Rizwan Rafique Syed *  and Hans Kristian Høidalen

Department of Electric Energy, Norwegian University of Science and Technology, 7034 Trondheim, Norway

* Correspondence: rizwan.r.syed@ntnu.no

Abstract

Software aging and the corresponding need for system rejuvenation are well-established concepts in computer science. As virtualization technologies are increasingly adopted within electric power utility infrastructures, early investigation into Software Aging and Rejuvenation (SAR) models, aging indicators, and empirical data collection becomes essential. Given the critical role of the electric power grid and the high dependability requirements of the protection and control systems that support its operation, proactive research in this area is timely and necessary. Motivated by this need, this work proposes a hierarchical framework that integrates an SAR model into the Reliability Block Diagram (RBD) representation of a Digital Substation Automation System (DSAS). The analysis shows that, for the selected parameter set, incorporating SAR into the VPAC reliability model results in higher estimated failure rates and increased annual downtime relative to hardware-only models. When combined with substation primary system indices, however, the overall reliability indices remain largely unchanged, aside from reduced outage duration attributed to improved switching performance enabled by the DSAS architecture. Further examination reveals that the limited influence of SAR is primarily due to the lack of historical failure-mode data for the secondary system. Availability of such empirical data is expected to significantly affect combined reliability indices and improve the accuracy of reliability evaluations. This highlights the importance of systematic data collection and aging-indicator analysis as utility infrastructures transition toward virtualized and software-dependent architectures.

Keywords: digital substation automation system (DSAS); semi-Markov process (SMP); Markov chain (MC); substation communication network (SCN); software aging and rejuvenation (SAR); virtual protection automation and control (VPAC); virtual machine (VM)

1. Introduction

The power grid is undergoing a significant transformation toward a smart grid paradigm. Since protection, control, and monitoring functions in a smart grid are predominantly executed within substations, achieving overall grid reliability fundamentally depends on the reliability of these substations. As the transition toward a net-zero energy system accelerates, digital substations (DS), serving as critical nodes within this evolving infrastructure, are advancing to leverage modern developments in communication and computing technologies [1]. The secondary system of a digital substation consists of a Substation Automation System (SAS) and a Protection, Automation, and Control (PAC) system. The SAS typically manages higher level functions within the substation, including communication with Supervisory Control and Data Acquisition (SCADA) systems, interfacing with the Human Machine Interface (HMI), and handling alarms. In contrast, the PAC system mainly provides protection for primary assets as well as real-time control and monitoring at the lower level. This complete secondary

system in an automated substation is oftentimes referred to as DSAS in the literature. PAC functions, which were mainly distributed in the beginning (i.e., different Bay Control Units (BCUs) and Protection IEDs in a substation) [2], are transitioning towards a centralized and virtualized PAC system. The transformation borrowed from the enabling technology of virtualization from the field of computer science has proved to be promising as a result of early performance investigations [3]. Traditionally, physical PAC devices comprised tightly integrated hardware and software layers, each designed and optimized for a specific set of protection and control functions. Because these devices were delivered as purpose-built units, their performance characteristics were predetermined and guaranteed by the manufacturer. With the introduction of virtualization, these layers are now decoupled, enabling the hardware platform to be selected independently of the PAC application. In this model, the vendor provides the PAC functionality as software, while utilities can deploy it on robust hardware computing platforms that meet their operational requirements and can be adapted as the grid evolves. This architectural separation offers utilities significant flexibility, as it allows them to mix and match components from different suppliers to construct a solution tailored to their needs. However, this increased freedom also introduces a critical challenge: ensuring that the independently sourced hardware and software components operate seamlessly and reliably together, an essential requirement given the stringent dependability expectations of power system infrastructure.

Aging of the software layer in the virtualized architecture poses a critical performance [5] and reliability challenge for long-running systems, where prolonged execution leads to gradual performance degradation and increased failure likelihood. This degradation typically results from aging-related bugs that accumulate over time through mechanisms such as memory leaks, resource exhaustion, and corrupted internal states. These effects progress along the fault-error-failure chain and are often reflected in observable indicators, including rising resource consumption and reduced responsiveness [6]. To counter these effects, software rejuvenation provides a proactive recovery strategy that restores system health through controlled restarts or state refresh operations [7]. Rather than removing underlying defects, rejuvenation aims to release consumed resources and reset deteriorated system states, thereby postponing failures associated with cumulative errors. This technique is especially effective in continuously operating environments where planned rejuvenation can prevent unscheduled outages. Its effectiveness depends on determining an appropriate rejuvenation schedule that balances system availability with maintenance overhead, making timing decisions a central challenge [8]. Considering the pivotal role of dependability and performance aspects during the design and operation stages of the system life cycle, an exploration of model choices and system failure mechanisms during the pilot deployment stage [4] can build confidence prior to real-world deployments.

To the author's knowledge, no work exists to date that addresses the Software Aging and Rejuvenation (SAR) model aspect and its impact on substation reliability indices; this may be due to the infancy of virtualization technology in digital substations. The phenomenon, however, is well understood and has been the topic of research in the field of computer science. Owing to this fact we will present here some important work of SAR pertinent to virtualization technology in the field of computer science. This background serves both as a relevant literature review and a brief introduction to the field for power system researchers.

In virtualized environments, software aging appears as performance degradation and resource exhaustion, exacerbated by the additional abstraction layers of virtualization. Aging analysis seeks to estimate the likely time to failure caused by these effects and enable timely rejuvenation. Existing techniques fall into three main categories: model-based, measurement-based, and hybrid approaches [8]. Model-based techniques use analytical and stochastic models to derive optimal rejuvenation schedules. Common formalism examples include continuous time Markov chain (CTMC) [9], semi-Markov process (SMP) [10], Markov regenerative process (MRGP) [11], and Petri-net-based models such as stochastic Petri-net (SPN) [12] and stochastic reward net (SRN) [13], applied across configurations involving single or multiple hosts and various Virtual Machine (VM) states (cold, warm, migrated, failover). These approaches enable systematic evaluation of alternative rejuvenation policies but

often rely on simplifying assumptions that reduce accuracy in dynamic virtualized environments [8]. Measurement-based approaches rely on observable aging indicators at both the Virtual Machine Monitor (VMM) and VM layers. System-level metrics capture CPU, memory, storage, and network usage, while VM-level indicators include latency, throughput, and Service Level Agreement (SLA) compliance. Collected data can be analyzed using time-series forecasting [14], machine learning [15], or threshold-based methods [16], with dimensionality reduction or feature selection applied when indicators are correlated [17]. Although effective in revealing unanticipated aging patterns, these methods require extensive monitoring effort and often lack generalizability across systems. Hybrid strategies combine the strengths of both approaches by parameterizing analytical models with empirical data, thereby improving model fidelity and adaptability to real workloads [18,19]. Rejuvenation mechanisms operate at multiple layers of the virtualization stack. At the VM layer, techniques include cold-VM restarts and failover [16]. At the VMM layer, options include cold- and warm-VM rejuvenation, [20] suspend/resume or quick reboot [21], various VM migration [22] strategies (stop-and-copy, pre-copy, return-back, stay-on), and micro-reboots [23] of virtual infrastructure components. However, virtualization can introduce additional aging challenges—such as increased memory fragmentation—that accelerate degradation and complicate rejuvenation planning [24]. Given the diversity and operational trade-offs among these techniques and the nonexistence of mission-critical virtualized PAC systems in utility infrastructures it is quite a challenge to propose a standard model at this stage. This challenge is further exacerbated due to unavailability of field data, lack of standardization at VMM or hypervisor and application level, and technology choices at the server hardware and network layer level. This leaves the choice of model and analysis methodology selection completely open for VPAC-based DSAS. This work is thus deemed to be of an exploratory nature, an impetus to draw electrical power system researchers' attention to explore the modeling possibilities and further investigation in this emerging area. The work contributes by:

- Extending and incorporating an existing SAR model using a hierarchical modeling framework into the availability model of a DSAS.
- Analyzing and evaluating the reliability indices of the primary and secondary systems of the substation and later deriving the combined substation indices.

The paper is organized as follows: Section 2 presents the primary system and secondary architecture chosen for the study and proposed modeling methodology. Section 3 presents, analysis of primary and secondary systems along with combined system evaluation model. Reliability indices of primary, secondary and combined system are derived with discussion in Section 4. Finally, the paper concludes with Section 5.

2. System Modeling and Methodology

2.1. Primary and Secondary Systems for Study

The primary system of a digital substation considered in this study is a breaker-and-a-half substation configuration, widely regarded as one of the most dependable high-voltage substation layouts [25]. The selected topology is partitioned into four protection zones, two bus zones and two diameter zones, as illustrated in Figure 1. All protection and control functions associated with these zones are hosted on a centralized virtualized VPAC system. These protection zones are not fixed and may be adapted according to the underlying protection philosophy.

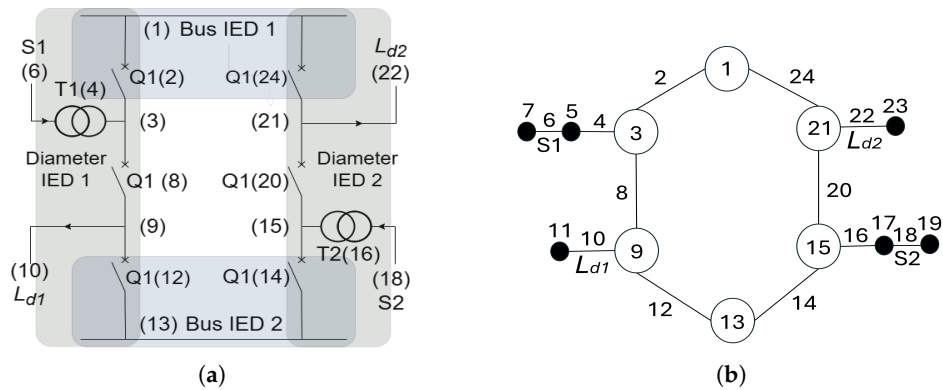


Figure 1. (a) CB-and-half scheme with protection zones and (b) corresponding graph [26].

The secondary system architecture comprises two redundant VPAC servers interconnected through the Parallel Redundancy Protocol (PRP), an established standard for high-availability industrial communication [27–29]. The architecture and corresponding Reliability Block Diagram (RBD) are presented in Figures 2 and 3, respectively. The RBD considers a single Power Supply (PS), a Time Source (TS), two VPAC servers, and four Process Interface Units (PIUs), with the Process Bus (PB) modeled as the PRP communication backbone. For the configuration in Figure 1, each VPAC server is responsible for complete substation protection, control, and monitoring functionality. The PIUs provide the interface to the primary equipment by acquiring current, voltage, and binary status signals, and by issuing control commands to circuit breakers and disconnectors. Both the PS and TS are required for the system to remain operational; the TS is essential for synchronized sampling, particularly for differential protection, and for maintaining consistent event timestamps across the substation. The PIUs are modeled as an n-out-of-n subsystem because certain protection functions and control actions, such as breaker failure protection or busbar fault clearing, require all PIUs to operate correctly to ensure coordinated switching, tripping, interlocking, and signaling at the PB. The VPAC servers form the core of both the PB and station bus (SB) communication networks, enabling full remote control and monitoring capabilities as expected in a fully digital substation.

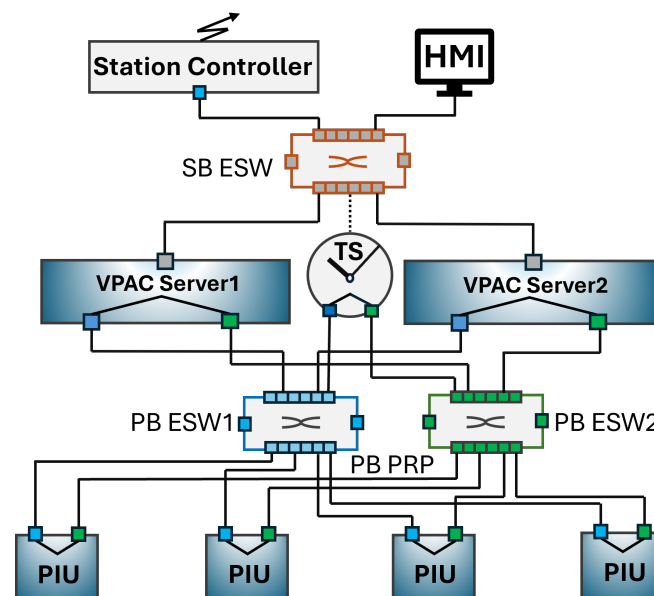


Figure 2. Redundant VPAC-based PRP architecture.

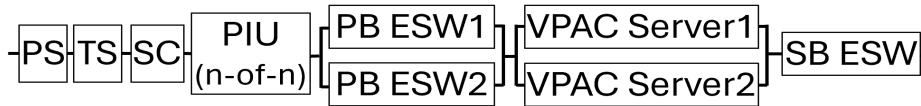


Figure 3. RBD of the VPAC-based redundant PRP architecture.

The availability of a system with n components in series is given by:

$$A_{ser} = \prod_{i=1}^n A_i \quad (1)$$

while the availability of two components in parallel is

$$A_{par} = A_1 + A_2 - A_1 \cdot A_2 \quad (2)$$

For a configuration requiring k out of n components to be operational, the availability is expressed as

$$A_{kofn}(k, n) = \sum_{i=k}^n \binom{n}{i} A^i \cdot A^{(n-i)} \quad (3)$$

The steady-state availability of a component having a failure λ and repair rate μ is defined as.

$$A_{ss} = \frac{\mu}{\lambda + \mu} \quad (4)$$

where typically, $\lambda \ll \mu$.

Based on the RBD in Figure 3 and using (1) the availability of the VPAC architecture can be written as

$$A_{Arch,VPAC} = A_{PS} \cdot A_{TS} \cdot A_{SC} \cdot A_{PIU} \cdot A_{VPAC} \cdot A_{PB} \cdot A_{SB} \quad (5)$$

The availability of the PRP-based PB can simply be obtained using (2), while the detailed evaluation of VPAC availability is presented in the following sections.

2.2. Proposed Methodology for Availability Modeling of VPAC-Based DSAS

2.2.1. Hierarchical Reliability Modeling of DSAS

Although RBDs offer an efficient means of computing reliability indices, they are limited in their ability to represent complex system states and the dynamic interactions between hardware and software components. More expressive formalisms, such as Markov chains, Petri nets or Bayesian networks, are often used for such purposes. In this work, we employ Markov chains to capture the dynamics of SAR, owing to their closed-form steady state solutions, memoryless property aligned with exponential failure and repair assumptions, and computational efficiency resulting from small, fixed-size transition matrices [30].

Model decomposition technique is used to manage system complexity and to integrate heterogeneous modeling approaches. Under this paradigm, a system is partitioned into subsystems whose outputs become inputs to higher-level models, forming a hierarchical structure that can be solved sequentially [30]. The VPAC-based DSAS adopts such a hierarchical modeling framework by embedding an SAR model within the RBD architecture shown in Figure 3. In this hierarchy, the availability of VPAC servers 1 and 2 is received through a lower-level SMP model (L1-SMP). The limiting up state probabilities represent steady-state availability, as illustrated in Figure 4. The resulting VPAC subsystem, comprising two servers in parallel, can then be evaluated using (2). This hierarchical approach preserves the analytical simplicity of RBDs while enabling the SMP layer to capture system interactions and SAR-related dynamics, thereby encapsulating model complexity without compromising computational tractability.

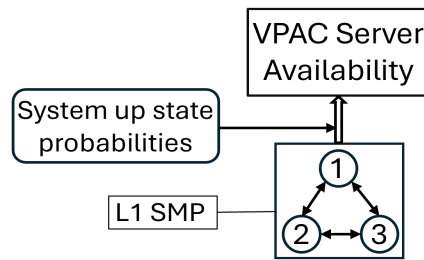


Figure 4. Hierarchical RBD Block with SMP.

2.2.2. Software Aging and Rejuvenation (SAR) Modeling

Software Rejuvenation Policy:

Software rejuvenation in a VPAC-based virtualized system may be carried out as either a partial or full rejuvenation procedure. Partial rejuvenation refers to refreshing a specific VM hosted on the hypervisor. This process is comparatively fast and can mitigate performance degradation of the targeted VM without affecting co-located VMs or applications; however, it does not fully restore the VM to its most robust operational state. Full rejuvenation involves restarting the hypervisor (Type-1), including the underlying operating system and VMM. This proactive maintenance action halts all applications, optionally preserves selected VM/VMM states, and reinitializes the hypervisor to release all accumulated resource burdens. If resource exhaustion occurs before any rejuvenation action can be initiated, the system transitions into a crash state. Recovery then requires either an automatic post-crash reboot or manual intervention, both of which return the system to a fully restored state. The aging and rejuvenation dynamics are modeled using an extended version of the two-level SMP proposed in [31], shown in Figure 5, with an additional eighth state (F). A summary of the state definitions is provided below:

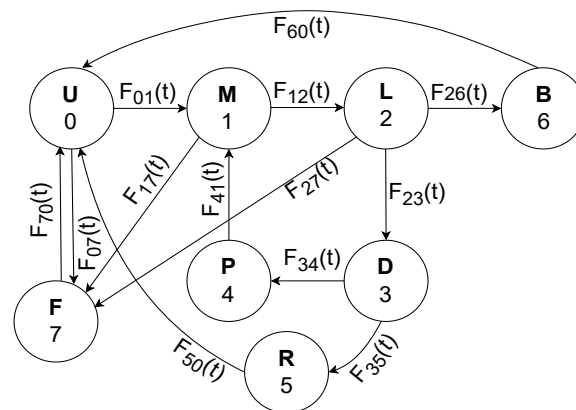


Figure 5. SMP model for software rejuvenation, including server hardware.

- *State U (UP)*: The most robust and fully available state. The system returns to this state after full rejuvenation, a crash recovery, or hardware repair.
- *State M (Medium-efficient)*: An available but moderately degraded state. The system enters this state from U due to performance decline or after completing partial rejuvenation.
- *State L (Low-efficient)*: The least robust available state. Severe degradation triggers an alert, prompting a rejuvenation decision before the system transitions to a crash.
- *State D (Decision)*: An instantaneous state where the system selects either partial (level-1) or full (level-2) rejuvenation. Its sojourn time is negligible.
- *State P (Partial rejuvenation)*: The system executes level-1 rejuvenation.
- *State R (Full rejuvenation)*: The system executes level-2 rejuvenation.
- *State B (Reboot)*: Represents the crash-recovery process. The system reaches this state when degradation becomes critical, requiring a reboot to restore state U .

- *State F (Server Failure)*: A hardware failure state that can be entered from any available state. All hardware-related failures of the server hosting the hypervisor, VPAC application, or other associated applications are aggregated here, whereas software aging dynamics are represented in the remaining states.

From a VPAC performance perspective, all states within the available set (U, M, L) are considered acceptable as long as the system continues to meet the most stringent protection performance requirements.

L1 SMP Modeling:

To derive the availability of a VPAC server from the underlying SMP, only the steady state solution of the SMP in Figure 5 is required. The analysis follows a two-step procedure in which the SMP is fully characterized by its kernel matrix $\mathbf{K}(t)$ [32,33]:

$$\mathbf{K}(t) = \begin{bmatrix} 0 & k_{01}(t) & 0 & 0 & 0 & 0 & 0 & k_{07}(t) \\ 0 & 0 & k_{12}(t) & 0 & 0 & 0 & 0 & k_{17}(t) \\ 0 & 0 & 0 & k_{23}(t) & 0 & 0 & k_{26}(t) & k_{27}(t) \\ 0 & 0 & 0 & 0 & k_{34}(t) & k_{35}(t) & 0 & 0 \\ 0 & k_{41}(t) & 0 & 0 & 0 & 0 & 0 & 0 \\ k_{50}(t) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ k_{60}(t) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ k_{70}(t) & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (6)$$

The kernel elements are defined as

$$\begin{aligned} k_{01}(t) &= \int_0^t \bar{F}_{07}(x) dF_{01}(x), & k_{07}(t) &= \int_0^t \bar{F}_{01}(x) dF_{07}(x), \\ k_{12}(t) &= \int_0^t \bar{F}_{17}(x) dF_{12}(x), & k_{17}(t) &= \int_0^t \bar{F}_{12}(x) dF_{17}(x), \\ k_{23}(t) &= \int_0^t \bar{F}_{26}(x) \cdot \bar{F}_{27}(x) dF_{23}(x), \\ k_{26}(t) &= \int_0^t \bar{F}_{23}(x) \cdot \bar{F}_{27}(x) dF_{26}(x), \\ k_{27}(t) &= \int_0^t \bar{F}_{23}(x) \cdot \bar{F}_{26}(x) dF_{27}(x), \\ k_{34}(t) &= \int_0^t \bar{F}_{35}(x) dF_{34}(x), & k_{35}(t) &= \int_0^t \bar{F}_{34}(x) dF_{35}(x), \\ k_{41}(t) &= F_{41}(t), & k_{41}(t) &= F_{41}(t), & k_{50}(t) &= F_{50}(t), \\ k_{60}(t) &= F_{60}(t), & k_{70}(t) &= F_{70}(t) \end{aligned}$$

Here, $F_{ij}(t)$ denotes the transition cumulative distribution function (CDF), while $\bar{F}_{ij}(t)$ is its complementary form. The allowable transition set is

$$\begin{aligned} \bar{F}_{ij}(t) &= 1 - F_{ij}(t), \quad \text{for } (i, j) \in \mathcal{E}, \\ \mathcal{E} &= \{(0, 1), (0, 7), (1, 2), (1, 7), (2, 3), (2, 6), (2, 7), \\ &\quad (3, 4), (3, 5), (4, 1), (5, 0), (6, 0), (7, 0)\}. \end{aligned}$$

Step 1. Embedded DTMC: The one-step transition probability matrix of the embedded Markov chain is:

$$\mathbf{P} = \mathbf{K}(\infty) = \begin{bmatrix} 0 & p_{01} & 0 & 0 & 0 & 0 & 0 & p_{07} \\ 0 & 0 & p_{12} & 0 & 0 & 0 & 0 & p_{17} \\ 0 & 0 & 0 & p_{23} & 0 & 0 & p_{26} & p_{27} \\ 0 & 0 & 0 & 0 & p_{34} & p_{35} & 0 & 0 \\ 0 & p_{41} & 0 & 0 & 0 & 0 & 0 & 0 \\ p_{50} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_{60} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ p_{70} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where $p_{ij} \triangleq k_{ij}(\infty)$.

The limiting state probabilities v of this embedded discrete time Markov chain (DTMC) are obtained from

$$v = v \cdot \mathbf{P} \quad \text{and} \quad v \cdot \mathbf{e} = 1 \quad (7)$$

Step 2. SMP Steady-State Probabilities: The SMP steady-state probabilities π_i are computed using the limiting state probabilities v_i and the mean sojourn times h_i :

$$\pi_i = \frac{v_i \cdot h_i}{\sum_j v_j \cdot h_j} \quad (8)$$

The mean sojourn times for the SMP in Figure 5 are

$$\begin{aligned} h_0 &= \int_0^\infty \bar{F}_{01}(t) \cdot \bar{F}_{07}(t) dt, & h_1 &= \int_0^\infty \bar{F}_{12}(t) \cdot \bar{F}_{17}(t) dt, \\ h_2 &= \int_0^\infty \bar{F}_{23}(t) \cdot \bar{F}_{26}(t) \cdot \bar{F}_{27}(t) dt, & h_3 &= \int_0^\infty \bar{F}_{34}(t) \cdot \bar{F}_{35}(t) dt, \\ h_4 &= \int_0^\infty \bar{F}_{41}(t) dt, & h_5 &= \int_0^\infty \bar{F}_{50}(t) dt, \\ h_6 &= \int_0^\infty \bar{F}_{60}(t) dt, & h_7 &= \int_0^\infty \bar{F}_{70}(t) dt \end{aligned}$$

Finally, VPAC availability is defined as the probability of the system residing in any of the three acceptable operational states (U, M, L), which correspond to states 0, 1, and 2:

$$A_{VPAC} = \pi_0 + \pi_1 + \pi_2 \quad (9)$$

This availability value is propagated to the each VPAC server blocks in the higher-level RBD of Figure 4, then A_{VPAC} can be found using (2), and substituted into (5), thereby incorporating the effects of software aging and rejuvenation into the overall architectural availability.

3. Primary and Secondary System Analysis

3.1. Secondary System Architectures Analysis

The modeling methodology described in Section 2 is applied to the secondary system architecture shown in Figure 2, with the corresponding reliability parameters summarized in Table 1.

Table 1. Failure rates and repair times of components [34].

Components	Failure rates - λ (f/yr)	MTTR + Logistics (h)
PS	9.13×10^{-3}	2+12
TS	3.65×10^{-2}	2+12
SC	5.48×10^{-3}	2+12
Server	1.83×10^{-2}	2+12
PIU	5.48×10^{-2}	2+12
Link	1.1×10^{-2}	2+12
ESW	7.03×10^{-2}	2+12

Choice of Model Parameters

The objective of this work is to demonstrate how the SAR model can be integrated into the DSAS architectural reliability analysis. Accordingly, in the absence of any field data from utility, the parameter selection is solely based on reasonable assumptions rather than site-specific measurements.

State 0 represents the most robust operating condition, from which the system initiates operation after deployment or a full refresh. As utility DSAS environments are highly constrained, where system access and software updates occur infrequently, an exponential distribution with an assumed mean time to failure (MTTF) of one year is assigned, i.e., $\lambda_{01} = 1 \text{ failures/year}$. This transition models the effect of updates or patches applied to the VPAC-VM, the VMM, or the host OS. The transition to state 1 reflects a moderate degradation phase and is assigned an increased failure rate $\lambda_{02} = 2 \text{ failures/year}$. State 2 captures the more pronounced degradation phase and is modeled using a Weibull distribution with increasing failure rate (IFR) parameters (4, 2), representing the accelerating impact of aging-related faults. State 3 is the decision state, in which the system is taken offline to select either level-1 or level-2 rejuvenation. This transition is modeled using a deterministic function $u(t - r)$, where r represents the scheduled rejuvenation time, consistent with maintenance policy or available downtime opportunities. Partial (level-1) and full (level-2) rejuvenation actions occur in states 4 and 5, respectively. These procedures can be executed remotely without physical site access, and are assigned fixed durations of $t_4 = 6 \text{ hours}$ and $t_5 = 10 \text{ hours}$. In contrast, the crash recovery state 6 may require dispatching maintenance personnel, including logistics time; thus, a fixed duration of $t_6 = 20 \text{ hours}$ is assumed.

Finally, state 7 models hardware failure of the server hosting the hypervisor, VPAC applications, or associated services. Hardware failure rates are typically provided by the manufacturer as MTTF values and are commonly represented using exponential distributions.

3.2. Primary System Analysis

Well-established analytical techniques exist for evaluating the reliability of primary substation busbar schemes. Among these, the minimal cut set method is widely applied and computationally efficient [35,36]. To automate this process, the procedure begins by assigning identifiers to all primary equipment in the bus scheme and constructing a Proposed Connection Matrix (PCM), derived from the topology illustrated in Figure 1. The PCM, together with the reliability parameters of the primary components, forms the complete input to the analysis program, which then automatically generates the minimal cut sets associated with loads L_{d1} and L_{d2} . Detailed descriptions of the minimal cut-set algorithm can be found in [37,38]. In this work, following cut sets up to second order are considered:

- First-Order Total Minimal Cut Sets (FOTMC)
- Second-Order Total Minimal Cut Sets (SOTMC)
- First-Order Active Minimal Cut Sets (FOAMC)
- Second-Order Minimal Cut Sets with Active + Total Failures (SOMCAT)
- Second-Order Minimal Cut Sets with Active + Active Failures (SOMCAA)
- First-Order Failure Events with Stuck Breaker (FOFES)
- Second-Order Failure Events with Stuck Breaker (SOFES)

These categories reflect different combinations of total failures, active failures, and switching-related events such as stuck breakers.

Analytical Method for Primary System

The analytical expressions used to compute the equivalent failure rate λ and repair rate r for each minimal cut set are summarized below. These expressions form the basis for the analytical results presented in Table 6.

1. FOTMC

$$\lambda = \lambda_1 \quad r = r_1 \quad (10)$$

2. SOTMC

$$\lambda = \lambda_1 \cdot \lambda_2 \cdot (r_1 + r_2) \quad r = \frac{r_1 \cdot r_2}{(r_1 + r_2)} \quad (11)$$

3. FOAMC

$$\lambda = \lambda_1^a \quad r = t_{sw1} \quad (12)$$

4. SOMCAT

$$\lambda = \lambda_1^a \cdot \lambda_2 \cdot (t_{sw1} + r_2) \quad r = \frac{t_{sw1} \cdot r_2}{(t_{sw1} + r_2)} \quad (13)$$

5. SOMCAA

$$\lambda = \lambda_1^a \cdot \lambda_2^a \cdot (t_{sw1} + t_{sw2}) \quad r = \frac{t_{sw1} \cdot t_{sw2}}{(t_{sw1} + t_{sw2})} \quad (14)$$

6. FOFES

$$\lambda = P_s \cdot \lambda_1^a \quad r = t_{sw1} \quad (15)$$

7. SOFES

$$\lambda = P_s \cdot \lambda_1^a \cdot \lambda_2 \cdot (t_{sw1} + r_2) \quad r = \frac{t_{sw1} \cdot r_2}{(t_{sw1} + r_2)} \quad (16)$$

Here, P_s denotes the probability of a stuck breaker; λ_1 , λ_2^a , λ_2 , and λ_2^a represent total and active failure rates of the involved components; and r_1 , r_2 , t_{sw1} , t_{sw2} denote their corresponding repair and switching times. Table 2 lists the reliability data for all primary components. A MATLAB implementation automated the entire process, generating minimal cut sets directly from the reliability dataset and the PCM matrix. Additional busbar configurations were also evaluated as test cases to validate the correctness and robustness of the developed tool.

Table 2. Primary Equipments Reliability Data.

Equipments	λ_a (f/yr)	λ_p (f/yr)	MTTR (h)	P_s
Line	0.1	–	4	–
Circuit Breakers	0.01	0.01	12	0.06
Transformers	0.1	–	50	–
Bus Sections	0.01	–	4	–

3.3. Evaluation of Combined Reliability Indices

Finally, the reliability contributions of the primary and secondary system models are integrated using the following relations. The overall system failure rate is obtained by summing the failure rates of all relevant minimal cut sets:

$$\lambda = \sum_{i=1}^k \lambda^k \quad (17)$$

The system unavailability U is determined by combining the repair characteristics of the first and second order minimal cut sets with the availability of the secondary system architecture:

$$U = r^1 + r^2 + \sum_{i=3}^k (U_{ds} \cdot r^k + A_{ds} \cdot t_{sw,ds}) \quad (18)$$

where λ^k and r^k correspond to the failure and repair parameters of the minimal cut sets defined in 10-16. Here, U_{ds} and A_{ds} denote the unavailability and availability of the DSAS, respectively. Additionally, r^1 and r^2 represent the repair times associated with the FOTMC and SOTMC groups defined in (10) and (11), while $t_{sw,ds}$ represents the switching time attributable to the DSAS. The overall outage duration is then computed directly using:

$$r = \frac{U}{\lambda} \quad (19)$$

which yields the complete reliability indices of integrated system.

4. Case Study for Combined System Reliability Indices

The reliability indices model for the individual primary, secondary and combined systems were presented in Section 3. The SAR models introduced in Section 2.2.2 enable the computation of the health and reliability attributes of the VPAC-based server through different software and hardware states. Correspondingly, the primary system failure modes are represented through the minimal cut sets summarized in Table 3.

Table 3. Minimal cut sets of L_{d1} based on Figure 1.

Cut set {ID}	Minimal cut set
FOTMC {1}	9, 10
SOTMC {2}	3+12, 3+13, 3+14, 3+15, 3+16, 3+18, 3+26, 4+15, 4+16, 4+18, 6+15, 6+16, 6+18, 8+12, 8+13, 8+14, 8+15
FOAMC {3}	8A, 12A
SOMCAT {4}	2A+12, 2A+13, 2A+14, 2A+15, 2A+16, 2A+18, 4A+12, 4A+13, 4A+14, 6A+12, 6A+13, 6A+14, 14A+6, 14A+6, 16A+8, 18A+8, 20A+3, 20A+4, 20A+6, 20A+8
SOMCAA {5}	2A+20A
FOFES {6}	2A+8S, 3A+8S, 4A+8S, 6A+8S, 13A+12S, 14A+12S
SOFES {7}	20S+21A+3, 20S+21A+4, 20S+21A+6, 20S+21A+8, 20S+22A+3, 20S+22A+4, 20S+22A+6, 20S+22A+8, 20S+24A+3, 20S+24A+4, 20S+24A+6, 20S+24A+8, 2S+1A+12, 2S+1A+13, 2S+1A+14, 2S+1A+15, 2S+1A+16, 2S+1A+18, 2S+24A+12, 2S+24A+13, 2S+24A+14, 2S+24A+15, 2S+24A+16, 2S+24A+18

All minimal cut sets obtained for load L_{d1} are presented in Table 3. Suffixes "A" and "S" denote conditions involving active failures and stuck breakers, respectively. To evaluate the overall system reliability, the primary and secondary models are integrated to determine the combined reliability indices for each load point. Either of the two load points. (L_{d1} or L_{d2}) may be considered, as the methodology applies identically to both. The combined analysis focuses on quantifying:

- **The failure rate of each load point**, reflecting the contribution of both primary equipment failures and secondary system outages; and
- **The interruption duration**, determined by the combined repair and switching characteristics of the two subsystems.

An analytical framework is employed to evaluate these combined indices, enabling systematic incorporation of the failure modes and architectural characteristics of both the primary and secondary systems depicted in Figure 2.

4.1. Primary System Reliability Indices

The reliability indices associated with each minimal cut set, along with their aggregated contributions, are presented in Table 4. As expected, the results indicate that first-order cut sets dominate both the overall failure rate and the annual downtime of the primary system. These cut sets represent single-component failures that have a direct and immediate impact on load-point reliability, and thus constitute the primary contributors to system risk.

Table 4. Reliability Indices of Breaker-and-half Scheme.

Minimal Cut set	Failure rate (f/yr)	Outage Duration (hr/f)	Annual Downtime (hr/yr)
FOTMC	0.11	16.00	1.76
SOTMC	3.90E-04	18.78	7.32E-03
FOAMC	0.02	1.00	0.0200
SOMCAT	6.78E-05	0.96	6.54E-05
SOMCAA	2.28E-08	0.50	1.14E-08
FOFES	0.0144	1.00	0.0144
SOFES	8.41E-06	0.97	8.18E-06
TOTAL	0.1449	12.44	1.80

In the second stage of the analysis, the RBD models of the secondary system architecture shown in Figure 3 were evaluated using the equations introduced in Section 2. The architectural logic and corresponding availability expressions were implemented in MATLAB to compute the reliability indices of DSAS.

4.2. Secondary System Reliability Indices

The reliability indices for the VPAC-based DSAS architecture in Figure 2, computed with and without incorporating the SAR model, are summarized in Table 5. The baseline case, which excludes SAR, considers only the hardware failure rate of the VPAC server irrespective of the failure modes associated with the software executed on the platform, reflecting the conventional assumption. Under the selected model parameters, this omission does not significantly influence the overall DSAS availability. However, when the SAR model is integrated, both the failure rate and annual downtime increase, capturing the impact of software aging processes.

Table 5. Reliability Indices of VPAC-based DSAS.

Architecture	Failure rate (f/yr)	Outage Duration (hr/f)	Annual Downtime (hr/yr)	Availability
VPAC without SAR	0.34063	13.995	4.7672	0.99946
VPAC with SAR	0.34161	13.970	4.7723	0.99946

The resulting availability of the VPAC server, computed using (9) from the steady-state solution of the SMP in Figure 5, is illustrated in the 3D surface plot in Figure 6. The rejuvenation interval r is expressed in years, and the maximum availability occurs at $r = 0.08 yr$ and $p = 0$, corresponding to a full rejuvenation strategy. This maximum or optimum availability value, for given parameters, can be obtained analytically or numerically, as explained in [31]. The values of indices for VPAC-based SAR architecture in Table 5 is listed for the maximum availability value occurred in Figure 6.

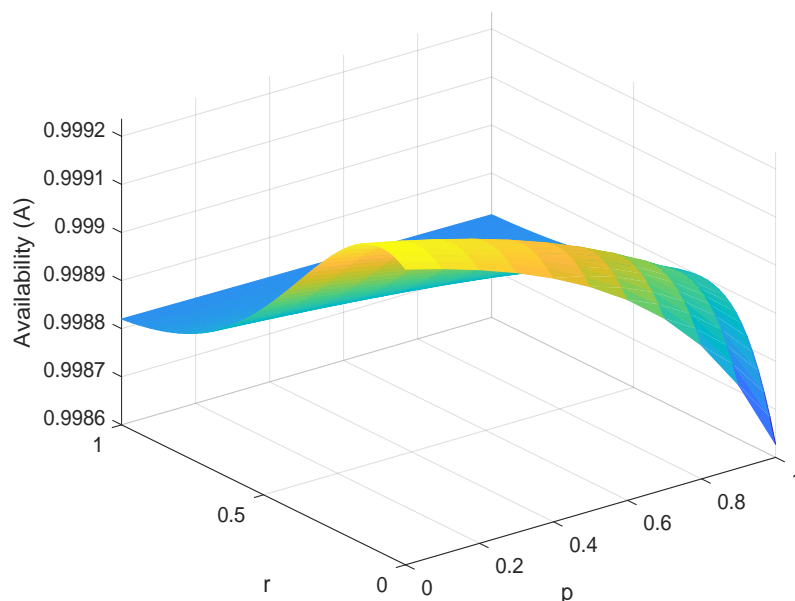


Figure 6. 3D plot of availability of SAR model for VPAC server.

These secondary system indices can then be combined with the primary system results from Table 4 using (17), (18), and (19), from Section 3.3 to derive the final reliability indices for each load point.

4.3. Combined System Reliability Indices

For the combine system analysis, the analytical method introduced in Section 3.3 is adopted. Two key observations arising from the results presented in Table 6 are now examined and discussed in detail.

Table 6. Reliability Indices of CB-and-half Scheme with and without SAR Model.

Architecture	Failure rate (f/yr)	Outage Duration (hr/f)	Annual Downtime (hr/yr)
CB-and-half without VPAC-SAR	0.14487	4.9111	0.71145
CB-and-half with VPAC-SAR	0.14487	4.9111	0.71145

4.3.1. Decrease in annual downtime achieved through DSAS

The indices in Table 4 were obtained assuming a manual switching time of $t_{sw,m} = 1 h$ for restoring supply to load point L_{d1} in the absence of a DSAS. When a DSAS is available, this time is reduced to $t_{sw,ds} = 0.1 h$ in Table 6, owing to automatic, pre-defined switching sequences. However, during periods when the DSAS itself is unavailable, the switching time reverts to the manual value of 1 h. Additionally, it is assumed that approximately two-thirds of feeder faults are transient in nature [39], enabling rapid isolation and restoration through reclosing operations or protection actions. The capability of the DSAS to automatically isolate feeder faults and to initiate swift switching actions significantly decreases both annual downtime and outage duration in the combined system results in Table 6 relative to the aggregated primary-system indices in Table 4. This demonstrates a clear operational advantage offered by the DSAS architecture.

4.3.2. Influence of DSAS Failure Modes

A comparison of the results in Table 6 with those in Table 4 shows that the overall failure rate remains unchanged. Moreover, no difference exists between combined indices with or without SAR architectures, see Table 6. This outcome follows directly from the evaluation methodology, which is based primarily on the failure modes of the primary system. Under the assumptions of this study, a

failure within the secondary system does not directly interrupt supply to the load point, and therefore does not alter the failure rate of the combined system which might be unrealistic.

Table 8 investigates this assumption further, where the “base case” refers to the architecture results obtained using the analytical method summarized in Table 6. Table 7 demonstrates how specific failure modes within a DSAS architecture can influence the reliability indices of load L_{d1} . For example, failure of the PS of a critical device such as a VPAC server or a PIU responsible for the protection and control of L_{d1} may force the primary system into a fail-safe state until the faulty component is repaired. In such scenarios, the secondary system failure effectively propagates to the load point, contributing to its interruption duration.

Table 7. Impact of incorporating various failure modes into the VPAC architecture on L_{d1} , in comparison with the results presented in Table 6.

Failure Mode	Failure rate (f/yr)	Outage Duration (hr/f)	Annual Downtime (hr/yr)
(Base case)	0.14487	4.9111	0.71145
PS	0.1540	5.4501	0.8393
SC	0.1503	5.2425	0.7882
PIU	0.1997	7.4057	1.4787
SB	0.2149	7.8722	1.6915

Table 8 introduces some other hypothetical failure modes pertinent to a VPAC server and shows how they influence the primary-side downtime. For clarity, only one failure mode is incorporated per row to highlight its isolated impact. In practice, however, multiple such failure modes may exist within an a system, and their combined effects must be considered to accurately assess overall system reliability. This observation underscores the importance of collecting and analyzing historical reliability data for VPAC-based DSAS architectures. As these systems are almost non-existent in utility environments; therefore, empirical data on secondary system failure behavior is unavailable, yet such information is essential for comprehensive reliability modeling and fully capturing the interdependence between primary and secondary system failure modes.

Table 8. Incorporation of hypothetical failure modes [40] into the VPAC architecture and their influence on L_{d1} , in comparison with the results presented in Table 6.

Failure Modes	Failure rate (f/yr)	Outage Duration (hr/f)	Annual Downtime (hr/yr)
Code	0.160	5.76	0.921
External	0.180	6.68	1.201
Hidden	0.148	5.10	0.753
Interoperability	0.220	8.01	1.761
Configuration	0.230	8.27	1.901
Commissioning	0.165	6.01	0.991
Compatibility	0.149	5.18	0.774
Memory leak	0.235	8.39	1.971
VMM (Hypervisor) crash/hang	0.210	7.73	1.621
VM (VPAC App) crash/hang	0.170	6.25	1.061

5. Conclusions and Suggestion

This paper presented a hierarchical modeling framework for integrating software aging dynamics into the RBD representation of a VPAC-based DSAS. The primary aim of this work is to stimulate early research into the selection of software aging models, identification of aging indicators, and deployment of appropriate instrumentation within the VMM/hypervisor and potential VPAC applications in utility grid to enable systematic data collection in laboratory environments or pilot installations. Using the selected parameters to populate the SMP model, the developed SAR model reliability indices

showed an increase in failure rate and annual downtime as compared to the model without SAR i.e. to only account for server hardware in the model. Combined substation reliability indices with both these models did not show any influence on substation indices except reduction in downtime due to improvement in switching time as a result of DSAS employment. However, subsequent analysis revealed that this can be attributed to the unavailability of historical failure modes data, and availability of such data can impact the overall substation indices more significantly.

The literature emphasizes the importance of adopting a hybrid approach for software aging prediction. Implementing such an approach requires identifying meaningful aging indicators or system-level metrics capable of revealing degradation trends. In newly deployed systems, the number of potential indicators may be large, necessitating the use of classification, feature selection, or dimensionality reduction techniques to isolate the most informative metrics. These selected indicators can then be analyzed using statistical or machine learning methods to detect early signs of aging. Once the relevant metrics are identified, a suite of established models from the SAR literature may be used to estimate optimal rejuvenation intervals, validate model accuracy, and enhance predictive performance.

Author Contributions: This framework was conceptualized and proposed by R.R.S. The paper is written by R.R.S. H.K.H. supervised the work and edited the manuscript.

Funding: This work was funded by the ProDig – Power System Protection and Control in Digital Substations project, supported by the Research Council of Norway under Project No. 295034.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest. The funding body had no role in the design of the study, data collection, analysis, interpretation of data, or in writing the manuscript.

Abbreviations

The following abbreviations are used in this manuscript:

λ	Failure rate
μ	Repair rate
A	Availability;
ESW	Ethernet Switch
IED	Integrated Electronic Device
PAC	Protection Automation and Control
PB	Process Bus
PRP	Parallel Redundancy Protocol
PS	Power Supply; PIU: Process Interface Unit
RBD	Reliability Block Diagram
SAR	Software Aging and Rejuvenation
SB	Station Bus
SC	Station Controller
TS	Time Source
U	Unavailability

References

1. IEC. Communication Networks and Systems for Power Utility Automation—Part 7-1: Basic Communication Structure—Principles and Models. *IEC Standard 61850-7-1* **2011**.
2. Valtari, J. Centralized Architecture of the Electricity Distribution Substation Automation: Benefits and Possibilities. *Tampere University of Technology, Finland* **2013**.

3. Kabbara, N. Virtualized Digital Substations: Exploring the Design, Simulation, and Validation of the Next-Generation Power System Backbone. *Utrecht University* **2025**.
4. Valtari, J.; Kulmala, A.; Schönborn, S.; Kozhaya, D.; Birke, R.; Reikko, J. Real-Life Pilot of Virtual Protection and Control: Experiences and Performance Analysis. *CIREC 2023, Rome, Italy* **2023**.
5. Avritzer, A.; Weyuker, E.J. The role of modeling in the performance testing of e-commerce applications. *IEEE Trans. Softw. Eng.* **2004**, *30*, 1072–1083.
6. Avizienis, A.; Laprie, J.; Randell, B.; Landwehr, C. Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. Dependable Secure Comput.* **1993**, *1*.
7. Huang, Y.; Kintala, C.; Kolettis, N.; Fulton, N.D. Software rejuvenation: analysis, model and applications. *FTCS* **1995**, 381–390.
8. Dohi, T.; Avritzer, A.; Trivedi, K. Handbook of Software Aging and Rejuvenation: Fundamentals, Methods, Applications, and Future Directions. *World Scientific* **2020**.
9. Changa, X.; Zhang, Z.; Li, X.; Trivedi, K.S. Model-Based Survivability Analysis of a Virtualized System. *IEEE LCN* **2016**, 611–614.
10. Machida, F.; Nicola, V.F.; Trivedi, K.S. Job completion time on a virtualized server with software rejuvenation. *ACM* **2014**, *10*.
11. Bai, J.; Chang, X.; Ning, G.; Zhang, Z.; Trivedi, K.S. Service Availability Analysis in a Virtualized System: A Markov Regenerative Model Approach. *IEEE Trans. Cloud Comput.* **2022**, *10*, 2118–2130.
12. Melo, M.; Araujo, J.; Matos, R.; Menezes, J.; Maciel, P. Comparative Analysis of Migration-Based Rejuvenation Schedules on Cloud Availability. *IEEE SMC* **2013**, 4110–4115.
13. Xu, J.; Li, X.; Zhong, Y.; Zhang, H. Availability Modeling and Analysis of a Single-Server Virtualized System with Rejuvenation. *J. Softw.* **2014**, *9*.
14. Araujo, J.; Matos, R.; Maciel, P.; Matias, R. Software aging issues on the eucalyptus cloud computing infrastructure. *IEEE SMC* **2011**, 1411–1416.
15. Alonso, J.; Belanche, L.; Avresky, D.R. Predicting Software Anomalies Using Machine Learning Techniques. *IEEE NCA* **2011**, 163–170.
16. Silva, L.M.; Alonso, J.; Silva, P.; Torres, J.; Andrzejak, A. Using Virtualization to Improve Software Rejuvenation. *IEEE NCA* **2007**, 33–44.
17. DeCelles, S.; Kandasamy, N. Entropy-Based Detection of Incipient Faults in Software Systems. *IEEE PRDC* **2012**, 70–79.
18. Vaidyanathan, K.; Trivedi, K.S. A comprehensive model for software rejuvenation. *IEEE Trans. Dependable Secure Comput.* **2005**, *2*, 124–137.
19. Liu, Y.; Liu, W.; Song, J.; He, H. An empirical study on implementing highly reliable stream computing systems with private cloud. *Ad Hoc Netw.* **2015**, *35*.
20. Machida, F.; Kim, D.S.; Trivedi, K.S. Modeling and analysis of software rejuvenation in a server virtualized system with live VM migration. *Elsevier* **2013**, *70*.
21. Kourai, K.; Chiba, S. Fast Software Rejuvenation of Virtual Machine Monitors. *IEEE Trans. Dependable Secure Comput.* **2011**, *8*, 839–851.
22. Machida, F.; Kim, D.S.; Trivedi, K.S. Modeling and analysis of software rejuvenation in a server virtualized system. *IEEE WoSAR* **2010**, 1–6.
23. Le, M.; Tamir, Y. Applying Microreboot to System Software. *IEEE SERE* **2012**, 11–20.
24. Alonso, J.; Matias, R.; Vicente, E.; Maria, A.; Trivedi, K.S. A comparative experimental study of software rejuvenation overhead. *Perform. Eval.* **2013**, *70*, 231–250.
25. Li, D.D.; Wu, X.Y.; Deng, H.Z. Reliability Evaluation in Substations Considering Operating Conditions and Failure Modes. *IEEE Trans. Power Deliv.* **2012**, *27*, 309–316.
26. Syed, R.R.; Høidalen, H.K. Investigating the Impact of Fault Handling Models on Reliability Indices of Digital Substation. *IEEE Trans. Power Deliv.* **2025**, 1–13.
27. IEC. Communication Networks and Systems for Power Utility Automation—Part 90-4: Network Engineering Guidelines. *IEC Standard 61850-90-4* **2020**.
28. IEC. Industrial Communication Networks—High Availability Automation Networks: General Concepts and Calculation Methods. *IEC 62439-1* **2017**.
29. IEC. Industrial Communication Networks—High Availability Automation Networks: PRP and HSR. *IEC 62439-3* **2017**.
30. Trivedi, K.S.; Bobbio, A. Reliability and Availability Engineering: Modeling, Analysis, and Applications. *Cambridge University Press* **2017**.

31. Xie, W.; Hong, Y.; Trivedi, K. Analysis of a two-level software rejuvenation policy. *Reliab. Eng. Syst. Saf.* **2005**, *87*, 13–22.
32. Sahner, R.A.; Trivedi, K.S.; Puliafito, A. Performance and Reliability Analysis of Computer Systems. *Springer* **1996**.
33. Limnios, N.; Oprışan, G. Semi-Markov Processes and Reliability. *Birkhäuser* **2012**.
34. Scheer, G.W. Answering Substation Automation Questions Through Fault Tree Analysis. *SEL* **1998**.
35. Billinton, R.; Allan, R.N. Reliability Evaluation of Engineering Systems. *Springer* **1992**.
36. Allan, R.N.; Billinton, R.; De Oliveira, M.F. An Efficient Algorithm for Deducing Minimal Cuts and Reliability Indices. *IEEE Trans. Reliab.* **1976**, *25*, 226–233.
37. Grover, M.S.; Billinton, R. A Computerized Approach to Substation Reliability Evaluation. *IEEE Trans. Power Appar. Syst.* **1974**, *93*, 1488–1497.
38. Hajian-Hoseinabadi, H. Computerized Algorithm for Deducing Minimal Cut Sets. *Int. Trans. Electr. Energy Syst.* **2014**.
39. Fundamental SQSS Review: Planning and Operational Contingency Criteria. *Working Group 4 Report* **2010**.
40. Falahati, B.; Chua, E. Failure Modes in IEC61850-enabled Substation Automation Systems. *IEEE PES T&D* **2016**.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.