Hindawi
Complexity
Volume 2023, Article ID 6690805, 12 pages
https://doi.org/10.1155/2023/6690805

*Research Article*

# Fault Detection and Isolation in Smart-Grid Networks of Intelligent Power Routers Modeled as Probabilistic Boolean Networks

**Pedro J. Rivera Torres**,[1,2,3] **Carlos Gershenson García**,[2,4] **and Samir Kanaan Izquierdo**[3]

[1]*Distributed Information and Automation Laboratory, Institute for Manufacturing, Department of Engineering, University of Cambridge, Cambridge CB3 0FS, UK*
[2]*Complexity Sciences Center, National Autonomous University of México, Circuito Centro Cultural S/N, Ciudad Universitaria, Coyoacán 04510, Ciudad de México, Mexico*
[3]*Bioinformatics and Biomedical Signals Laboratory, Biomedical Engineering Research Center, School of Industrial Engineering, Universitat Politècnica de Catalunya, H Building, 4th Floor, Av. Diagonal, 647, Barcelona 08028, Catalunya, Spain*
[4]*Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, USA*

Correspondence should be addressed to Pedro J. Rivera Torres; pjr62@cam.ac.uk

Received 15 April 2023; Revised 9 May 2023; Accepted 1 December 2023; Published 14 December 2023

Academic Editor: Jesus M. Munoz-Pacheco



A self-organizing complex-network modeling method, probabilistic Boolean networks, is presented as a model-based diagnostic system for detecting and isolating different types of faults, failures, and modes of operation in which a network of intelligent power routers is deployed over a standard power test case: the Western System Coordinating Council 9 Bus System. Such a system allows designers and engineering professionals to make educated decisions pertaining to the design of smart-grid systems endowed with intelligent power routers. There is a recurrent necessity to design reliable and fault-tolerant smart power systems, maintaining adequate operation and adherence to performance specifications, while keeping costs at the minimum. This diagnostics system will help achieve such goals: better design through thorough analysis of the conditions that lead to a fault on a smart grid, proper detection of these faults, and isolation of the respective assets.

## 1. Introduction

Complex-adaptive systems (CAS) are systems that are composed of different smaller elements that are interconnected [1], because the etymology of the word complex derives from the Latin word, *complexus*, past participle of *complecti*, which means interwoven, threaded, or interconnected. Therefore, a complex system cannot be separated or fragmented because the interactions between the components give rise to new or emergent information or behavior. Individual parts have their specific functions, and emerging from these interactions is what makes the difference.

One of these mechanisms, used widely in bioinformatics and biomedical engineering, is Kauffman N-K or Boolean networks (BN). A Kauffman BN is a discrete collection of Boolean variables represented as nodes on the network, where each of these variables has a Boolean function associated with it. These networks have discrete time and states and were postulated by Dr. Stuart Kauffman in the late 1960s with the purpose of modeling gene regulatory networks [2]. There are several variations from the N-K to random Boolean networks, autonomous Boolean networks, deterministic asynchronous random Boolean networks, etc. One of their main features or characteristics is their dynamics and the fact that they will sooner or later fall into a steady or cyclical state, called attractor. These represent the behavior of the network in the long run, or its steady state. They can be a single repeating state or a set of states that repeat. Since these networks are deterministic, there is only one Boolean function per node, so they cannot model probabilistic behavior. There are extensions to the BN, such as Shumlevich and Dougherty's Probabilistic Boolean Networks (PBN) [3] or Dorigo's much earlier interpretation of the same concept [4]. These allow modeling the probabilistic behavior that BN's are incapable of. In this study, we utilize Shmulevich's model. These networks are a probabilistic extension of the Kauffman networks, where there is also a collection of nodes/variables, but instead of possessing only one Boolean Function per variable, there can be 1 or more, that are selected with a probabilistic weight. Therefore, a PBN can be thought of as a tree of several constituent Kauffman networks that are chosen probabilistically. A complete description of these PBNs is available in [3]. We have been studying the use of PBNs outside the gene regulatory network modeling, its main field of application, and have been using these networks to model other phenomena, such as industrial machines [5], reliability-driven process design [6], preventive maintenance [7], and intelligent power routers (IPR) [8]. This study continues the application of PBN modeling of smart-grid systems, taking advantage of the characteristics and advantages that these complex networks possess to model a power network endowed with intelligent power routers [9] in a standard test system, the Western Systems Coordinating Council (WSCC) 9 Bus model. We will present how a PBN smart-grid model can be used for fault detection and isolation (FDI) [10], a scheme where the main idea is to efficaciously ascertain faults and scrupulously isolate them from a failed or defective asset or component in the quickest time possible. This enables a significant reduction of diagnosis time or the general downtime of the system to upsurge its availability, reducing the downtime of a system with complicated maintenance and vital importance to modern society.

## 2. Materials and Methods

PBNs have been used for reinforcement learning on an intelligent power router model [8], and to perform fault detection and isolation on an IPR [11]. In this last study, a model of the IPR was done with a PBN, which enabled it to detect faults in the device's components and isolate the faulty asset. The stochastic nature of the PBN model allowed for a prognosis on when a particular asset would present a fault.

We have established a PBN model of the WSCC 9-Bus System, a test case representing a system with nine buses and three generation sources, serving three loads, as shown in Figure 1.

In this diagram, an approximation of an equivalent power system for the purposes of testing (easy to control), there are 9 buses, 3 two-winding power transformers, 3 generators (163 MW, 334 MW, and 85 MW), 6 lines, and 3 loads (166 MW, 100 MW, and 165 MW), with base kV levels of 13.8, 16.5, 18, and 230 kV. The FDI method described in [10] was adapted in this study, in which models are established for the normal operation of the system and for each of the faults it exhibits, as shown in Figure 2.

This is done by using the self-organizing characteristics that PBN modeling provides, where the steady-state tenue of the system is achieved through attractor states, and these relate to the fault conditions and the operating modes of the modeled device or system. In this case, an IPR is placed on each of the buses to intelligently manage the network, and a PBN model of each of the IPRs is developed, along with models for the other components of the system, such as the system's loads. The model was constructed using the PRISM Model Checker [13].

The way this model is constructed and its semantics are very similar to the one presented in [8]. The failure modes and operation of the model is characterized and through model checking property verification in PRISM permits the analysis of the state of the components of the test case system and the relevancy of which assets correlate to the different faults. Each of the assets is a PBN, and the overall test system is also a PBN. The PBN that describes the IPRs is presented in [8] along with the IPR components, their failure mode classification and state, the PBN's Boolean predictor functions, and the selection probability for each. The PBN modeling method presented here and in previous studies is flexible and scalable. A description of this method is shown in Figure 3.

The normal operation and failure modes presented herein are based on the reliability assessment of the IPR device performed and the failure modes expressed in [8, 9], and they are based on each component's mean time between failures (MTBF) and based on a design failure mode and effects analysis (DFMA) performed on the device. Thus, the model can detect single and multiple faults for the entire test system and for each IPR involved. This model is relatively large in terms of the number of states it can assume, which is $7.4\times10^{23}$ states, $5.9\times10^{25}$ transitions, and $4.62\times10^{25}$ choices, with a reachability of 279 nodes. It has a transition matrix of 60,533 nodes and 10 terminal nodes. It takes 13.37 minutes to build this model in PRISM.

The different experiments that are presented in this study were conducted in PRISM using Property Verification in Probabilistic Computational Tree Logic (PCTL) [13]. This allows the determination of the maximum probability of occurrence of the different states of the failure modes in which the different assets in the test system can be. We are able to perform in this manner detection, diagnosis, and isolation of assets in failure or fault conditions.

To perform FDI, there needs to be a model for every fault condition and for the system's normal operation. In our case, the model for the normal operation of the device is shown in Figure 4.

The normal operation module contains the main IPR components and the failure probability of each of them based on the IPR's specifications. For the IPR, a probabilistic Boolean network model was established using PRISM, as shown in Figure 5.

The IPR PBN module contains the IPR's main components (nodes) and the predictor functions. We also classify each of the states in which the IPR can be, so we can determine the failure

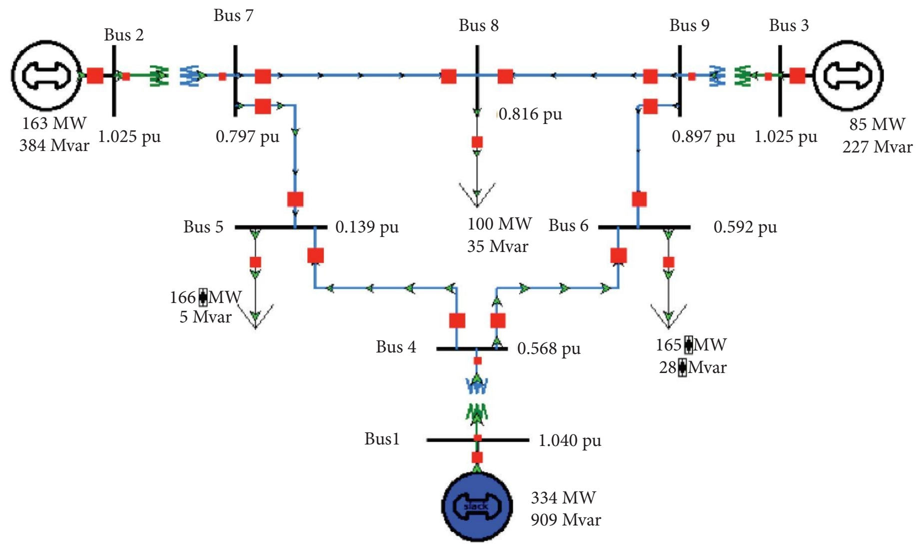


Figure 1: Western system coordinating council 9-bus model [12].

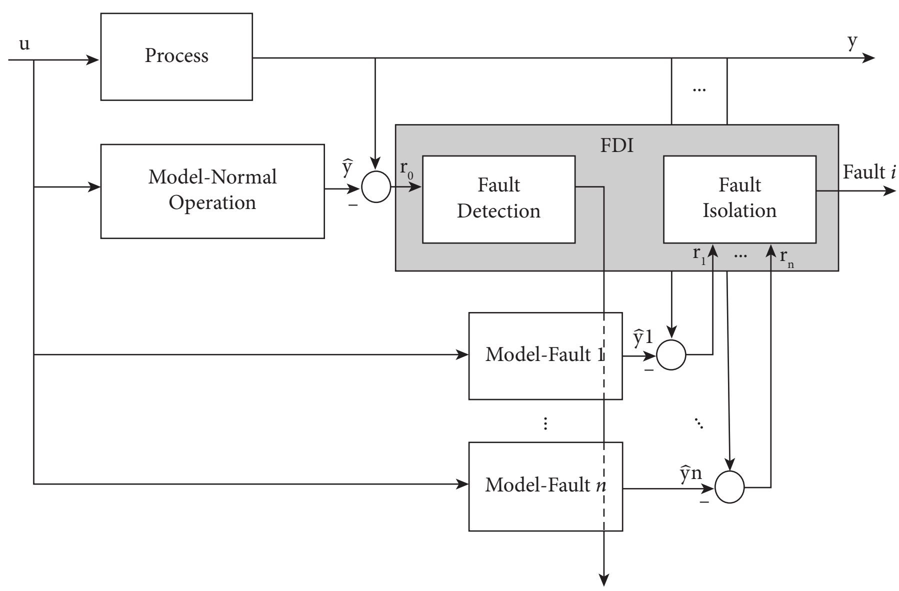


Figure 2: Fault detection and isolation methodology [10].

modes of the device. Based on the state of each of the components, after the application of the predictors, the state of the IPR is determined. The model for each fault condition, which is based on the IPR's description and reliability analysis, is handled in PRISM as shown in Figure 6.

Recalling the IPR's reliability analysis from [11], the device can be in 12 different states, and these states can be reduced to 4 classifications: two types of faults, a catastrophic failure, and the normal operation. Based on the condition of the IPR's components, per the analysis in [9], we classify the IPR's states into these categories, therefore representing the operation modes.

## 3. Results and Discussion

Quantitative validation of the model and property verification experiments were performed with PRISM. The system is a collection of IPRs interconnected like the WSCC 9-bus model, and each IPR is modeled as a 4-node PBN as per [8, 11]. The experiments that were conducted have a PBN representing each IPR, and its main components, the breakers, software, and router, are considered the model's nodes. The state of each of the components combined represents the overall state of each of the IPRs, which is represented in Table 1.

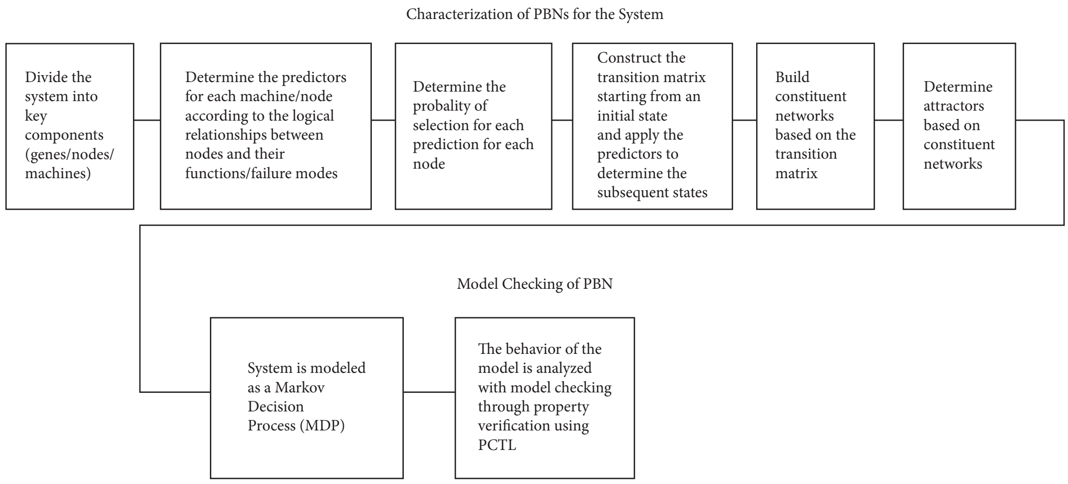


Figure 3: Method for modeling assets and/or systems as PBNs [5].

```
module IPRNormpOp

router : bool init false;
software : bool init false;
breakera : bool init false;
breakerb : bool init false;


//MTBF
[] true -> 0.90009:(router' = false) + 0.09991:(router' = true);
[] true -> 0.99:(software' = false) + 0.01:(software' = true);
[] true -> 0.99330:(breakera' = false) + 0.0067:(breakera' = true);
[] true -> 0.99330:(breakerb' = false) + 0.0067:(breakerb' = true);

endmodule
```

Figure 4: Model for the IPR's normal operation.

The relationships between the components are captured in the Boolean Predictors Functions, which are shown on Table 2.

These predictors are based on the physical relationships between components and the IPR's reliability analysis [8, 9, 11]. In all of our experiments and the figures that represent them, time is expressed in hours ($h$). We performed property verification and maximum probability of occurrence experiments on the operational modes to verify the model's fitness to emulate the device dynamics. For FDI, a model of each of the fault conditions is created, along with a model of the normal operation of the device. Figure 4 presents the model of the normal operation of the device.

Detection and isolation: the model can accurately detect those IPR categories that represent faults or failure, as those presented in Table 1. From an initial state, the PBN can select an appropriate context or sub-BN, and given the dynamics of these complex networks, it will select one of the attractor states of the constituent BNs. The attractors in these contexts represent the steady states and one of the categories in which the devices will be. These methods have been previously used to assess faults and failures in manufacturing processes and machines [5–8, 11]. As discussed in the previous section, the IPR's PBN model in PRISM contains the fault conditions and operating modes that were identified in [11] along with the predictor functions, and after their application, the state of the IPR is evaluated and a determination of the test system as a whole is made (an evaluation of the state of all IPRs on the network), and we are able to detect all failure modes and isolate the components causing them. This permits detection and isolation of an IPR's component faults, faults in individual IPRs, multiple component faults in an IPR, component fault in multiple IPRs, and multiple IPR faults. As a first test, we present the maximum probability of occurrence of the failure of any IPR within the test system, as shown in Figure 7.

```
module IPR1

	//IPR1 components
	router1 : bool init false; // Router
	software1 : bool init false; // Software
	breaker1a : bool init false; // Breaker 1
	breaker1b : bool init false; // Breaker 2

	//States
	ipr1_State : [0..11] init 0; // Initially all systems work properly

	//Classes
	class1 : [0..3];

	//Predictors
	[] true -> 0.9611:(router1' = software & router) + 0.0389:(router1' = software | router);
	[] true -> 1.0:(software1' = software);
	[] true -> 0.9611:(breaker1a' = software & breakera) + 0.0389:(breaker1a' = software | breakera);
	[] true -> 0.9611:(breaker1b' = software & breakerb) + 0.0389:(breaker1b' = software | breakerb);

	//Possible IPR States
	[] (router1 = false & software1 = false & breaker1a = false & breaker1b = false) -> (class1' = 1) & (ipr1_State' = 0);
	[] (router1 = false & software1 = false & breaker1a = false & breaker1b = true) -> (class1' = 3) & (ipr1_State' = 1);
	[] (router1 = false & software1 = false & breaker1a = true & breaker1b = false) -> (class1' = 3) & (ipr1_State' = 1);
	[] (router1 = false & software1 = false & breaker1a = true & breaker1b = true) -> (class1' = 3) & (ipr1_State' = 2);
	[] (router1 = false & software1 = true & breaker1a = false & breaker1b = false) -> (class1' = 2) & (ipr1_State' = 3);
	[] (router1 = false & software1 = true & breaker1a = false & breaker1b = true) -> (class1' = 2) & (ipr1_State' = 4);
	[] (router1 = false & software1 = true & breaker1a = true & breaker1b = false) -> (class1' = 2) & (ipr1_State' = 4);
	[] (router1 = false & software1 = true & breaker1a = true & breaker1b = true) -> (class1' = 3) & (ipr1_State' = 5);
	[] (router1 = true & software1 = false & breaker1a = false & breaker1b = false) -> (class1' = 3) & (ipr1_State' = 6);
	[] (router1 = true & software1 = false & breaker1a = false & breaker1b = true) -> (class1' = 3) & (ipr1_State' = 7);
	[] (router1 = true & software1 = false & breaker1a = true & breaker1b = false) -> (class1' = 3) & (ipr1_State' = 7);
	[] (router1 = true & software1 = false & breaker1a = true & breaker1b = true) -> (class1' = 3) & (ipr1_State' = 8);
	[] (router1 = true & software1 = true & breaker1a = false & breaker1b = false) -> (class1' = 0) & (ipr1_State' = 9);
	[] (router1 = true & software1 = true & breaker1a = false & breaker1b = true) -> (class1' = 2) & (ipr1_State' = 10);
	[] (router1 = true & software1 = true & breaker1a = true & breaker1b = false) -> (class1' = 2) & (ipr1_State' = 10);
	[] (router1 = true & software1 = true & breaker1a = true & breaker1b = true) -> (class1' = 3) & (ipr1_State' = 11);

	// State Classifications
	[] (ipr1_State = 0) -> (class1' = 1);
	[] (ipr1_State = 1 | ipr1_State = 2 | ipr1_State = 5 | ipr1_State = 6 | ipr1_State = 7 | ipr1_State = 8 | ipr1_State = 11) -> (class1' = 3);
	[] (ipr1_State = 3 | ipr1_State = 4 | ipr1_State = 10) -> (class1' = 2);
	[] (ipr1_State = 9) -> (class1' = 0);

endmodule
```

Figure 5: Model for the IPR's probabilistic Boolean network.

```
	//States
	ipr1_State : [0..11] init 0; // Initially all systems work properly

	//Classes
	class1 : [0..3];

// State Classifications
[] (ipr1_State = 0) -> (class1' = 1);
[] (ipr1_State = 1 | ipr1_State = 2 | ipr1_State = 5 | ipr1_State = 6 | ipr1_State = 7 | ipr1_State = 8 | ipr1_State = 11) -> (class1' = 3);
[] (ipr1_State = 3 | ipr1_State = 4 | ipr1_State = 10) -> (class1' = 2);
[] (ipr1_State = 9) -> (class1' = 0);
```

Figure 6: Model for the IPR's fault condition.

In Figure 8, the maximum probability of the occurrence of a type-1 fault on any of the IPRs of the test system is presented.

Figure 9 showcases the maximum probability of the occurrence of a type-2 fault on any of the IPRs of the test system.

In Figure 10, we present the maximum probability of the occurrence of a failure in IPR1 of the test system.

In Figure 11, the maximum probability of the occurrence of a simultaneous failure in IPR1 and IPR2 is presented.

PRISM can place labels that can be used to highlight specific or sets of states that can be used to diagnose faults, be they single or multiple. The labels point out specific states or sets of states, which in turn can be used to distinguish between single and multiple faults. In these experiments, whenever the PBN model is used and one of the constituent Boolean networks is chosen, PRISM labels provide a mechanism for filtering the fault conditions that may be occurring, be they failures, faults, or the normal operation of the system. All of the failure modes possible in the IPR, according to what was identified in [9], that are caused by the device's components are modeled, allowing the determination of the device's future state. We can distinguish specific fault, or combinations of faults, and through the use of PRISM's property verification in PCTL, labels produce a time prognosis, or when a fault is expected to happen. Using simulations in PRISM, we can produce graphs such as the ones in Figures 12–16, which show the detection and isolation of different faults and failures.

Table 1: IPR state classification and operational modes.

| Category | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| Type | Fault 1 | Normal operation | Failure | Fault 2 |
| Description | On switching event, the IPR works as intended (activates switches). On nonswitching event, IPR activates switches unnecessarily | On active signal (AS, switching event) the IPR works correctly. On inactive signal (IS, nonswitching event) the IPR does not activate switches (works as intended) | On AS, the IPR does not activate switches. On IS, the IPR activates switches unnecessarily | On AS, the IPR does not activate switches. On IS, the IPR works correctly |
| State(s) | $S_9$ | $S_0$ | $S_3$, $S_4$, $S_{10}$ | $S_1$, $S_2$, $S_5$, $S_6$, $S_7$, $S_8$, $S_{11}$ |

Table 2: Boolean predictor functions and their probability of occurrence for the IPR PBNs.

| Component | Predictor | Selection probability |
|---|---|---|
| Software | $\mathbf{x_1(t+1) = x_1(t)}$ | 1 |
| Router | $\mathbf{x_2(t+1) = x_1(t) \wedge x_2(t)}$ | 0.9611 |
| | $\mathbf{x_2(t+1) = x_1(t) \vee x_2(t)}$ | 0.0389 |
| Main breaker | $\mathbf{x_3(t+1) = x_1(t) \wedge x_3(t)}$ | 0.9611 |
| | $\mathbf{x_3(t+1) = x_1(t) \vee x_3(t)}$ | 0.0389 |
| Secondary breaker | $\mathbf{x_4(t+1) = x_1(t) \wedge x_4(t)}$ | 0.9611 |
| | $\mathbf{x_4(t+1) = x_1(t) \vee x_4(t)}$ | 0.0389 |

The significance of the values is determined by the scientists behind the study and we are asking you to keep them in bold.

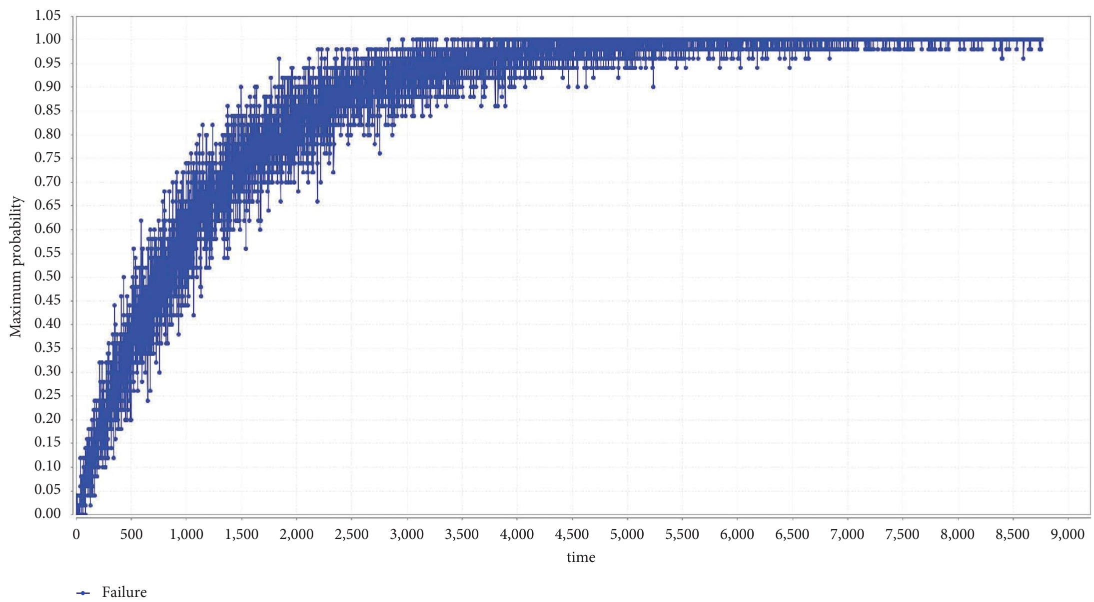


Figure 7: Maximum probability of failure of any IPR in the test system.

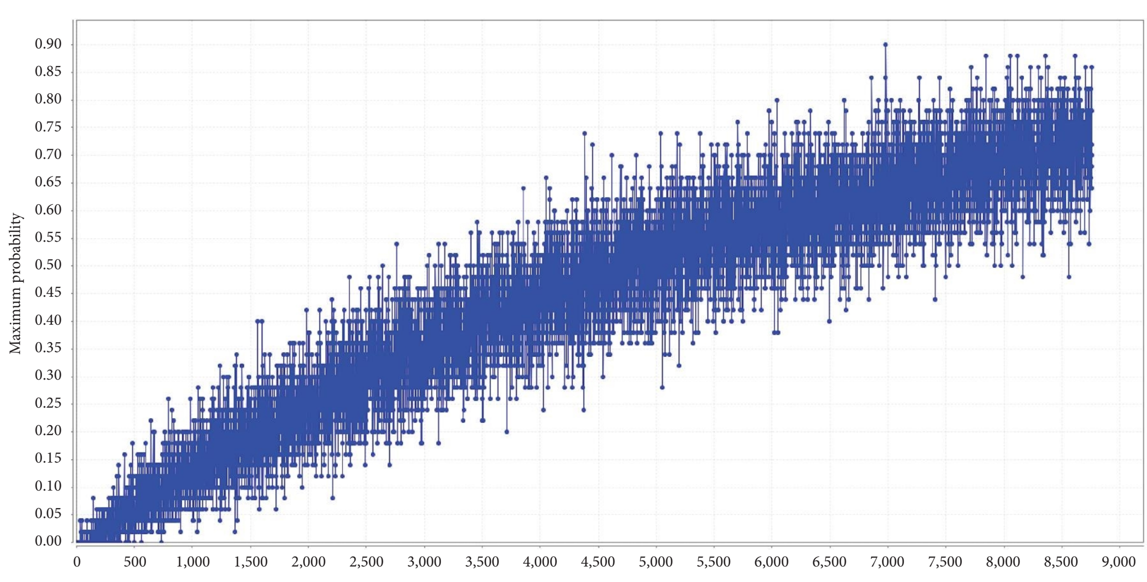


Figure 8: Maximum probability of occurrence of a type-1 fault in any IPR of the test system.

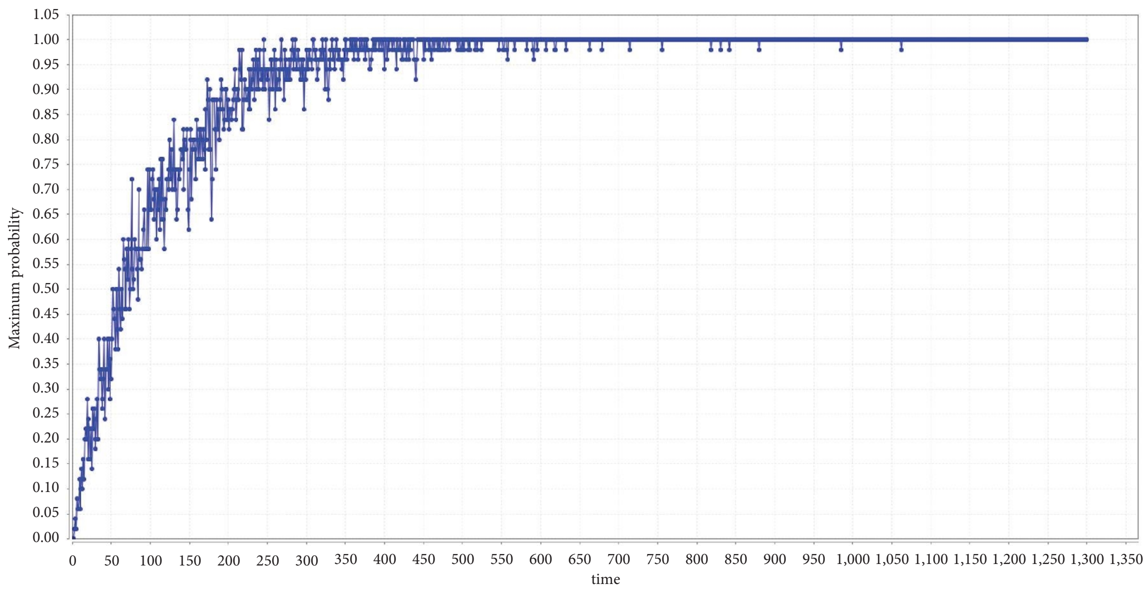


Figure 9: Maximum probability of occurrence of a type-2 fault in any IPR of the test system.

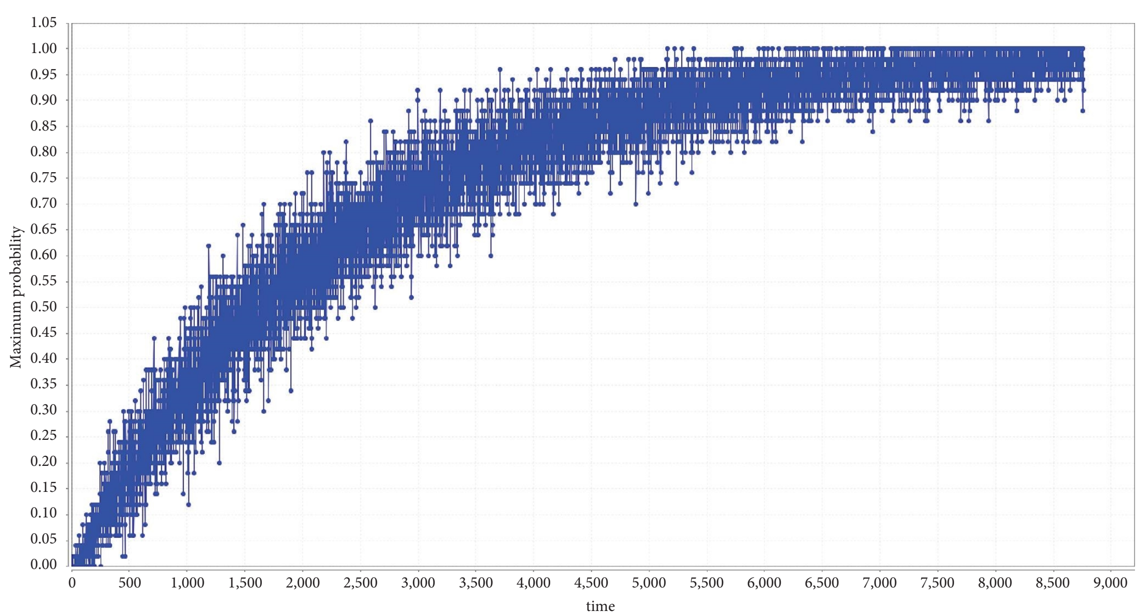


Figure 10: Maximum probability of occurrence of a failure in IPR1.

Figure 12 presents a detection and isolation example of the diagnostic system across one year of operation. In it, we can see single and multiple faults/failures of the IPRs.

In Figure 13 can detect and diagnose, because of the state of the device, the failure modes of IPR1 in one year.

We are able to do the same for two IPRs, and we can see single and multiple IPR faults. Figure 14 presents this for IPRs 1 and 2.

Since each IPR can be in 12 different states and those states in 4 classifications, the states of the IPR have been labeled in PRISM from 0 to 11 and this means that they can be individually identified, per IPR. We are also able to detect specific component failures and faults for each IPR on the test system. Figures 15 and 16 present software failures and failures of the routers connected to the loads (5, 6 and 8) in all IPRs, respectively, after a year of operation.

## 4. Discussion

This diagnosis system is based on a probabilistic Boolean network model of the WSCC 9 bus test system with intelligent power routers using the PRISM model checker. The experiments that can be performed in this software allow for the calculation of the maximum probability of occurrence of the different failure modes of all the IPRs and their respective components. We are also able to detect and isolate faults and

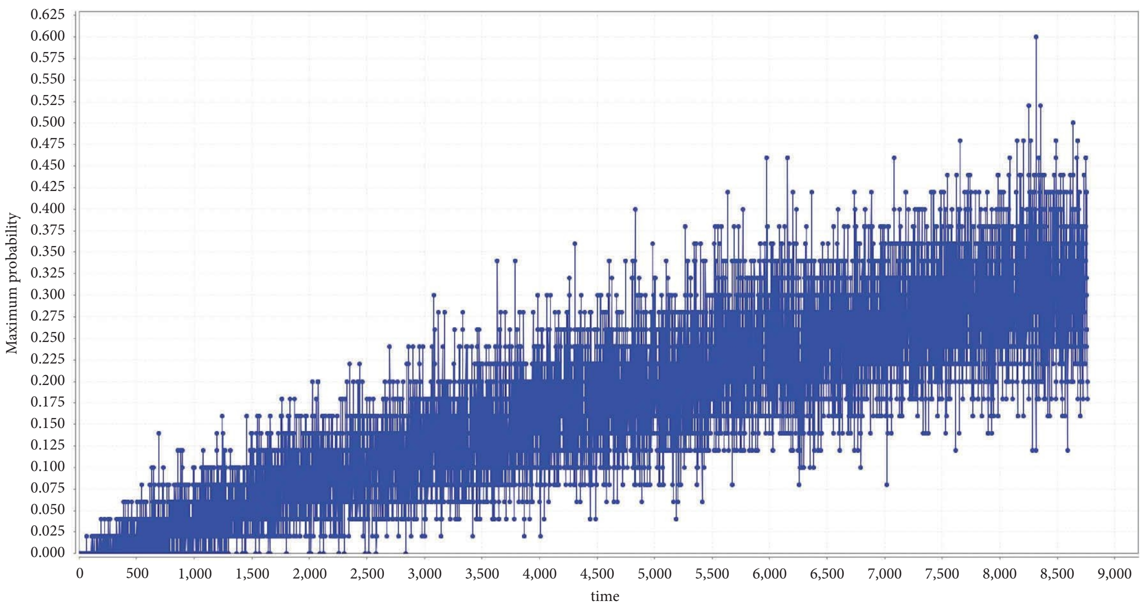


Figure 11: Maximum probability of occurrence of a simultaneous failure in IPR1 and IPR2.

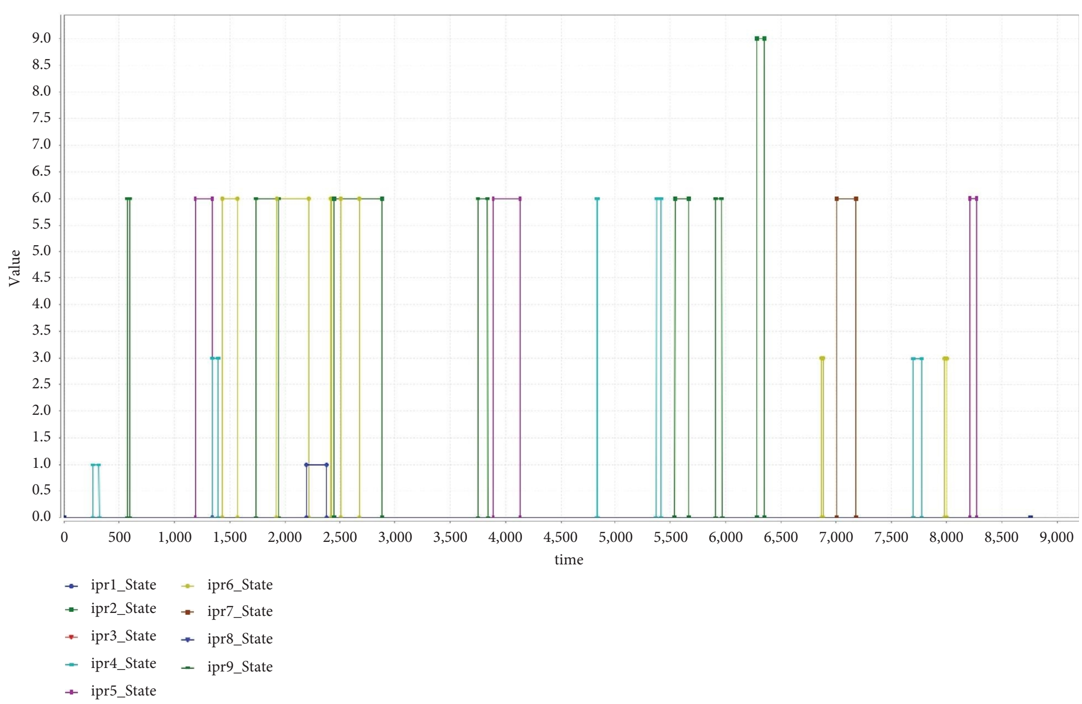


Figure 12: Detection and isolation of single and multiple failures in one year of operation.

failures based on the FDI approach, where we have produced a model for the normal operation of the test system within PRISM and a model for each of the failure modes. With this information, we can simulate the modeled system over time. We have performed simulations based on periods of 8760 hours, or a year of operation. We are able to diagnose, detect, and isolate failure modes, as well as have a prognosis on when these will occur based on the simulations performed. This can be tuned to specific IPR components, to simultaneous faults in different IPR components, to faults in single IPRs, to faults in multiple IPRs, etc. With the model, we can formulate with PCTL in PRISM questions relative to the failure probability of a component and simulate (be it in hours, years, days, minutes, etc.) the system's performance. This simulation produces the state of many components, which can be used further to predict component-level faults as well as device- and system-wide faults. The modeling environment allows us to identify these components and devices individually and isolate the different fault conditions to the respective component or device. This provides

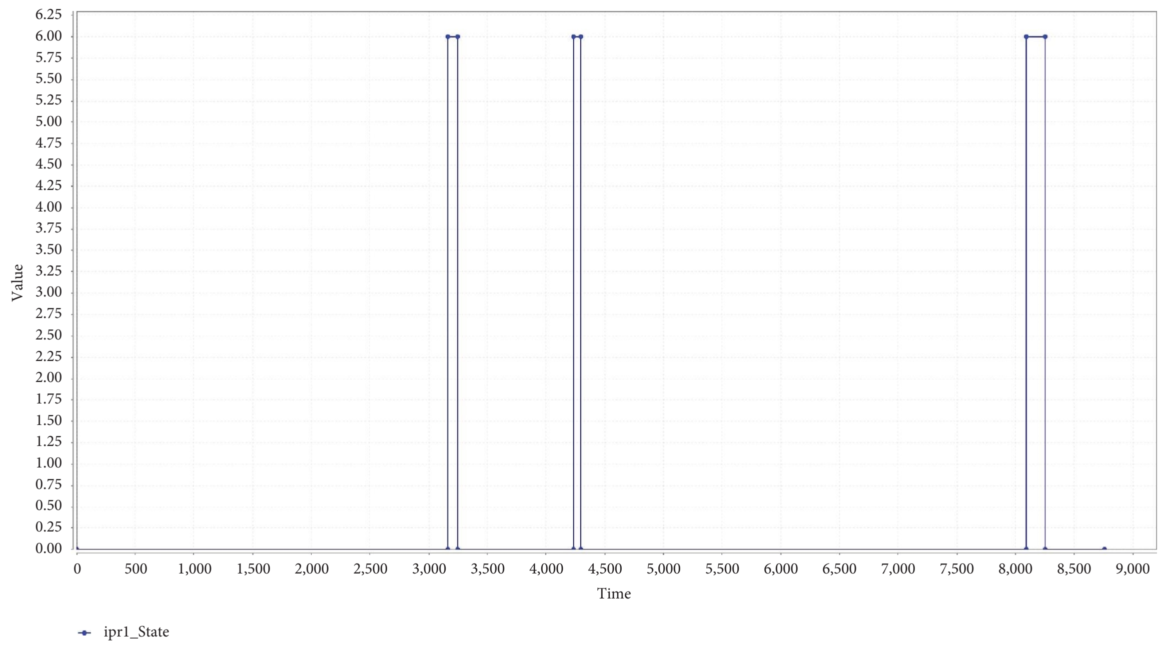


Figure 13: Detection and isolation of single and multiple failures in one year of operation of IPR1.

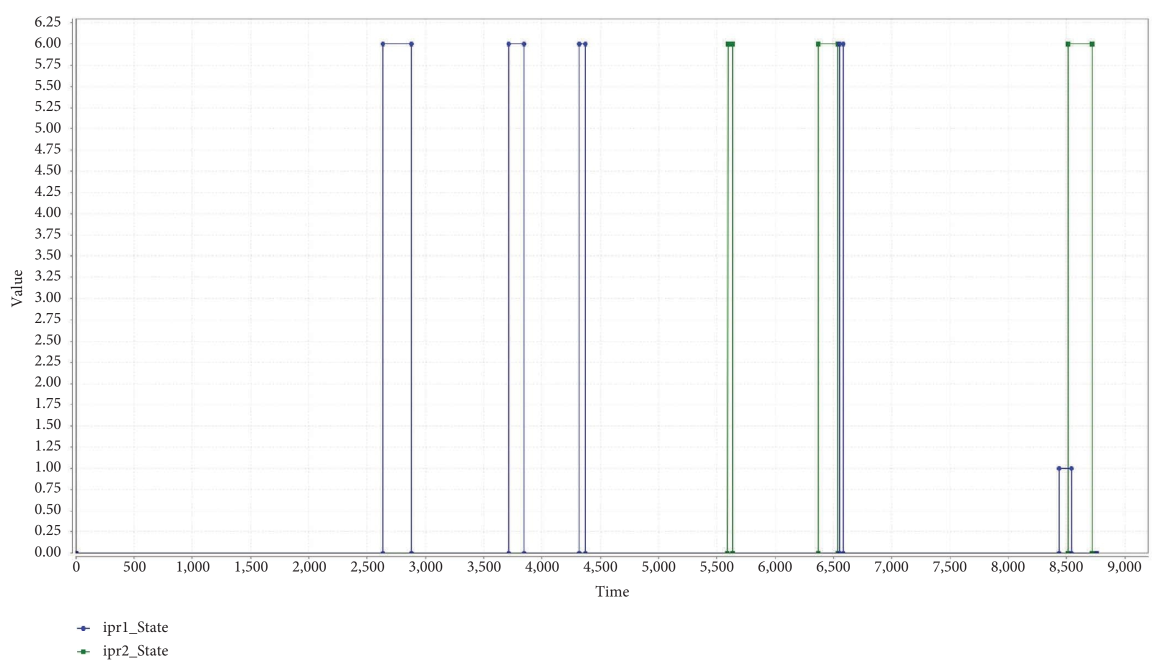


Figure 14: Detection and isolation of failures in one year of operation of IPR1 and IPR2.

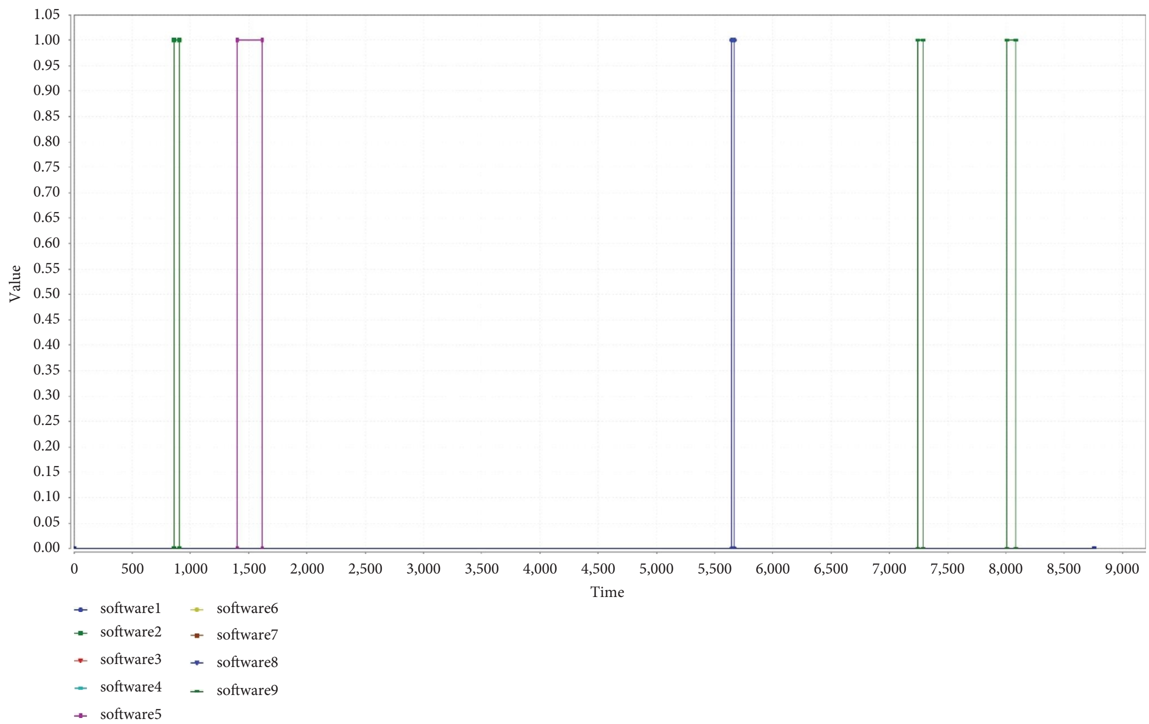


Figure 15: Detection and isolation of software failures in all IPRs after one year of operation.

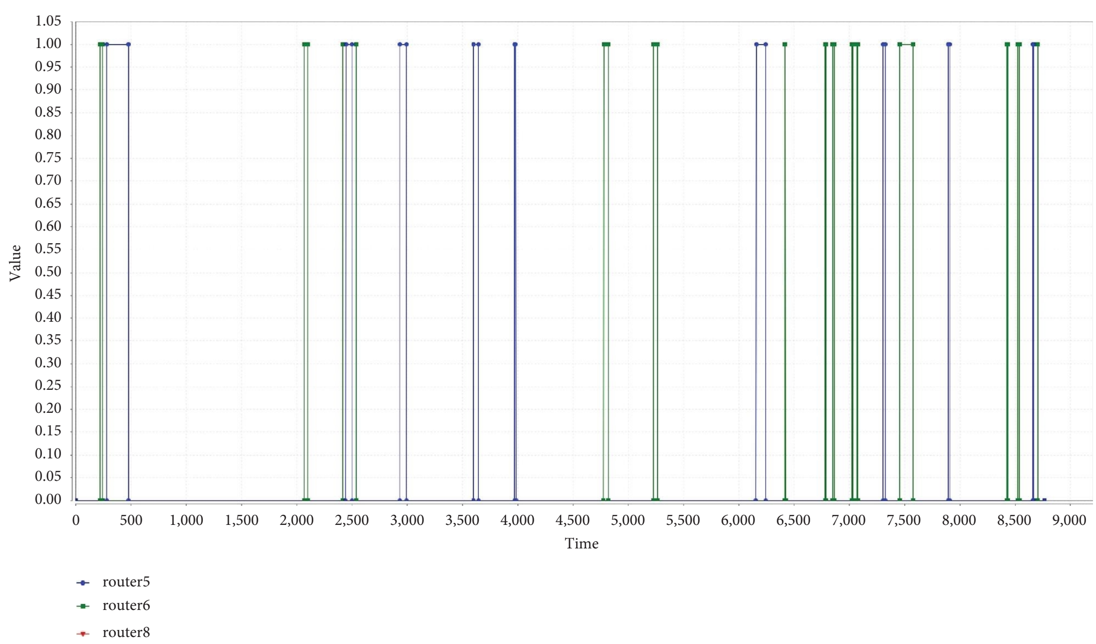


Figure 16: Detection and isolation of router failures in IPRs 5, 6 and 8, after one year of operation.

a comprehensive tool to predict, diagnose, detect, and isolate faults and failures in a smart grid. This provides designers and engineers with a method of designing more robust and resilient smart power systems.

## 5. Conclusions

We presented a diagnostics system for smart-grid systems using intelligent power routers based on a complex systems technique, PBNs, that is able to diagnose, detect, and isolate single and multiple faults and failures, from the component to the device level. In the future, we would like to imbue this system with control techniques based on the PBNs self-organization characteristics and use these to guide the evolution of the system away from fault conditions.

## Data Availability

The data presented in this study are openly available on Github at. https://github.com/pedritostia/IPR-FDI-WSCC9bus.

## Conflicts of Interest

The authors declare that there are no conflicts of interest.

## Acknowledgments

The authors acknowledge the National Autonomous University of México's Postdoctoral Fellowship. This project was self-funded.

## References

[1] M. De Domenico, C. Camargo, C. Gershenson et al., *Complexity Explained: A Grassroot Collaborative Initiative To Create a Set of Essential Concepts of Complex Systems*, 2019,https://complexityexplained.github.io/.
[2] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of Theoretical Biology*, vol. 22, pp. 437–467, 1969.
[3] I. Shmulevich and E. R. Dougherty, *Probabilistic Boolean Networks: Modeling and Control of Gene Regulatory Networks*, SIAM, Philadelphia, PA, 2008.
[4] M. Dorigo, "Learning by probabilistic boolean networks," in *Proceedings of World Congress on Computational Intelligence-IEEE International Conference on Neural Networks*, Orlando, FL, USA, July 1994.
[5] P. J. Rivera Torres, E. I. Serrano Mercado, and L. E. Anido Rifón, "Probabilistic boolean network modeling of an industrial machine," *Journal of Intelligent Manufacturing*, vol. 29, pp. 875–890, 2018.
[6] P. J. Rivera Torres, E. I. Serrano Mercado, and L. E. Anido Rifón, "Probabilistic Boolean Network modeling and model checking as an approach for DFMEA for manufacturing systems," *Journal of Intelligent Manufacturing*, vol. 29, pp. 1393–1413, 2018.
[7] P. J. Rivera Torres, E. I. Serrano Mercado, O. Llanes Santiago, and L. E. Anido Rifón, "Modeling preventive maintenance of manufacturing processes with probabilistic boolean networks with interventions," *Journal of Intelligent Manufacturing*, vol. 29, pp. 1941–1952, 2018.
[8] P. J. Rivera Torres, C. Gershenson García, M. F. Sánchez Puig, and S. Kanaan Izquierdo, "Reinforcement learning with probabilistic boolean network models of smart grid devices," *Complexity*, vol. 2022, Article ID 3652441, 15 pages, 2022.
[9] A. Irizarry Rivera, M. Rodríguez Martínez, and B. Velez, "Intelligent Power Routers: distributed coordination for electric energy processing power networks," in *Operation And Control of Electric Energy Processing Systems*, J. Momoh and L. Mili, Eds., pp. 47–85, John Wiley and Sons, Hoboken, NJ, USA, 2010.
[10] R. Isermann, *Fault-diagnosis Applications: Model-Based Condition Monitoring: Actuators, Drives, Machinery, Plants, Sensors, and Fault-Tolerant Systems*, Springer-Verlag, London, UK, 2011.
[11] P. J. Rivera Torres and O. Llanes Santiago, "Fault Detection and Isolation in smart grid devices using probabilistic Boolean networks," in *Computational Intelligence in Emerging Technologies for Engineering Applications*, vol. 872, pp. 165–185, Springer-Nature, London, UK, 2020.
[12] ICSEG, "WSCC 9-bus system," 2023, https://icseg.iti.illinois.edu/files/2013/10/WSCC9.png.
[13] M. Z. Kwiatkowska, G. Norman, and D. Parker, "Prism 4.0: verification of probabilistic real-time systems," in *Computer Aided Verification. Lecture Notes in Computer Science*, G. Gopalakrishnan and S. Qadeer, Eds., vol. 6806, pp. 585–591, Springer, Berlin, Germany, 2011.