FailSafe: A Generalized Methodology for Converter Fault Detection, Identification, and Remediation in Nanogrids

Jason Poon*, Ioannis C. Konstantakopoulos*, Reza Arghandeh‡, Palak Jain†, Jaime F. Fisac*, S. Shankar Sastry*, Sanjib Kumar Panda†, Costas Spanos*, Seth R. Sanders*

*Department of EECS, University of California, Berkeley
†ECE Department, National University of Singapore
‡Center for Advanced Power Systems, ECE Department, Florida State University

Email: jason@berkeley.edu

***Abstract*—We present the design, implementation, and experimental validation of FAILSAFE—a generalized methodology for fault detection, identification, and remediation (FDIR) for switching power converters in nanogrids. FAILSAFE is a dynamical systems approach to FDIR for switching power converters, and can be applied to a broad class of converters and fault types. FAILSAFE operates as part of the control loop of a switching power converter, and uses the measurements and inputs of the converter to achieve both fault detection and identification (FDI) and fault remediation. In this paper, we present two Modules for FDI—a model-based residual approach and a data-driven multiclass Support Vector Machine (one-vs-one) approach. Moreover, we describe the design of a fault remediation Module by designing optimal control actions in a pre-computed reach-avoid set. We present simulation and experimental results using a prototype nanogrid testbed. Simulation results for the multiclass Support Vector Machine (one-vs-one) FDI Module on a 6-phase interleaved boost converter demonstrate fault detection and identification with a classification accuracy of 98.9% for a current sensor fault and 90.8% for an output capacitor fault. Experimental results for the model-based residual FDI Module on a boost converter demonstrate fault detection and identification in $600\ \mu s$ for a capacitor fault and $250\ \mu s$ for a voltage sensor fault.**

I. INTRODUCTION

Commercial buildings consume nearly one-fifth of the primary energy in the United States. In recent years, the concept of a 'smart building' has emerged as an important academic and industrial effort towards realizing significant improvements in building efficiency, comfort, and intelligence. Integral to the concept of a smart building is its power distribution network, or *nanogrid*, as shown in Fig. 1. As opposed to buildings that purely consume energy, these nanogrids can contain on-site energy resources, such as rooftop photovoltaics or wind turbines. Energy storage buffers, such as batteries or mechanical flywheels, store excess generated energy, which can be used for building power or sold back to the utility. Moreover, electrical loads can be scheduled based on dynamic energy pricing, enabling demand response. Indeed, smart building nanogrids introduce a new paradigm of how buildings consume, generate, and store energy.

However, the confluence of power electronics systems and buildings in these nanogrids has introduced new challenges, particularly with respect to system vulnerability and fault tolerance. Smart building nanogrids are challenging networks to manage and control since electronic loads and distributed energy resources (DERs) impose significant intermittency, uncertainty, and dynamics. DERs make it difficult for system operators to cope with multiple or sequential point of failures. Switching power converters introduce new failure points in a power distribution network. Moreover, the interaction between converters and the propagation or cascading effect of faults through a nanogrid remain open research questions.

In general, systems with high reliability and safety requirements are designed with mechanisms for *fault tolerance*. Fault tolerance is the ability of a system to adapt and compensate, in a systematic way, to random component, sensor, or input faults, while providing completely or partially its intended functionality [1]. There are three key elements to any fault-tolerant system design–component redundancy, a fault detection and identification system [2]–[4], and a remediation or reconfiguration system that, once a fault has been detected and identified, substitutes the faulty component with a redundant one, or reconfigures the control to compensate for the fault.

In this paper, we present FAILSAFE—a generalized methodology for fault detection, identification, and remediation (FDIR) for switching power converters in nanogrids. FAILSAFE is a dynamical systems approach to FDIR for switching power converters, and can be applied to a broad class of converters and fault types.

Fig. 2 provides an overview of the proposed FDIR concept. As shown, FAILSAFE operates as part of the control loop of a switching power converter, and uses the measurements and inputs of the converter to achieve both fault detection and identification (FDI) and fault remediation.

For conceptual simplicity, the operational objectives of FAILSAFE are divided into two *stages*: (1) the fault detection and identification (FDI) stage, and (2) the fault remediation stage. Each stage is comprised of FAILSAFE *Modules*—a

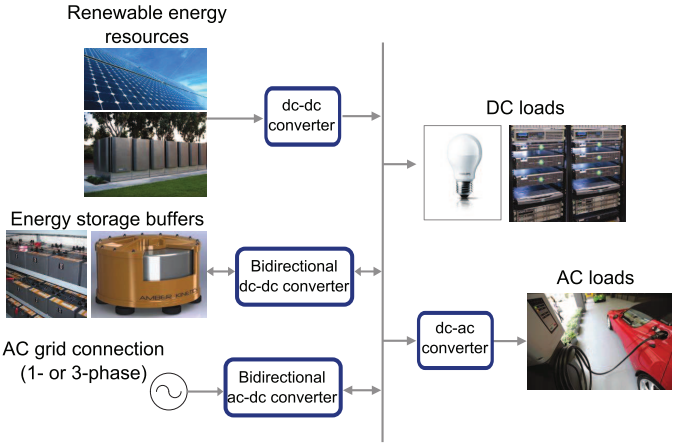


Fig. 1: A prototype nanogrid for power distribution in a smart building.

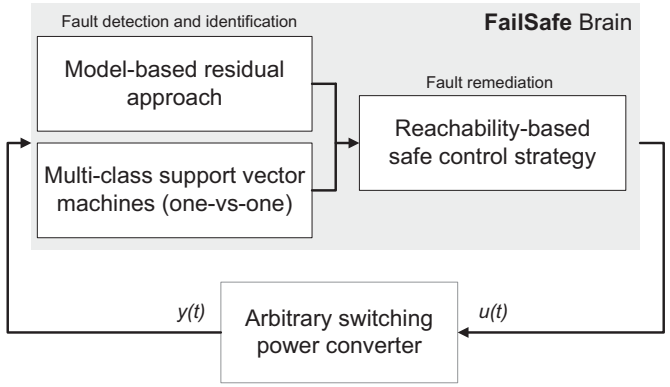


Fig. 2: Overview of FAILSAFE, as applied to an arbitrary switching power converter.

single Module is an algorithmic implementation that can provide the desired FDI or fault remediation functionality. A salient feature of FAILSAFE is that a particular Module or combination of Modules can be chosen for either stage depending on requirements for fault tolerance or limitations of the computation platform.

Moreover, FAILSAFE is implemented on the same computation platform as the converter control system, thus requiring no additional computational hardware. Thus, FAILSAFE complements existing legacy fault protection schemes, in essence, adding a layer of intelligence on top of existing protection hardware and controls, such as fuses, circuit breakers, or current limiting control.

The remainder of the paper is organized as follows. Section II presents an overview of the FAILSAFE methodology, and proposes a set of Modules for converter FDI and fault remediation. Section III presents a real-time hardware implementation of FAILSAFE and an experimental nanogrid testbed that we use to validate the performance of the proposed FDI and fault remediation algorithms. Section IV presents simulation and experimental results for fault detection, identification, and remediation in a boost converter circuit. Section V concludes the paper.

II. FAILSAFE OVERVIEW

In this section, we present the design methodology for FAILSAFE and provide examples of FAILSAFE Modules for the fault detection and identification (FDI) stage and for the fault remediation stage.

A. Stage 1: Fault detection and identification

The objectives of the fault detection and identification stage are two fold: (1) detection, which makes a binary decision whether or not a fault has occurred, and (2) identification, which determines the location of the faulty component, sensor, or input. Here, we present two Modules for FDI—a model-based residual approach and a data-driven multiclass Support Vector Machine (one-vs-one) approach.

1) Model-based residual FDI Module: We propose a model-based residual FDI Module that uses a linear-switched model of the switching power converter. This approach is discussed by the authors in detail in [5].

Fundamentally, the model-based residual FDI Module accepts the same input $u(t)$ as the converter (e.g. PWM signals, input voltages, load currents) and outputs (1) a binary decision whether a fault has occurred, and (2) if a fault has occurred, an index that identifies the particular fault from a fault signature library.

The model-based residual FDI Module is comprised of (1) the real-time model-based estimator, (2) the fault detection logic, (3) the fault signature library, and (4) the fault identification logic.

First, given a switching power converter, we construct a *real-time model-based estimation* that captures the large-signal dynamics of the converter.

The *fault detection logic* consists of a model-based estimator or observer for the switching power converter, which generates an error residual vector of the difference between the measured outputs of the converter and the estimated outputs.

Next, we identify faults of interest in the components and sensors. The dynamics of each of these faults can be uniquely modeled by a scalar *fault magnitude function* and a vector *fault signature*, and are collected in the *fault signature library*.

In the presence of a particular fault, the error residual will evolve according to the dynamics of the fault magnitude function and fault signature. Since these dynamics are calculated a priori, the *fault identification logic* can identify the fault by computing the sliding window L^2-inner product between the error residual vector and the set of fault signatures.

2) Multiclass Support Vector Machines FDI Module: We propose a second FDI Module using multiclass Support Vector Machines (one-vs-one) to classify and detect sensor and component faults.

Multiclass Support Vector Machines (MSVM), specifically MSVM (one-vs-one) and MSVM (one-vs-all), are an extension of binary Support Vector Machines (SVM) [6], [7]. Binary SVM is used for binary classification, that is, classifying data into one of two classes, which can be separated with a hyperplane. Detailed presentation of binary SVM can be found in [8]–[10]. MSVM (one-vs-one) extends binary SVM

Algorithm 1 Proposed algorithm for a multiclass SVM (one-vs-one) FDI Module

1: Measure the states (i.e. voltages and currents) of the converter in the normal and faulted modes of operation.
2: Build an *observation matrix* $\mathfrak{U}$ as follows:

$$\mathfrak{U} = \begin{pmatrix} \mathfrak{u}_n^\top \\ \mathfrak{u}_{n-1}^\top \\ \vdots \\ \mathfrak{u}_2^\top \\ \mathfrak{u}_1^\top \end{pmatrix}$$

where $\mathfrak{u}_\mathfrak{i} \in \mathbb{R}^m$ is a measurement of the converter states in a particular normal or faulted mode of operation, and n is the number of the total measurements. Create a *label vector* $\mathfrak{Y}$ as follows:

$$\mathfrak{Y} = \begin{pmatrix} \mathfrak{y}_n \\ \mathfrak{y}_{n-1} \\ \vdots \\ \mathfrak{y}_2 \\ \mathfrak{y}_1 \end{pmatrix}$$

where $\mathfrak{y} \in \{+1, -1, -2, -3, ..., -f\}$ are the set of f fault labels of the converter ($+1$ represents the normal mode of operation), and n is the number of the total observations.
3: Tune the cost parameter C, and train the MSVM (one-vs-one) model by using the value of C that gives the lowest misclassification rate (MR). A 10-fold cross-validation can be used in tuning and training in order to improve the bias-variance trade-off.
4: In real-time operation, receive new test data and classify using the trained MSVM model.

to a general case with more than two classes by training binary SVM classifiers for all possible combinations of classes.

Multiclass Support Vector Machines (MSVM) have been used in several classification applications, such as detection of multiple power quality disturbances in a power distribution network [11], fault diagnosis of a steam turbine generator [12], assessment of power system security [13], and detection of knee pathologies [14] among others.

Here, we present the design of a FailSafe FDI Module using MSVM (one-vs-one).

First, we will design a binary SVM classifier. Consider a training data set with n samples of the form $\{x_i, y_i\}, i = 1, ..., n$ where $x_i \in \mathbb{R}^m$ represents the different states of the system and $y_i \in \{1, -1\}$ represents the labels for the binary SVM. The objective of binary SVM is to find a hyperplane, $w^\top x + b = 0$, which will optimally separate the data into two classes.

Thus, the binary linear SVM solves the following optimization problem:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & \frac{1}{2}\|w\|^2 + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ \text{subject to} \quad & y_i(w^\top x_i + b) \geq 1 - \xi_i, \xi_i \geq 0,\ i = 1, \ldots, n. \end{aligned}$$

The value of C (cost parameter) in the optimization problem above controls the bias-variance trade-off of the binary SVM. A large [small] value of C will result in a wide [narrow] margin for the optimal hyperplane. A high value of C will cause more observation violations in the hyperplane margin, with higher bias and smaller variance. Thus, one must properly tune the value of C in order to minimize the overall training (misclassification) error.

Next, we will extend the binary SVM classifier to the case with more than two classes by considering a MSVM (one-vs-one) framework. MSVM (one-vs-one) trains binary SVM classifiers for all possible combinations of classes. Thus, for a case with p classes, the classification model will train $\frac{p(p-1)}{2}$ binary SVM classifiers. Given a test sample x, we evaluate each of the $\frac{p(p-1)}{2}$ binary SVM classifiers, and x is classified according to the class with the majority of classification events.

Finally, we define the following performance evaluation metrics:

(a) *Classification accuracy* (CA) determines the percentage correctly classified samples, that is:

$$\frac{\#\text{ of correctly classified samples}}{\#\text{ of total samples}} \times 100$$

(b) *Misclassification rate* (MR) determines the percentage of incorrectly classified samples, that is:

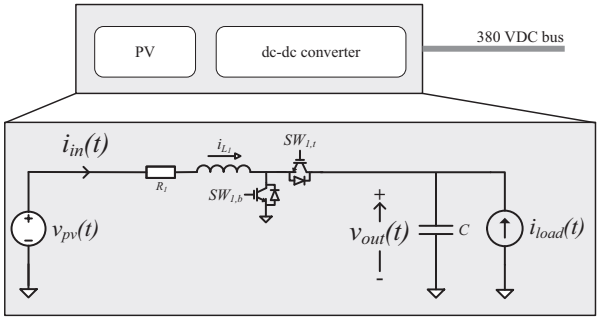
$$\frac{\#\text{ of misclassified samples}}{\#\text{ of total samples}} \times 100$$

B. Stage 2: Fault remediation

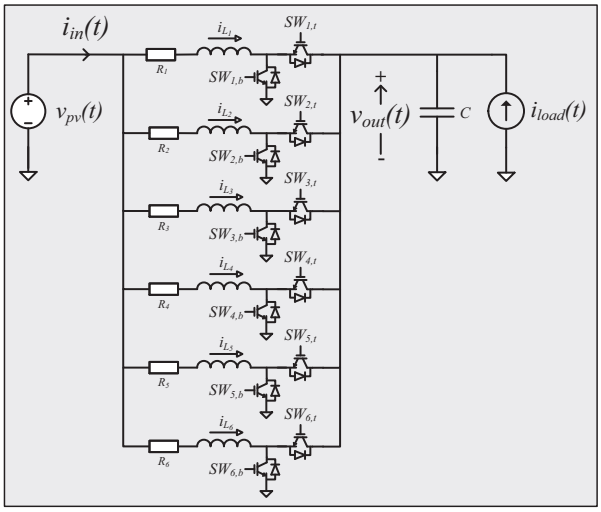
Once the FDI stage has successfully detected and identified a fault in the converter, a fault remediation algorithm can update the dynamical model of the converter and find a control action to either maintain the desired service under this faulty operation, or at least keep the system in a safe operating condition until the fault is duly repaired.

The updated dynamical system will describe the operation of the converter under the identified fault with some bounded tolerance, or uncertainty. It is then possible to use viability theory [15] to calculate a safety region, or *safe set*, for the converter. In particular, given a set of state constraints $\mathcal{K} \subset \mathbb{R}^m$ that need to be met at all times, one would like to compute the associated *viability kernel* $\mathrm{Viab}(\mathcal{K})$, defined as the set of states $x \in \mathcal{K}$ for which there exists a control strategy $u = \kappa(x)$ that will keep any future states from leaving $\mathcal{K}$. Under *uncertain* faulty dynamics, one can also define the robust counterpart to $\mathrm{Viab}(\mathcal{K})$, known as the *discriminating kernel* $\mathrm{Disc}(\mathcal{K})$, which requires this 'safe' control strategy to keep the system in $\mathcal{K}$ for *all* possible behaviors of the system within the defined tolerance.

A family of numerical Hamilton-Jacobi methods [16], [17]



(a) Boost converter topology.



(b) Interleaved boost converter topology.

Fig. 3: Two converter topologies that are used to validate the proposed FDIR concept in simulation and hardware.

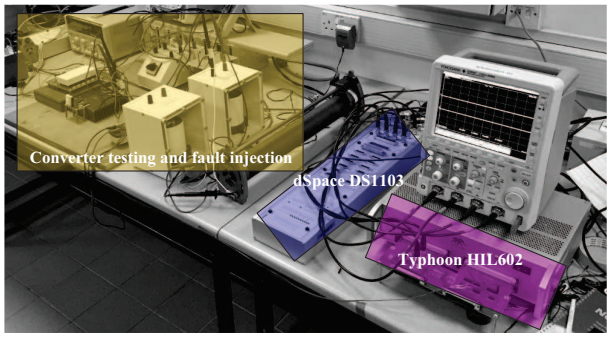


Fig. 4: Nanogrid testbed for validating the FAILSAFE implementation. The converter and FAILSAFE computational hardware are shown.

TABLE I: Specifications and ratings for experimental nanogrid testbed.

Switching power converter	
Topology	1- and 6-phase interleaved boost
Rated input voltage	$0 - 200$ V
Rated output voltage	380 V
Rated output power	2 kW
Switching frequency	50 kHz
R_i	0.1 Ω
L_i	0.1 mH
C	400 μF
Converter control and FDIR computation platform	
Platform	dSpace DS1103 Controller Platform
Simulation time step	100 μs
Real-time power electronics simulator	
Platform	Typhoon HIL602
Simulation time step	500 ns
Analog input sampling rate	1 MHz
PWM switching frequency	50 kHz

makes it possible to compute these safe sets and control inputs for systems with low dimensionality, using dynamic programming under a differential game formulation [18], [19].

It will often be desirable to obtain control actions to not only keep the system within the state constraints $\mathcal{K}$, but in addition drive it into some desired target operating region $\mathcal{T} \subset \mathcal{K}$ (typically to restore nominal service after an initial perturbation caused by the fault). The set of points $x \in \mathcal{K}$ from which it is possible to safely reach $\mathcal{T}$ without leaving $\mathcal{K}$ is usually referred to as the *reach-avoid set* $\mathcal{RA}$, and can be computed with the same numerical tools using a similar formulation [20]. The fault remediation stage will employ such methods to precompute these sets and controls for each fault.

III. IMPLEMENTATION

In order to validate the proposed FAILSAFE FDIR concept, we use a simulation and experimental nanogrid testbed to implement and test the FDIR algorithms.

Both testbeds are modeled as a nanogrid power distribution network in a smart building. Specifically, we focus on dc-dc converters used for interfacing a photovoltaic (PV) energy source with a 380 VDC distribution bus. We consider two dc-dc converter topologies: a boost converter (Fig. 3a) and a 6-phase interleaved boost converter (Fig. 3b). The parameter values for these converters is shown in Table I.

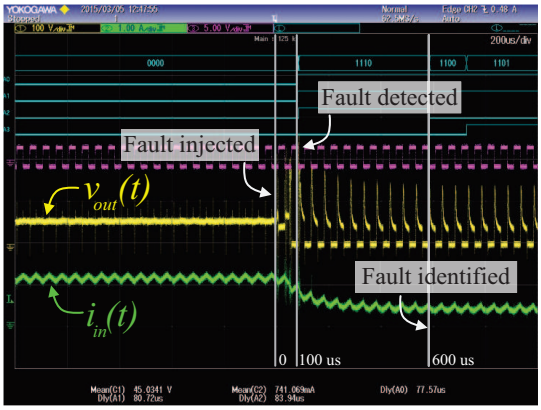
First, we construct a simulation testbed in the MATLAB/Simulink environment using the Piece-wise Linear Electrical Circuit Simulation (PLECS) toolbox [21]. Specifically, this simulation testbed is used to validate multiclass SVM (one-vs-one) FDI Module.

Next, we construct an experimental nanogrid testbed as shown in Fig. 4. The nanogrid testbed consists of: (1) a hardware converter and fault injector, (2) a computational platform for converter control and real-time FDIR implementation, and (3) a high-fidelity real-time power electronics simulator for model-based FDI Modules.

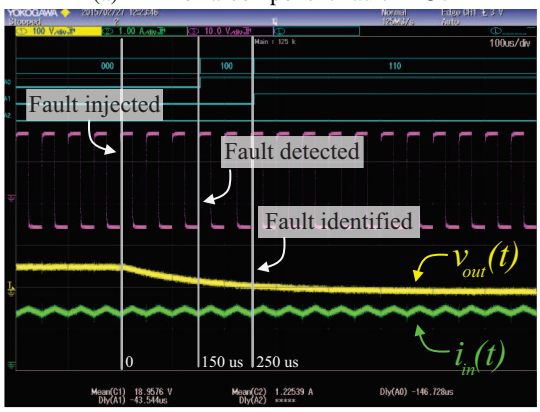
We use a dSpace DS1103 controller board to implement the converter control and FDIR, which operates with a time step of 100 μs. We use a Typhoon HIL 602 to implement the high-fidelity real-time simulator [22], which operates with a time step of 500 ns. This experimental nanogrid testbed is used to validate the model-based residual FDI Module.

IV. RESULTS AND DISCUSSION

In this section, we present simulation and experimental results for the FDI stage of FAILSAFE. First, we validate the model-based residual FDI Module on a boost converter topology using the hardware nanogrid testbed. Next, we validate



(a) FDI for a component fault in C.



(b) FDI for a sensor fault in v_{out}.

Fig. 5: Experimental nanogrid prototype results for the model-based residual FDI Module.

the MSVM (one-vs-one) FDI Module on a 6-phase interleaved boost converter using the simulation testbed.

A. Model-based residual FDI Module

The proposed model-based residual FDI Module is designed for the boost converter according to the methodology presented in Section II-A1. We test the FDI Module on the nanogrid testbed described in Section III.

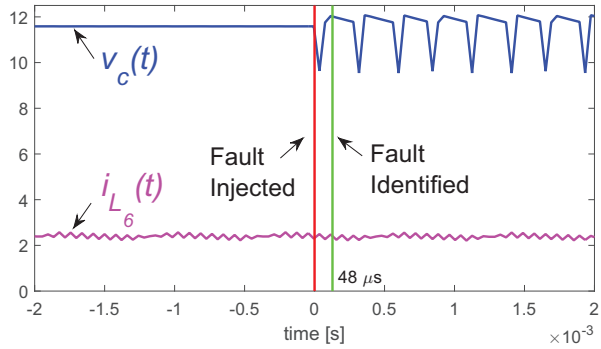
First, we inject a component fault that causes the output capacitance to become zero. As shown in Fig. 5a, the fault causes a large ripple in the output voltage and also causes the input current to fall. The FDI Module detects the fault in 100 μs, and identifies the fault in 600 μs.

Next, we inject a fault that forces the sensor gain of v_{out} to zero. As shown in Fig. 5b, the fault causes the voltage measurement of the output to become zero. The FDI Module detects the fault in 150 μs, and identifies the fault in 250 μs.

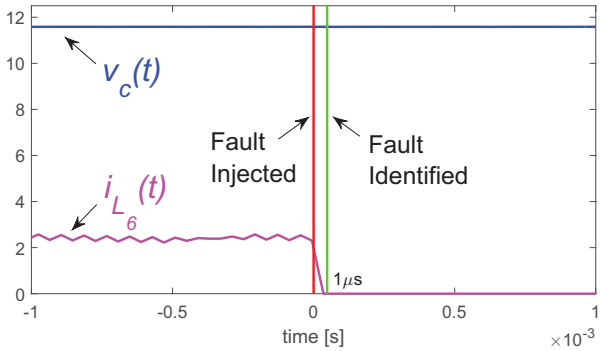
Generally, the time to fault detection depends on the time step of the FDI computation platform (in this case, 100 μs). Fault identification for various component and sensor faults requires the L^2-inner product calculation whose solution generally reaches steady state in under 1 ms.

TABLE II: Classification accuracy and fault identification time of MSVM (one-vs-one) FDI Module with a tuned cost parameter $C = 100$.

Operating mode	CA	t_{fi}
Normal (fault-free)	99.0%	N/A
Capacitor fault	90.8%	32 μs
Sensor fault	98.9%	1 μs



(a) FDI for a component fault in C.



(b) FDI for a sensor fault in i_{L_6}.

Fig. 6: Simulation results for the MSVM (one-vs-one) FDI Module.

B. MSVM (one-vs-one) FDI Module

Here, we will explain the implementation and results of the MSVM (one-vs-one) FDI Module. First, as proposed in Section II-A2, we train the MSVM model using labeled simulation training data for the 6-phase interleaved boost converter, and we obtain a tuned cost parameter of $C = 100$ using a 10-fold cross-validation. In Table II, we show the classification accuracy and fault identification time for the normal operating mode, a fault in the capacitor, and a fault in the sensor. The time to fault identification is determined by the density of the classification events.

Finally, Figs. 6a and 6b show a simulation of the MSVM FDI classifier detecting and identifying a fault in the capacitor and a fault in the current sensor, respectively. As shown, the MSVM FDI Module identifies the capacitor fault in 32 μs and the current sensor fault in 1 μs.

V. Conclusions

We have presented the design, implementation, and experimental validation of a generalized methodology for fault detection, identification, and remediation (FDIR) for switching power converters in nanogrids. The dynamical systems approach enables the technique to be applied to a broad class of converters and fault types. In this paper, we demonstrated simulation and experimental results on a nanogrid testbed for two fault detection and identification Modules—one using a model-based residual approach, and the other using a data-driven multiclass Support Vector Machine (one-vs-one) approach. Moreover, we presented the design of a fault remediation Module by designing optimal control actions in a pre-computed reach-avoid set. The proposed methodology can encapsulate the dynamics of a broad class of converter topologies and faults. In this way, FAILSAFE enables a flexible and scalable solution for improving reliability and fault tolerance in an array of power electronics applications.

References

[1] J. Laprie, *Dependability: Basic Concepts and Terminology*. New York, NY: Springer-Verlag, 1991.

[2] R. J. Patton, P. M. Frank, and R. N. Clarke, *Fault Diagnosis in Dynamic Systems: Theory and Application*. Upper Saddle River, NJ: Prentice-Hall, Inc., Oct. 1989.

[3] R. J. Patton, P. M. Frank, and R. N. Clark, *Issues of Fault Diagnosis for Dynamic Systems*. Springer Publishing Company, Incorporated, Oct. 2010.

[4] J. Chen and R. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Springer Publishing Company, Incorporated, Nov. 2012.

[5] J. Poon, I. Konstantakopoulos, C. Spanos, and S. Sanders, "Real-time model-based fault diagnosis for switching power converters," in *Applied Power Electronics Conference and Exposition (APEC), 2015 IEEE*, pp. 358–364, March 2015.

[6] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *Neural Networks, IEEE Transactions on*, vol. 13, no. 2, pp. 415–425, 2002.

[7] A. Mathur and G. Foody, "Multiclass and binary svm classification: Implications for training and classification users," *Geoscience and Remote Sensing Letters, IEEE*, vol. 5, no. 2, pp. 241–245, 2008.

[8] C. Huang, L. Davis, and J. Townshend, "An assessment of support vector machines for land cover classification," *International Journal of remote sensing*, vol. 23, no. 4, pp. 725–749, 2002.

[9] G. M. Foody and A. Mathur, "A relative evaluation of multiclass image classification by support vector machines," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 42, no. 6, pp. 1335–1343, 2004.

[10] G. James, D. Witten, and T. Hastie, "An introduction to statistical learning: With applications in r.," 2014.

[11] W.-M. Lin, C.-H. Wu, C.-H. Lin, and F.-S. Cheng, "Detection and classification of multiple power-quality disturbances with wavelet multiclass svm," *Power Delivery, IEEE Transactions on*, vol. 23, no. 4, pp. 2575–2582, 2008.

[12] C. Zou, E.-h. Zheng, H.-w. Xu, and L. Chen, "Svm-based multiclass cost-sensitive classification with reject option for fault diagnosis of steam turbine generator," in *Machine Learning and Computing (ICMLC), 2010 Second International Conference on*, pp. 66–70, IEEE, 2010.

[13] S. Kalyani and K. S. Swarup, "Classification and assessment of power system security using multiclass svm," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 41, no. 5, pp. 753–758, 2011.

[14] P. Levinger, D. T. Lai, R. Begg, K. Webster, J. Feller, and W. Gilleard, "The application of multiclass svm to the detection of knee pathologies using kinetic data: A preliminary study," in *Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on*, pp. 589–594, IEEE, 2007.

[15] J.-P. Aubin, A. Bayen, and P. Saint-Pierre, *Viability Theory: New Directions*. Springer, 2011.

[16] J. Sethian, "A fast marching level set method for monotonically advancing fronts.," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 93, pp. 1591–1595, Feb. 1996.

[17] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*, vol. 153. Springer Science & Business Media, 2003.

[18] J. Lygeros, "On reachability and minimum cost optimal control," *Automatica*, vol. 40, pp. 917–927, June 2004.

[19] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, pp. 947–957, July 2005.

[20] K. Margellos and J. Lygeros, "Hamilton-Jacobi formulation for reach-avoid differential games," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1849–1861, 2011.

[21] J. Alimeling and W. Hammer, "Plecs-piece-wise linear electrical circuit simulation for simulink," in *Proc. of IEEE International Conference on Power Electronics and Drive Systems*, (Hong Kong), 1999.

[22] D. Majstorovic, I. Celanovic, N. D. Teslic, N. Celanovic, and V. A. Katic, "Ultralow-latency hardware-in-the-loop platform for rapid validation of power electronics designs," *IEEE Transactions on Industrial Electronics*, vol. 58, pp. 4708–4716, Oct. 2011.