Robust Testing of Cascading Failure Mitigations Based on Power Dispatch and Quick-Start Storage

Zhe Xu, *Student Member, IEEE*, A. Agung Julius, *Member, IEEE*, and Joe H. Chow, *Fellow, IEEE*

***Abstract*—We present a formal robust testing method for power system cascading failure mitigations. The approach is model-based, using simulated trajectories of the system and proving that uncertainties, e.g., in the initial states or disturbances, do not perturb the trajectories beyond a robust neighborhood around them. We model power systems as hybrid systems with locations representing different swing dynamics and relay dynamics. We present implementations of our robust testing approach in a three-machine system model from the 2003 Italian Blackout and the IEEE 39-bus system model. We apply the robust testing method in two scenarios for averting the cascading failures: 1) robust testing of safety for various generator mechanical power dispatch schedules and 2) robust testing of safety for postfault remedial actions based on quick-start storage.**

***Index Terms*—Cascading failures, hybrid system, quick-start storage, robust neighborhood, robust testing.**

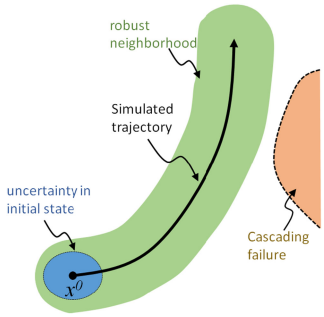


Fig. 1. Simulated trajectory from initial state x^0 (also called the *nominal trajectory* in this paper) can be equipped with a robust neighborhood such that variations in the initial state or bounded disturbance will not deviate the trajectory beyond the robust neighborhood. If this neighborhood excludes the states leading to cascading failures, we can formally verify the safety of the compact set of initial states.

I. Introduction

CASCADING failures of the power grid (such as the 2003 blackouts in the Northeastern U.S. and in Italy [1]) can have devastating effects on almost every aspect of modern society [2], [3]. A cascading failure happens when the failure of one power line or generator triggers the failure of other power lines or generators, which may further lead to power instability. Cascading outages are essentially related to the power system transient stability and various dynamics such as swing dynamics and relay dynamics [4], [5].

For a long time, time-domain numerical simulations and direct methods [6] are the two main approaches in analyzing power system dynamics and transient stability. Recently, reachability analysis has emerged as an effective technique to compute overapproximated reachable sets of systems trajectories considering uncertainties of initial states, disturbances, and others. For example, the level set method [7] is utilized to analyze and design power grid dynamic performances by computing the reachable sets. In the meantime, the trajectory (or simulation, test) based approach is increasingly applied because of its simplicity and highly parallelizability. Probabilistic methods such as Monte–Carlo simulations are applied to simulate many trajectories to cover the state space. However, these methods can only offer a probabilistic statement about system safety. In this paper, we propose a robust neighborhood approach to cover an uncountable compact set of states with finitely many simulated trajectories and robust neighborhoods around them. See Fig. 1 for a conceptual illustration of our approach.

For linear systems, Lyapunov stability analysis can be used to compute the robust neighborhood [8], [9]. For nonlinear systems, several different approaches have been proposed. In [10], the authors propose a trajectory-based reachability analysis using local finite-time invariance property. In [11], the authors propose a conservative linearization method for reachability analysis for power systems, where the reachable sets are represented by zonotopes. In [12], the authors propose the discrepancy function, which generalizes other measures of trajectory convergence and divergence such as contraction metrics and incremental Lyapunov functions. In [13], the authors propose the local discrepancy function and use matrix perturbation theory to calculate the robust neighborhood in the form of balls or ellipsoids iteratively at each local time point.

In this paper, we use the robust neighborhood approach to perform robust testing of power systems in mitigating cascading failures. We model the dynamics of the system as the dynamics of a hybrid system, combining the swing dynamics and the relay dynamics. We propose the bounded disturbance local discrepancy function (BDLDF) based on [13] for calculating the local discrepancy function with the effect of bounded

Manuscript received October 5, 2016; revised January 10, 2017; accepted March 17, 2017. Date of publication April 14, 2017; date of current version November 22, 2018. This work was supported in part by the National Science Foundation under Grant CNS-1218109, Grant CNS-1550029, and Grant CNS-1618369.

The authors are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: xuz8rpi.edu; juliua2rpi.edu; chowj@rpi.edu).

Digital Object Identifier 10.1109/JSYST.2017.2686401

disturbances. We use ellipsoids to represent the robust neighborhood and bound the error term utilizing the special characteristics of power system dynamics. We find the robust neighborhood so that every possible trajectory in the robust neighborhood satisfies the safety criterion specified by signal temporal logic (STL) specifications [14]–[16] with the knowledge of relay dynamics. The safety criterion guarantees that subsequent line overloading as a result of initial line trips is not severe enough for further line trips (cascading failures).

The two cases that we consider in this paper are the following.

1) *Robust testing of various generator mechanical power dispatch schedules:* We test the robustness of a power dispatch schedule in averting cascading failures, i.e., a line trip does not lead to the tripping of other lines.
2) *Robust testing of postfault remedial actions based on quick-start storage:* When a major line trips, the distribution of power flow is changed significantly. In [17], generation rescheduling and load shedding are proposed as two main methods for redistributing the power flow to avert cascading failures. While generation rescheduling has the disadvantage of slow response time (a few seconds to a few minutes), load shedding also suffers from the significant instantaneous cost and the inconveniences brought to consumers. Recently, energy storage [18], [19] has been increasingly utilized as a third alternative for cascading failure mitigations. Although energy storage is also costly to install and maintain, it can facilitate effective utilization of variable sources with much shorter response time (a few milliseconds for quick-start storages such as Lithium-ion batteries and supercapacitors) and avoid the instantaneous cost of load shedding. We test the robustness of the quick-start storage strategy in averting cascading failures. Particularly, we perform robust testing with respect to various starting rates of the storage.

This paper is organized as follows. Section II briefly reviews the definitions of trajectories and simulations of hybrid systems. Section III presents the algorithm for the robust neighborhood computation for nonlinear hybrid systems with STL specifications. Section IV presents the robust testing algorithms and the implementations of the algorithms on a three-machine power system model and the IEEE 39-bus system model to test the robustness of the cascading failure mitigations for both generator power dispatch schedules and emergency remedial actions. Finally, some conclusions are presented in Section V.

II. Hybrid System Trajectories and Simulations

Definition 1 (Hybrid automaton): A hybrid autonomous system is defined to be a 5-tuple $H = (L \times X, L^0 \times X^0, f, E, \text{Inv})$ [20], [21].

1) $L \times X$ is a set of hybrid states (ℓ, x), where $\ell \in L$ is the discrete state (location), and $x \in X$ is the continuous state.
2) $L^0 \times X^0 \subset L \times X$ a set of initial states.
3) f associates with each location $\ell \in L$ the autonomous continuous time-invariant dynamics, $f_\ell : \dot{x} = f_\ell(x)$, which is assumed to admit a unique global solution $\xi_\ell(t, x_\ell^0)$, where ξ_ℓ satisfies $\frac{\partial \xi_\ell(t, x_\ell^0)}{\partial t} = f_\ell(\xi_\ell(t, x_\ell^0))$, and $\xi_\ell(0, x_\ell^0) = x_\ell^0$ is an initial condition in ℓ.
4) $\text{Inv} : L \to 2^X$ associates an invariant set $\text{Inv}(\ell) \subset X$ with each location. Only if the continuous state satisfies $x \in \text{Inv}(\ell)$, can the discrete state be at the location ℓ.
5) E is a set of events. In each location ℓ, the system state evolves continuously according to f_ℓ until an event $e := (\ell, \ell', g, r), e \in E$ occurs. The event is guarded by $g \in \text{Inv}(\ell)$. Namely, a necessary condition for the occurrence of e is $x \in g$. After the event, the state is reset from (ℓ, x) to $(\ell', r(x))$.

When a hybrid system runs, the system state alternately flows continuously and triggers events in E. For convenience, we also define an initialization event $e^0 \notin E$. Then, a trajectory of the system can be defined as a sequence.

Definition 2 (Trajectory): A trajectory of a hybrid system H is denoted as $\rho = \{(e^i, \ell^i, x^i, T^i)\}_{i=0}^{N_\ell}$,

where

1) $\forall i \geq 0$, $(\ell^i, x^i) \in L \times X$, and $(\ell^0, x^0) \in L^0 \times X^0$;
2) $\forall i \geq 0$, $T^i \in \mathbb{R}_{\geq 0}$ (nonnegative real), and $\forall t \in [0, T^i]$, $\xi_{\ell^i}(t, x^i) \in \text{Inv}(\ell^i)$;
3) $\forall i \geq 1$, $e^i = (\ell^{i-1}, \ell^i, g^i, r^i)$, $\xi_{\ell^{i-1}}(T^{i-1}, x^{i-1}) \in g^i$, $x^i = r^i(\xi_{\ell^{i-1}}(T^{i-1}, x^{i-1}))$, i.e., (ℓ^i, x^i) is the reset state for $(\ell^{i-1}, \xi_{\ell^{i-1}}(T^{i-1}, x^{i-1}))$.

Definition 3 (Simulation): A $\{(x^i, \tau, \epsilon, T^i)\}_{i=0}^{N_\ell}$-simulation ψ of the hybrid system H is a sequence of time-stamped compact sets $(R^i[k], t^i[k])$ (for every i, where $k = 1, 2, \ldots, N_T^i$) and a sequence of hyperrectangles or parallelepipeds $Q(t^i[k-1], t^i[k])$ satisfying the following.

1) The first time in each location is $t^i[0]$ and the last time in each location is $t^i[N_T^i] = t^i[0] + T^i$. $\forall i \geq 0, k \geq 1$, $0 < t^i[k] - t^i[k-1] \leq \tau$, where the parameter τ is the maximal sampling period.
2) $\forall i \geq 0, k \geq 1$, $\xi_{\ell^i}(t^i[k] - t^i[0], x^i) \in R^i[k] \subseteq \mathbb{R}^n$. The diameter of $R^i[k]$ is bounded by ϵ, i.e., $\sup_{x_1, x_2 \in R^i[k]} \|x_1 - x_2\| \leq \epsilon$.
3) $\forall t \in [t^i[k-1] - t^i[0], t^i[k] - t^i[0]]$, $\xi_{\ell^i}(t, x^i) \in Q(t^i[k-1], t^i[k])$.

Validated simulation libraries such as VNODE-LP [22] and CAPD [23] can give strict error bounds in the forms of hyperrectangles or parallelepipeds that include all possible simulation errors between adjacent simulation time points. For example, in the VNODE-LP library, the error bounds between adjacent simulation time points are referred to as the apriori bounds (which can be calculated by [24, Algorithm I]) and the error bounds for each simulation point are referred to as the tight bounds (which can be calculated by [24, Algorithm II]).

For an $\{(x^i, \tau, \epsilon, T^i)\}_{i=0}^{N_\ell}$-simulation ψ of the hybrid system H, we can define the nominal (simulated) trajectory of ψ. With slight abuse of notation, we define the nominal (simulated) trajectory ρ_ψ as a sequence of points $\xi_{\ell^i}(t^i[k] - t^i[0], x^i) = \text{center}(R^i[k]) = \arg\max_{x \in R^i[k]} \inf\{\|x - y\| | y \in \text{cl}(\mathbb{R}^n \setminus R^i[k])\}$, where $\text{cl}(\cdot)$ means the closure of a set, $\|\cdot\|$ denotes the 2-norm.

III. Robust Neighborhood Computation for Nonlinear Hybrid Systems With STL Specifications

In this section, we introduce the robust neighborhood computation algorithms for nonlinear hybrid systems with STL specifications. We first present the algorithm of computing the BDLDF for a general nonlinear system in Section III-A and then present the specific bounding method for the error term in Section III-B. Finally, we extend the methods to hybrid systems and present the algorithm of robust neighborhood computation for nonlinear hybrid systems with STL specifications in Section III-C.

A. Bounded Disturbance Local Discrepancy Function

We consider a nonlinear dynamical system of the form

$$\dot{x} = f(x). \tag{1}$$

In addition, we also consider a perturbed version of the same dynamics

$$\dot{x}_\zeta = f(x_\zeta) + \zeta(t) \tag{2}$$

where $\zeta(t)$ is a bounded disturbance. Following the convention in the previous section, we denote the solution trajectories of (1) with initial condition x^0 as $\xi(t, x^0)$, the solution of (2) with the same initial condition and disturbance $\zeta(t)$ as $\hat{\xi}(t, x^0; \zeta)$.

For two initial states x^0 and $\tilde{x}^0$ with disturbance $\zeta(t)$, define $z(t) \triangleq \hat{\xi}(t, \tilde{x}^0; \zeta) - \xi(t, x^0)$. Intuitively, $\xi(t, x^0)$ and $\hat{\xi}(t, \tilde{x}^0; \zeta)$ represent the nominal (simulated) trajectory and a perturbed trajectory.

Following [13], we can write

$$\dot{z}(t) = \left(\int_0^1 J_f\left(\xi(t, x^0) + sz(t)\right)\, ds \right) z(t) + \zeta(t) \tag{3}$$

where $J_f(x) \triangleq \frac{\partial f(x)}{\partial x}$ is the Jacobian of $f(x)$. Furthermore,

$$\frac{d}{dt}\|z(t)\|^2 = z^T(t)\Big(\int_0^1 (J_f(\xi(t, x^0) + sz(t)) + J_f^T(\xi(t, x^0) + sz(t)))ds \Big) z(t) + z^T(t)\zeta(t) + \zeta^T(t)z(t). \tag{4}$$

For this nonlinear dynamics, we find a time-varying function to bound the right-hand side of (4) and, therefore, bound the growth of the difference between the nominal trajectory and the perturbed one. The time-varying function is described in [13] as the local discrepancy function. To make the bound even tighter, we define a linear coordinate transformation as in [13], $v(t) \triangleq Vz(t)$. Then,

$$\frac{d}{dt}\|v(t)\|^2 = v^T(t)\left(\int_0^1 \left(\tilde{J}_f\left(\xi(t, x^0) + sz(t)\right) + \tilde{J}_f^T(\xi(t, x^0) + sz(t))\right) ds \right) v(t) + v^T(t)V\zeta(t) + \zeta^T(t)V^T v(t) \tag{5}$$

where $\tilde{J}_f(x) = VJ_f(x)V^{-1}$. One effective candidate of V is the inverse of the (generalized) eigenvector matrix of the average of the Jacobian $J_f(\xi(t, x^0))$ for a period of time (for details on the selection of V, see [13, Sec. 3.3]). We denote $M \triangleq V^T V$, $\phi_M(x, y) \triangleq \sqrt{(x-y)^T M (x-y)}$.

Algorithm 1: BDLDF computation for nonlinear dynamics.

procedure ComputeBDLDF(ψ, ρ_ψ, $\tilde{J}_f(x)$, L_f, $\gamma[0]$, μ)
  for $k \leftarrow 1$ to N_T **do**
    $\tau \leftarrow t[k] - t[k-1]$
    $\tilde{J} \leftarrow \tilde{J}_f\big((\xi(t[k-1], x^0) + \xi(t[k], x^0))/2\big)$
    $\lambda \leftarrow \lambda_{\max}(\tilde{J} + \tilde{J}^T)/2$
    $d \leftarrow \gamma[k-1]\left\|V^{-1}\right\|$
    $\tilde{d} \leftarrow e^{L_f \tau} d$
    $\tilde{S}[k] \leftarrow \text{Hull}\Big(\text{Box}\big(\text{Ball}(\xi(t[k-1], x^0), \tilde{d})\big),$
    $\text{Box}\big(\text{Ball}(\xi(t[k], x^0), \tilde{d})\big)\Big) \oplus Q(t[k-1], t[k])$
    $Error \leftarrow \max_{x \in \tilde{S}[k]} \left\| \tilde{J}_f(x) + \tilde{J}_f^T(x) - \tilde{J} - \tilde{J}^T \right\|$
    $\tilde{b}[k] \leftarrow \lambda + Error/2$
    $\gamma[k] \leftarrow (\gamma[k-1] + \epsilon)e^{\tilde{b}[k]\tau} + \mu\|V\|/\tilde{b}[k](e^{\tilde{b}[k]\tau} - 1)$
  end for
  return $\gamma[k] (k = 1, 2, \ldots, N_T)$
end procedure

In the following, we assume the disturbance $\zeta(t)$ is bounded by μ, i.e., $\|\zeta(t)\| \le \mu, \mu \in \mathbb{R}_{\ge 0}$.

Definition 4: Given the system of equations (1), (2), if the initial states is a compact set X^0, then a uniformly continuous function γ: $X^0 \times X^0 \times [0, T] \to \mathbb{R}_{\ge 0}$ is a BDLDF of the system if there exists positive-definite matrix M such that for any initial states $\tilde{x}^0, x^0 \in X^0$, $\phi_M(\hat{\xi}(t, \tilde{x}^0; \zeta), \xi(t, x^0)) \le \gamma(\tilde{x}^0, x^0, t)$.

The BDLDF actually defines the robust neighborhood of a trajectory at every time point in the shape of an ellipsoid with the radius being $\gamma(\tilde{x}^0, x^0, t)$. In the following, for discrete time-stamped trajectories in an (x^0, τ, ϵ, T)-simulation, we denote $\gamma[k] \triangleq \gamma(\tilde{x}^0, x^0, t[k])$. We denote $B(x, \gamma[k]) \triangleq \{x' | \phi_M(x, x') \le \gamma[k]\}$ as the robust neighborhood around x with radius $\gamma[k]$.

Theorem 1: Considering the system of equations (1), (2) with the initial set X^0 and a sequence of time points $t[0] < t[1] < \ldots < t[N_T] = T$, for any $\tilde{x}^0$ and $x^0 \in X^0$, $\gamma(\tilde{x}^0, x^0, t)$ is a BDLDF of the system of equations (1), (2), if

$$\gamma(\tilde{x}^0, x^0, t) = \begin{cases} \phi_M(\hat{\xi}(t, \tilde{x}^0; \zeta), \xi(t, x^0)), & \text{if } t = t[0] \\ (\gamma[k-1] + \epsilon)e^{\tilde{b}[k](t - t[k-1])} & \\ \quad + \mu\|V\|/\tilde{b}[k](e^{\tilde{b}[k](t-t[k-1])} - 1), & \text{if } t \in (t[k-1], t[k]] \end{cases}$$

where $\tilde{b}[k]$ is defined as in Algorithm 1.

The algorithm to calculate the BDLDF for nonlinear dynamics is shown in Algorithm 1. The input of the ComputeBDLDF function are an (x^0, τ, ϵ, T)-simulation ψ and its nominal (simulated) trajectory ρ_ψ, the Jacobian matrix $\tilde{J}_f(x)$, the Lipschitz constant L_f, the initial BDLDF value $\gamma[0]$, and a disturbance bound μ. Algorithm 1 starts from the initial set

(ellipsoid) $B(x^0, \gamma[0])$ and calculate the time-varying radius $\gamma[k]$ iteratively. We use the following notations: $\lambda_{\max}(A)$ is the largest eigenvalue of matrix A, Hull(S_1, S_2) is the convex hull of the two sets S_1, S_2, Ball$(x, r) \triangleq \{x' | \|x' - x\| \leq r\}$, the set-based addition $\mathcal{X} \oplus \mathcal{Y} = \{x + y | x \in \mathcal{X}, y \in \mathcal{Y}\}$, and the outer box of a set S of states is the minimal multidimensional interval that contains S, i.e. Box$(S) \triangleq [x_{\text{lower}}, x_{\text{upper}}]$, where x_{lower} and x_{upper} are the lower and upper bounds of x in each dimension in S. The calculation of the convex over-approximated region $\tilde{S}[k]$ takes two steps: 1) (lines 6 and 7) the radius of the smallest ball that covers the ellipsoid $B(\xi(t[k-1], x^0), \gamma[k-1])$ is calculated as d and bloated through the Lipschitz constant L_f to obtain $\tilde{d}$; 2) (lines 8 and 9) $\tilde{S}[k]$ is calculated as the set-based addition of $Q(t[k-1], t[k])$ and the convex hull of the outer boxes of the two balls of radius $\tilde{d}$ centered at $\xi(t[k-1], x^0)$ and $\xi(t[k], x^0)$, respectively. Now, the only term that needs to be determined in Algorithm 1 is the error term in line 10, and it can be calculated with the method described in Section III-B.

Remark 1: Algorithm 1 is modified from Algorithm ComputeLDF in [13], but as the bounded disturbance is added, the representation of $\tilde{b}[k]$ is different from $b[i]$ in [13]. Besides, we take the outer box of Ball$(\xi(t[k-1], x^0), \tilde{d})$ and Ball$(\xi(t[k], x^0), \tilde{d})$ to make $\tilde{S}[k]$ a polytope (which is to make the computation of the error term easier in Section III-B), and we use the $Q(t[k-1], t[k])$ to include the simulation errors between adjacent simulation points, so $\tilde{S}[k]$ is an overapproximation of S in [13], and thus, by [13, Lemma 3.2], $\tilde{S}[k]$ is an overapproximation of the reachable set of states between $t[k-1]$ and $t[k]$.

B. Upper Bound of the Error Term

Following [13], we use $E(x)$ to denote $\tilde{J}_f(x) + \tilde{J}_f^T(x) - \tilde{J} - \tilde{J}^T$ for each iteration in Algorithm 1. We denote cond(V) as the condition number of the matrix V.

Theorem 2: $\|E(x)\| \leq 2\text{cond}(V) \|J_f(x) - J\|_F$.

The upper bound of $\|J_f(x) - J\|_F$ in the convex hull $\tilde{S}[k]$ can be calculated as follows:

$$\max_{x \in \tilde{S}[k]} \|J_f(x) - J\|_F \leq \sqrt{\sum_{i=1}^{N} \sum_{j=1}^{N} \max_{x \in \tilde{S}[k]} (J_f(x)_{ij} - J_{ij})^2} \quad (6)$$

where N is the dimension of state x, $J_f(x)_{ij}$ is the (i,j)th entry of the matrix $J_f(x)$, and $\max_{x \in \tilde{S}[k]} (J_f(x)_{ij} - J_{ij})^2$ can be calculated as the maximum of $(\max_{x \in \tilde{S}[k]} J_f(x)_{ij} - J_{ij})^2$ and $(\min_{x \in \tilde{S}[k]} J_f(x)_{ij} - J_{ij})^2$.

We find conservative upper and lower bounds for each term $J_f(x)_{ij}$ using interval arithmetics. Specifically, for the power system dynamics that we consider in this paper, $J_f(x)_{ij}$ can be expressed as the sum of terms of the following form:

$$\varrho(\chi) = m \sin(a\chi + b), \chi \in [\chi_{\text{lower}}, \chi_{\text{upper}}] \quad (7)$$

where $m, a,$ and b are nonnegative real numbers, χ represents either state or linear combination of states, and χ_{lower} and χ_{upper} are the minimal and maximal values of χ in $\tilde{S}[k]$ (which can be easily computed by linear programming as $\tilde{S}[k]$ is convex). The upper and lower bounds of each term $\varrho(\chi)$ can be calculated by interval arithmetics of sinusoidal functions.

C. Algorithm for Robust Neighborhood Computation for Nonlinear Hybrid Systems With STL Specifications

In this section, we apply the analysis of bounding the growth of differences between trajectories of a hybrid system. In the following, we use the superscript i to denote the corresponding variables in location ℓ^i. For brevity, we denote $\phi^i(x, y) = \sqrt{(x-y)^T M^i (x-y)}$. We want all trajectories initiated from within $B(x^0, \gamma^0[0]) = \{x | \phi^0(x, x^0) \leq \gamma^0[0]\}$ to have the same qualitative behavior as the nominal (simulated) trajectory (i.e., triggering the same guards, satisfying the same STL specifications, etc.). As some of the trajectories may leave later than the nominal (simulated) trajectory, we use the method in [8] to extend the time horizon to $[t^i[0], t^i[0] + T^i + T_{\text{lag}}]$ so that all trajectories initiated from the robust neighborhood are guaranteed to leave Inv(ℓ^i) before $t^i[0] + T^i + T_{\text{lag}}$. While in general any kind of STL specification can be used, we consider the following two forms (as they are used in the power system cascading failure mitigation scenarios in Section IV):

$$\varphi_s^i = \neg \Diamond_{[t^i[0], t^i[0] + T^i + T_{\text{lag}}]} \pi^i \quad (8)$$

$$\varphi^i = \neg \Diamond_{[t^i[0], t^i[0] + T^i + T_{\text{lag}}]} \Box_{[0, T'^i]} \pi^i \quad (9)$$

φ_s^i reads as "During the entire time interval of $[t^i[0], t^i[0] + T^i + T_{\text{lag}}]$, the state should never be in the unsafe set π^i." φ^i reads as "During the entire time interval of $[t^i[0], t^i[0] + T^i + T_{\text{lag}}]$, the state should not stay in the unsafe set π^i for more than T'^i." Note that if i is the last location of the simulated trajectory, then $[t^i[0], t^i[0] + T^i + T_{\text{lag}}]$ should be replaced by $[t^i[0], T_{\text{end}}]$, where T_{end} is the end of simulation time.

The algorithm to calculate the robust neighborhood of the nominal (simulated) trajectory with respect to STL specifications $\varphi^i (i = 0, 1, \ldots, N_\ell)$ is shown in Algorithm 2. Note that Algorithm 2 is specifically written for the STL formula φ^i in the form of (9). If other forms of STL formulas are used, one needs to change Algorithm 2 (such as line 22) according to the specific STL formulas. The computation proceeds in reverse order, i.e., from the last location to the first location. In each location, we calculate the distances (in the metric of $\phi^i(x, y)$) between the state of nominal (simulated) trajectory and the states of the avoided sets (lines 4–11). The avoided sets have two parts: the unsafe sets π^i and the avoided part of the guards. We define Ξ^i as the collection of guards of location ℓ^i, $\breve{g}^{i+1}$ as the part of g^{i+1} such that the image through the reset map r^{i+1} is inside the robust neighborhood around the reset initial state (as shown in Fig. 2), which can be expressed formally as follows:

$$\breve{g}^{i+1} \leftarrow \{x \in g^{i+1} | r^{i+1}(x) \in B^{i+1}(x^{i+1}, \gamma^{i+1}[0])\}. \quad (10)$$

The avoided part of guards can be expressed as follows:

$$\Xi^i \setminus \breve{g}^{i+1} \triangleq (\Xi^i \setminus g^{i+1}) \cup (g^{i+1} \setminus \breve{g}^{i+1})$$

Algorithm 2: Robust neighborhood computation with respect to STL specifications $\varphi^i(i=0,1,\ldots,N_\ell)$.

1: **procedure** ROBUSTTUBE(H, ψ, ρ_ψ, r_0, a_s, T_lag, μ)
2: **for** $i \leftarrow N_\ell$ to 0 **do**
3: $k \leftarrow 0$
4: **while** $t^i[k] - t^i[0] < T^i + T_\mathrm{lag}$ **do**
5: $k \leftarrow k+1$
6: **if** $i < N_\ell$ **then**
7: Obtain $\breve{g}^{i+1}$ using (10)
8: $d^i_\mathrm{g}(t^i[k]) \leftarrow \inf_{y \in \Xi^i \setminus \breve{g}^{i+1}} \phi^i(\xi_{\ell^i}(t^i[k], x^i), y)$
9: **end if**
10: $d^i(t[k]) \leftarrow \inf_{y \in \pi^i} \phi^i(\xi_{\ell^i}(t^i[k], x^i), y)$
11: **end while**
12: flag $\leftarrow 0$, $\gamma^i[0] \leftarrow r_0$, $N^i_T \leftarrow k$
13: **while** flag $= 0$ **do**
14: flag $\leftarrow 1$
15: Compute $\tilde{J}^i_f(x)$ and L^i_f from H
16: $\gamma^i[k](k=1,\ldots,N^i_T) \leftarrow ComputeBDLDF$
17: $(\psi^i, \rho^i_\psi, \tilde{J}^i_f(x), L^i_f, \gamma^i[0], \mu)$
18: **for** $k \leftarrow 0$ to N^i_T **do**
19: **if** $i < N_\ell \wedge \gamma^i[k] > d^i_\mathrm{g}(t^i[k])$ **then**
20: flag $\leftarrow 0$
21: **end if**
22: **if** there exists j such that $t^i[j] - t^i[k] > T'^i$ and $\gamma^i[k] > d^i(t^i[k])$ and $\gamma^i[k+1] > d^i\ (t^i[k+1])$ and $\ldots \gamma^i[j] > d^i(t^i[j])$ **then**
23: flag $\leftarrow 0$
24: **end if**
25: **end for**
26: **if** flag $= 0$ **then**
27: $\gamma^i[0] \leftarrow a_\mathrm{s}\gamma^i[0]$ $\triangleright 0 < a_\mathrm{s} < 1$
28: **end if**
29: **end while**
30: **end for**
31: **return** $\gamma^0[0]$
32: **end procedure**

where $\Xi^i \setminus g^{i+1}$ should be avoided because it triggers events different from the event e^{i+1} triggered by the nominal trajectory, and $\breve{g}^{i+1}$ is excluded from the avoided part of guards since it is allowed for trajectories initiated from the computed neighborhood to reach $\breve{g}^{i+1}$ and trigger e^{i+1}. The distance between the nominal (simulated) trajectory and the unsafe set π^i at time $t^i[k]$ is denoted as $d^i(t^i[k])$, while the distance between the nominal (simulated) trajectory and the avoided part of the guards at time $t^i[k]$ is denoted as $d^i_\mathrm{g}(t^i[k])$.

Next, we give an initial guess of $\gamma^i[0]$ as r_0 (line 12) and gradually shrink the radius until the robust neighborhood of the nominal trajectory is found (lines 13–29). In each iteration, Algorithm 1 is called to calculate $\gamma^i[k](k=0,1,\ldots,N^i_T)$ in the time horizon $[t^i[0], t^i[0]+T^i+T_\mathrm{lag}]$ (lines 16 and 17, where ψ^i and ρ^i_ψ are the segment of ψ and ρ_ψ in location ℓ^i). $\gamma^i[k]$ is compared with $d^i_\mathrm{g}(t^i[k])$ (lines 19–21) and $d^i(t^i[k])$ (lines 22–24) at relevant time points to determine whether the STL specifications $\varphi^i(i=0,1,\ldots,N_\ell)$ are satisfied for the trajectories in the current guess of the robust neighborhood. If the answer is yes, then $\gamma^0[0]$ is returned; otherwise, $\gamma^0[0]$ is shrunk by $a_\mathrm{s}(0<a_\mathrm{s}<1)$ and the whole procedure is rerun until $\varphi^i(i=0,1,\ldots,N_\ell)$ are satisfied.

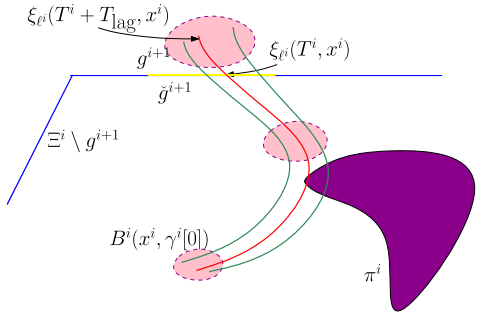


Fig. 2. Trajectories that start from the robust neighborhood $B^i(x^i, \gamma^i[0])$ are guaranteed to enter ℓ^{i+1} through $\breve{g}^{i+1}$ and not enter π^i for more than T'^i.

Remark 2: As exact values of $d^i(t^i[k])$ can be hard to compute when π^i is nonconvex, a lower bound of $d^i(t^i[k])$ can be calculated instead for conservative computation of the robust neighborhood. Another way to circumvent the computation of $d^i(t^i[k])$ is elaborated in the examples in Section IV. It applies to the situation where the unsafe set can be represented in the form of $F^i(x)>0$, where $F^i(x)$ is a function of the state x. In short, we compute $\bar{F}^i[k]$ as the upper bound of $F^i(x)$ for all x in the current guess of the robust neighborhood around $\xi_{\ell^i}(t^i[k], x^i)$. If $\bar{F}^i[k] \le 0$ is always true in $[t^i[0], t^i[0]+T^i+T_\mathrm{lag}]$, then all trajectories in the current guess of the robust neighborhood are guaranteed to satisfy φ^i_s. If $\bar{F}^i[k] \le 0$ is false for less than T'_i in $[t^i[0], t^i[0]+T^i+T_\mathrm{lag}]$, then all trajectories in the current guess of the robust neighborhood are guaranteed to satisfy φ^i.

IV. Robust Testing of Power System Cascading Failure Mitigations

A. Three-Machine Power System Model

In this section, we give a case study of the Italian blackout happened on September 28, 2003. In [25], the authors built a model to simulate the dynamics of the cascading process and concluded that the cascading failure happened because of the interaction between the transient stability governed by the swing dynamics and the protection operation. In this paper, we use a three-machine power system model to simulate the Italian blackout. Fig. 3 shows the three-machine power system model modified from [25]. The synchronous machines G_i $(i=1,2,3)$ represent the power systems of Switzerland and France, and the loads represent the Italian power system. Arrows in the figure denote constant impedance loads.

The classical model describes the swing dynamics of machine G_i $(i=1,2)$ by the following ordinary differential equations:

$$\begin{cases} \dot{\delta}_i = \omega_i \\ \dfrac{P_{\mathrm{r}i}}{P_b}\dfrac{H_i}{\pi f_\mathrm{s}}\dot{\omega}_i = P_{\mathrm{G}i} - D_i\omega_i - p_{\mathrm{e}i}(\delta_1, \delta_2) \end{cases} \tag{11}$$

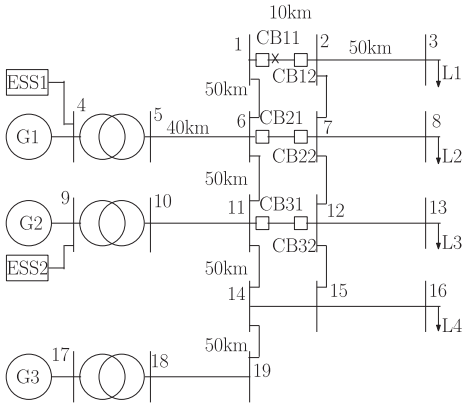


Fig. 3. Modified three-machine power system model for the Italian blackout.

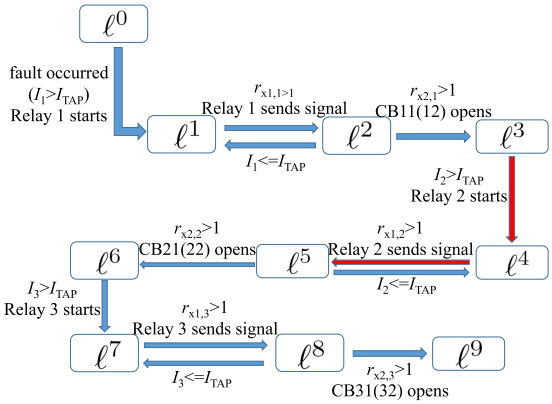


Fig. 4. Locations of the hybrid system H describing the series of events of the cascading failure.

where δ_i is the rotor angle position of G_i with respect to the infinite bus at G_3, ω_i is the rotor speed deviation of G_i relative to system angular frequency $2\pi f_\mathrm{s}$, P_b is the base MVA in the per unit system, $P_{\mathrm{r}i}$ is the rated power of G_i, H_i is the per-unit inertia constant, $P_{\mathrm{G}i}$ is the mechanical input power to G_i, D_i is the damping coefficient of G_i, and $P_{\mathrm{e}i}$ stands for the electrical output power of G_i and is given by the following function of δ_1, δ_2 ($\delta_3 = 0$ in the following equation):

$$P_{\mathrm{e}i}(\delta_1,\delta_2) \triangleq G_{ii}E_i^2 + \sum_{j=1,j\neq i}^{3} E_iE_j\{G_{ij}\cos(\delta_i-\delta_j) + B_{ij}\sin(\delta_i-\delta_j)\} \quad (12)$$

where E_i is the terminal voltage of G_i, G_{ii} is its internal conductance, and $G_{ij}+jB_{ij}$ is the transfer admittance between G_i and G_j.

There are several kinds of overcurrent relays, which are in use in the current power systems. In this paper, we focus on the induction-disc-type overcurrent relays as described in [26]. The internal dynamics of the induction-disc type overcurrent relays is described by ordinary differential equations of the internal state variable $r_{\mathrm{x}1}$ for the relay and the internal state variable $r_{\mathrm{x}2}$ for the circuit breaker. The internal dynamics of the relay is as follows:

$$\begin{aligned} \dot{r}_{\mathrm{x}1} &= f_\mathrm{r}(I)(1-\nu(r_{\mathrm{x}1}-r_{\mathrm{TDS}})g_\mathrm{r}(I) - \nu(-r_{\mathrm{x}1})(1-g_\mathrm{r}(I))) \\ z_\mathrm{r} &= \nu(r_{\mathrm{x}1}-r_{\mathrm{TDS}}) \end{aligned} \quad (13)$$

where

$$f_\mathrm{r}(I) = K((I/I_{\mathrm{TAP}})^2-1) \quad g_\mathrm{r}(I) = \nu(I-I_{\mathrm{TAP}})$$
$$\nu(x) \triangleq \begin{cases} 0, & \text{if } x \le 0 \\ 1, & \text{if } x > 0 \end{cases} \quad (14)$$

where the function ν is a step function of state $r_{\mathrm{x}1}$, I_{TAP} is the prescribed threshold value of input current I, r_{TDS} is the threshold value of state $r_{\mathrm{x}1}$, K is the acceleration factor of the internal dynamics, and z_r is the discrete output of the overcurrent relay.

The internal dynamics of the circuit breaker is as follows:

$$\begin{aligned} \tau_{\mathrm{tr}}\dot{r}_{\mathrm{x}2} &= z_\mathrm{r}\big(1-\nu(r_{\mathrm{x}2}-r_{\mathrm{tr}})\big) \\ z_{\mathrm{cb}} &= \nu(r_{\mathrm{x}2}-r_{\mathrm{tr}}) \end{aligned} \quad (15)$$

where r_{tr} is the threshold value of $r_{\mathrm{x}2}$, and τ_{tr} is the time constant for the circuit breaker internal dynamics, z_{cb} is the discrete output of the circuit breaker (when z_{cb}=1, the circuit breaker opens).

In this paper, we set $P_\mathrm{b} = 2000$ MVA, $f_\mathrm{s} = 50$ Hz, $P_{\mathrm{r}1} = P_{\mathrm{r}2} = 2000$ MW, $P_{\mathrm{r}3} = 200$ GW, the voltage base is 400 kV, the transformer impedance is 0.15 (all the values here and in the following are per unit values unless otherwise specified), and the initial active power flow to load $\mathrm{L}_1, \mathrm{L}_2, \mathrm{L}_3$, and L_4 are 0.25, 0.95, 0.2, and 0.1, respectively. For $i = 1, 2, 3$, $H_i = 5$ s, $D_i = 0.05$, and $E_i = 1$. The line inductance is 0.9 mH/km, and the line resistance and capacitance are neglected. The relay parameters $I_{\mathrm{TAP}} = 1.05$, $K = 16$, $r_{\mathrm{TDS}} = 1$, the circuit breaker parameters $\tau_{\mathrm{tr}} = 0.1$ s, $r_{\mathrm{tr}} = 1$. We assume that a three-phase line-to-ground fault occurs near CB11; then, the cascading behavior of the power system can be modeled as a hybrid system H with ten locations corresponding to different generator swing dynamics and relay dynamics, as shown in Fig. 4. We use I_1, I_2, and I_3 to denote the line current amplitudes of line 1-2, line 6-7, and line 11-12, respectively. $r_{\mathrm{x}1,1}, r_{\mathrm{x}1,2}$, and $r_{\mathrm{x}1,3}$ are the internal state variable for the relays corresponding to CB11(12), CB21(22), and CB31(32), respectively. $r_{\mathrm{x}2,1}, r_{\mathrm{x}2,2}$, and $r_{\mathrm{x}2,3}$ are the internal state variable for the circuit breaker CB11(12), CB21(22), and CB31(32), respectively. The simulated trajectories start from the fault-on location ℓ^1 and x^1 (the initial state at location ℓ^1) is also the initial state of the entire nominal (simulated) trajectory.

After the fault happens, if the relay corresponding to CB11(12) sends the signal to the circuit breaker (which normally is bound to happen as a protection for in-section fault), then the dynamics move to location ℓ^2. If CB11 and CB12 are opened (the time interval between CB11 opens and CB12 opens is neglected), then the dynamics move to location ℓ^3. Fig. 5 shows that when $P_{\mathrm{G}1} = 0.9$ and $P_{\mathrm{G}2} = 0.6$, the circuit breakers CB11(12), CB21(22), and CB31(32) open consecutively at 0.12889, 0.58421, and 0.78355 s. After the three lines trip, the machines lose synchronism with the infinite bus and the system becomes unstable.

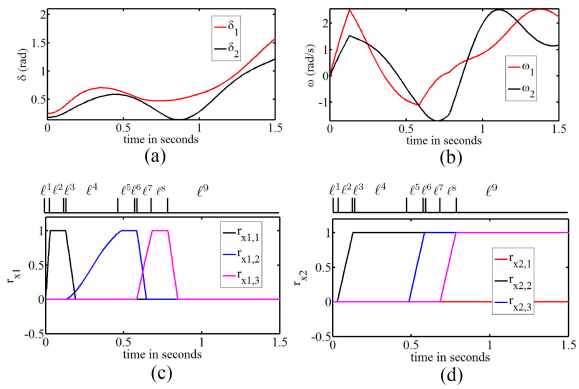


Fig. 5. Swing dynamics and protection operation in the three-machine power system model when $P_{\mathrm{G1}} = 0.9$ and $P_{\mathrm{G2}} = 0.6$.

1) Robust Testing of Generator Power Dispatch Schedules: To avert the cascading failure, it is essential for CB21(22) not to open after CB11(12) opens. To achieve this, we present the following two methods.

Method 1: In the hybrid system H, we simulate a nominal trajectory ρ_ψ that starts from ℓ^1 and we treat the guard set between location ℓ^4 and ℓ^5 as the unsafe set π^4. The state $x = [\delta_1, \delta_2, \omega_1, \omega_2, P_{\mathrm{G1}}, P_{\mathrm{G2}}, r_{\mathrm{x1},1}, r_{\mathrm{x2},1}, r_{\mathrm{x1},2}, r_{\mathrm{x2},2}]^T$ and the continuous dynamics in location ℓ^1 to ℓ^4 are given as follows:

For location ℓ^1,

$$\begin{cases} \dot{x}_1 = x_3,\ \dot{x}_2 = x_4 \\ \dot{x}_3 = 31.42x_5 - 1.57x_3 - 34.56(0.033 + 0.032\cos(x_1 \\ \qquad -x_2) + 0.44\sin(x_1 - x_2) + 0.019\cos(x_1) \\ \qquad +0.23\sin(x_1)) \\ \dot{x}_4 = 31.42x_6 - 1.57x_4 - 34.56(0.032\cos(x_2 - x_1) + \\ \qquad 0.44\sin(x_2 - x_1) + 0.039 + 0.025\cos(x_2) + \\ \qquad 0.52\sin(x_2)) \\ \dot{x}_5 = 0,\ \dot{x}_6 = 0 \\ \dot{x}_7 = 16((1.2971 + 1.1342\cos(x_1 - x_2) + 0.61\cos(x_1) \\ \qquad +0.4766\cos(x_2))/1.05/1.05 - 1) \\ \dot{x}_8 = 0,\ \dot{x}_9 = 0,\ \dot{x}_{10} = 0. \end{cases} \tag{16}$$

For location ℓ^2,

$$\begin{cases} \dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{x}_4, \dot{x}_5, \dot{x}_6 \text{ are the same as in } \ell^1 \\ \dot{x}_7 = 0,\ \dot{x}_8 = 10,\ \dot{x}_9 = 0,\ \dot{x}_{10} = 0. \end{cases} \tag{17}$$

For location ℓ^3,

$$\begin{cases} \dot{x}_1 = x_3,\ \dot{x}_2 = x_4 \\ \dot{x}_3 = 31.42x_5 - 1.57x_3 - 34.56(0.27 + 0.21\cos(x_1 \\ \qquad -x_2) + 1.05\sin(x_1 - x_2) + 0.12\cos(x_1) \\ \qquad +0.55\sin(x_1)) \\ \dot{x}_4 = 31.42x_6 - 1.57x_4 - 34.56(0.21\cos(x_2 - x_1) \\ \qquad +1.05\sin(x_2 - x_1) + 0.17 + 0.098\cos(x_2) \\ \qquad +0.74\sin(x_2)) \\ \dot{x}_5 = \dot{x}_6 = \dot{x}_7{}^1 = \dot{x}_8 = \dot{x}_9 = \dot{x}_{10} = 0. \end{cases} \tag{18}$$

[1]Note that x_7 ($r_{\mathrm{x1},1}$) actually gradually decreases to 0 after CB11(12) opens, but as x_7 ($r_{\mathrm{x1},1}$) is not relevant any more after CB11(12) opens, we decide to simplify it by assuming that x_7 ($r_{\mathrm{x1},1}$) stays the same after CB11(12) opens.

Algorithm 3: Robust testing algorithm for generator power dispatch schedules.

```
flag ← 0
while flag = 0 do
    Simulate ψ and ρ_ψ of H from the initial point x^1
    if ρ_ψ is safe with respect to φ_s^4 then
        γ^1[0] ← RobustTube(H, ψ, ρ_ψ, r_0, a_s, T_lag, μ)
        if γ^1[0] ≥ η then flag ← 1
        else
            x_5^1 ← x_5^1 − h, x_6^1 ← x_6^1 + h
        end if
    else
        x_5^1 ← x_5^1 − h, x_6^1 ← x_6^1 + h
    end if
end while
return x^1, γ^1[0]
```

For location ℓ^4,

$$\begin{cases} \dot{x}_1, \dot{x}_2, \dot{x}_3, \dot{x}_4, \dot{x}_5, \dot{x}_6, \dot{x}_7, \dot{x}_8 \text{ are the same as in } \ell^3 \\ \dot{x}_9 = 16((1.3 + 0.92\cos(x_1 - x_2 - 2.041) + 0.6\cos(x_1 \\ \qquad -2.1666) + 0.2762\cos(x_2 - 0.1256))/1.05/1.05 - 1) \\ \dot{x}_{10} = 0. \end{cases} \tag{19}$$

The invariant sets are $\mathrm{Inv}(\ell^1) = \{x | x_7 \le 1\}$, $\mathrm{Inv}(\ell^2) = \{x | x_8 \le 1, 1.2971 + 1.1342\cos(x_1 - x_2) + 0.61\cos(x_1) + 0.4766\cos(x_2) > 1.05^2\}$, $\mathrm{Inv}(\ell^3) = \{x | 1.3 + 0.92\cos(x_1 - x_2 - 2.041) + 0.6\cos(x_1 - 2.1666) + 0.2762\cos(x_2 - 0.1256) \le 1.05^2\}$, and $\mathrm{Inv}(\ell^4) = \mathbb{R}^{10}$. Corresponding to boundaries of invariant sets, four events are modeled: $e^2 = (\ell^1, \ell^2, g^2, r^2)$, $g^2 = \{x | x_7 = 1\}$, $r^2(x) = x$; $e^3 = (\ell^2, \ell^3, g^3, r^3)$, $g^3 = \{x | x_8 = 1\}$, $r^3(x) = x$; $e^4 = (\ell^3, \ell^4, g^4, r^4)$, $g^4 = \{x | 1.3 + 0.92\cos(x_1 - x_2 - 2.041) + 0.6\cos(x_1 - 2.1666) + 0.2762\cos(x_2 - 0.1256) = 1.05^2\}$, $r^4(x) = x$.

We express the safety criterion in the following STL form:

$$\varphi_s^4 = \neg\Diamond_{[t^4[0], T_{\mathrm{end}}]}(r_{\mathrm{x1},2} > 1) \tag{20}$$

which reads as "During the time period $[t^4[0], T_{\mathrm{end}}]$, $r_{\mathrm{x1},2}$ should never be greater than 1." We say that a simulated (nominal) trajectory ρ_ψ is safe with respect to φ_s^4 if ρ_ψ satisfies φ_s^4 (for details on satisfaction of trajectories with respect to STL formulas, see [27]).

The robust testing algorithm for the existing generator power dispatch schedules is shown in Algorithm 3. We modified MATLAB Toolbox STRONG [9] to incorporate the nonlinear dynamics. The error bounds of the simulations are obtained from the simulation library VNODE-LP. We first simulate a nominal trajectory ρ_ψ and compute $\gamma^1[0]$ using Algorithm 2. We set a lower threshold η to avoid too small robust neighborhood. If ρ_ψ is not safe or ρ_ψ is safe but $\gamma^1[0]$ is too small, then we move the initial point x^1 along the line $P_{\mathrm{G1}} + P_{\mathrm{G2}} = P_{\mathrm{D}}$ (equality constraint for power system economic dispatch, where P_{D} is the total active power demand) for stepsize h and verify if the new trajectory starting from the new initial point is safe. We set $r_0 = 0.1, a_s = 0.75, T_{\mathrm{lag}} = 0.01s, \mu = 0.002\%, \epsilon =$

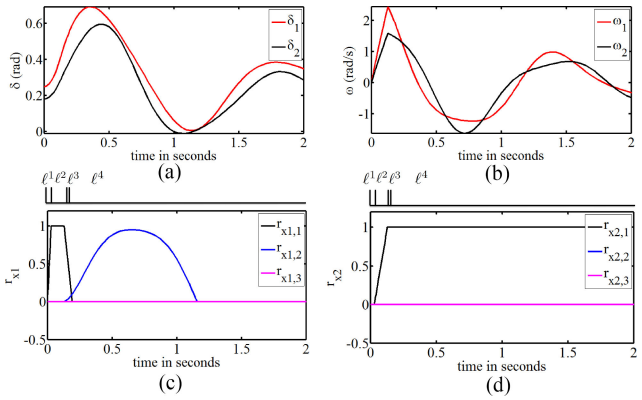


Fig. 6. Swing dynamics and protection operation in the three-machine power system model when $P_{G1} = 0.88$ and $P_{G2} = 0.62$.

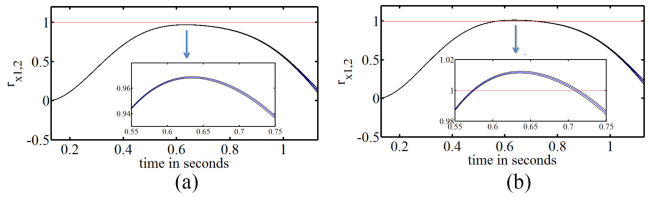


Fig. 7. Simulation of $r_{x1,2}$ with robustness bounds at location ℓ^4 with $\gamma_1^0 = 0.000032$ and (a) $P_{G1} = 0.88$, $P_{G2} = 0.62$ and (b) $P_{G1} = 0.885$, $P_{G2} = 0.615$.

$10^{-6}, h = 0.005, \eta = 0.00001,$ and $T_{\text{end}} = 0.4$ s. When $P_{G1} = 0.885$ and $P_{G2} = 0.615$, the nominal trajectory violates φ_s^4; when $P_{G1} = 0.88$ and $P_{G2} = 0.62$, both the nominal trajectory and all the other trajectories in the robust neighborhood satisfy φ_s^4 (as shown in Figs. 6 and 7). By setting the elements in the M matrix that are not related to P_{G1} and P_{G2} to be 0, we have

$$(P_{G1} - 0.88, P_{G2} - 0.62)\begin{bmatrix} M_{55} & M_{56} \\ M_{65} & M_{66} \end{bmatrix}$$
$$(P_{G1} - 0.88, P_{G2} - 0.62)^T \leq 0.000032^2 \qquad (21)$$

where M_{ij} is the (i,j)th entry of the matrix M, specifically, $M_{55} = 22.1684, M_{56} = 12.94, M_{65} = 12.94, M_{66} = 20.4193$. The above-obtained inequality describes the robust neighborhood of P_{G1} and P_{G2} assuming all other state variables $x_1, x_2, x_3, x_4, x_7, x_8, x_9$ and x_{10} remain the same. Considering that $P_{G1} + P_{G2} = 1.5$, it is verified that if P_{G1} is 0.88 ± 0.00001 (1760 MW $\pm$ 0.02 MW), P_{G2} is 0.62 ∓ 0.00001 (1240 MW $\mp$ 0.02 MW), the cascading failure would not happen.

Method 2: In the hybrid system H, we simulate a nominal trajectory ρ_ψ that starts from ℓ^1 and we treat the guard set between location ℓ^3 and ℓ^4 as the unsafe set π^3, which cannot be entered for more than 0.1 s. In this way, we can save the computation burden of $r_{x1,2}$ and $r_{x2,2}$ and define the state $x = [\delta_1, \delta_2, \omega_1, \omega_2, P_{G1}, P_{G2}, r_{x1,1}, r_{x2,1}]^T$. We express the safety criterion as $\varphi^3 = \neg\Diamond_{[t^3[0], T_{\text{end}}]}(\Box_{[0,0.1]} I_2 > 1.05)$, which reads as "During the time period $[t^3[0], T_{\text{end}}]$, I_2 should not be greater than 1.05 for over 0.1 s." It is a conservative safety criterion as CB21(22) takes 0.1 s plus the relay's response time to open.

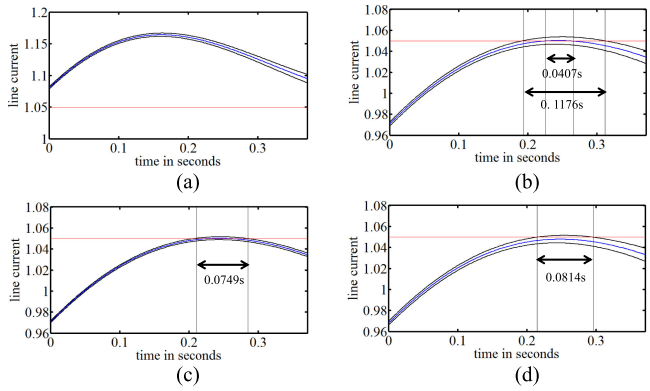


Fig. 8. Simulation of the line current I_2 with robustness bounds at location ℓ^3 with (a) $P_{G1} = 0.9$, $P_{G2} = 0.6$, $\gamma_1^0 = 0.0011$, (b) $P_{G1} = 0.695$, $P_{G2} = 0.805$, $\gamma_1^0 = 0.0011$, (c) $P_{G1} = 0.695$, $P_{G2} = 0.805$, $\gamma_1^0 = 0.00056$, and (d) $P_{G1} = 0.69$, $P_{G2} = 0.81$, $\gamma_1^0 = 0.0011$.

After the fault is cleared, $I_2(t)$ can be expressed as follows:

$$I_2(t)^2 = 1.3 + 0.92\cos(\delta_1(t) - \delta_2(t) - 2.041) + 0.6\cos(\delta_1(t) - 2.1666) + 0.2762\cos(\delta_2(t) - 0.1256). \qquad (22)$$

It is clear that the unsafe set is nonconvex, and finding the global minimal distance between a point in the unsafe set and a given point of the nominal trajectory is hard. However, as $I_2(t)^2$ can be represented as the sum of sinusoidal terms as (7), the upper and lower bounds of $I_2(t)^2$ in the current guess of robust neighborhood of the nominal trajectory can be calculated in a similar manner as the error term is bounded in Section III-B.

We modify Algorithm 3 by replacing φ_s^4 with φ^3, and we use the same parameters of Method 1. When $P_{G1} = 0.9$ and $P_{G2} = 0.6$, as shown in Fig. 8(a), the simulated trajectory clearly violates φ^3. When $P_{G1} = 0.695$ and $P_{G2} = 0.805$, as shown in Fig. 8(b), the simulated trajectory satisfies φ^3, but not every trajectory in the current guess of robust neighborhood (with $\gamma_1^0 = 0.0011$) are safe as the upper bound current exceeds I_{TAP} for 0.1176 s. When $\gamma^1[0]$ is shrunk to 0.0001, as shown in Fig. 8(c), the robust neighborhood is found as the upper bound current exceeds I_{TAP} for only 0.0749 s. Therefore, it is verified that if P_{G1} is 0.695 ± 0.000136 (1390 MW $\pm$ 0.272 MW), P_{G2} is 0.805 ∓ 0.000136 (1610 MW $\mp$ 0.272 MW), then the cascading failure would not happen. To make $\gamma^1[0]$ larger, we set $\eta = 0.001$. This time, Algorithm 3 does not terminate until P_{G1} moves to 0.69 and P_{G2} moves to 0.81, as shown in Fig. 8(d). Similarly, it is verified that if P_{G1} is 0.69 ± 0.00027 (1380 MW $\pm$ 0.54 MW), P_{G2} is 0.81 ∓ 0.00027 (1620 MW $\mp$ 0.54 MW), the cascading failure would not happen.

In comparison with Method 1, Method 2 chieves a more conservative result, but the computational complexity is also reduced by not incorporating the relay and circuit breaker dynamics of the second line (line 6-7).

2) Robust Testing of Remedial Action Based on Quick-start Storage: Various storages are increasingly utilized to generate or store electricity after a line trip to reduce imbalance within a short time. When the storage starts, the classical model described

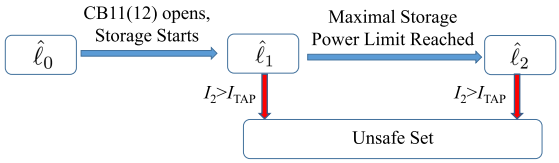


Fig. 9. Locations of the hybrid system $\hat{H}$ with an unsafe set.

in (11) is modified as follows:

$$\begin{cases} \dot{\delta}_i = \omega_i \\ \dfrac{P_{\text{ri}}}{P_{\text{b}}}\dfrac{H_i}{\pi f_{\text{s}}}\dot{\omega}_i = P_{\text{G}i} + P_{\text{ESS}i} - D_i\omega_i - p_{\text{e}i}(\delta_1, \delta_2) \end{cases} \tag{23}$$

where $P_{\text{ESS}i}$ is the power flowing out of the storage.

We assume that the storage is always connected to the grid and $P_{\text{ESS}i}$ cannot change instantaneously. We denote $dP_{\text{ESS}i}$ as the starting rate of the storage (ramping up or down) for the ith generator. In order to still balance the supply and demand, we have $\sum_{i=1}^{2} dP_{\text{ESS}i} = 0$. In order to prevent CB21(22) from opening after CB11(12) opens, we need to decrease the active power flowing into the grid from bus 4. Thus, we specify $dP_{\text{ESS1}} < 0$ (the active power flowing into the storage ESS1 is increasing) and correspondingly $dP_{\text{ESS2}} > 0$ (the active power flowing out of the storage ESS2 is increasing). We set a maximal power limit of the storage as P_{lim} and $dP_{\text{ESS}i}$ are set to zero once P_{lim} is reached.

We construct a new hybrid system $\hat{H}$ with three different locations, as shown in Fig. 9. The new state variable $y = [\delta_1, \delta_2, \omega_1, \omega_2, P_{\text{ESS1}}, P_{\text{ESS2}}, dP_{\text{ESS1}}, dP_{\text{ESS2}}]$. We specify that the remedial action happens after CB11(12) opens and y^1 (the initial point of location $\hat{\ell}^1$) is the initial point of the nominal (simulated) trajectory. The continuous dynamics of $\hat{\ell}^1$ and $\hat{\ell}^2$ are given as follows:

For location $\hat{\ell}^1$,

$$\begin{cases} \dot{y}_1 = y_3,\ \dot{y}_2 = y_4 \\ \dot{y}_3 = 31.42(0.71 + y_5) - 1.57y_3 - 34.56(0.28 \\ \quad +0.21\cos(y_1 - y_2) + 1.05\sin(y_1 - y_2) + 0.12\cos(y_1) \\ \quad +0.55\sin(y_1)) \\ \dot{y}_4 = 31.42(0.79 + y_6) - 1.57y_4 - 34.56(0.21\cos(y_2 \\ \quad -y_1) + 1.05\sin(y_2 - y_1) + 0.35 + 0.098\cos(y_2) \\ \quad +0.74\sin(y_2)) \\ \dot{y}_5 = y_7,\ \dot{y}_6 = y_8,\ \dot{y}_7 = \dot{y}_8 = 0. \end{cases} \tag{24}$$

For location $\hat{\ell}^2$,

$$\begin{cases} \dot{y}_1, \dot{y}_2, \dot{y}_3, \dot{y}_4, \dot{y}_7, \dot{y}_8 \text{ are the same as in } \hat{\ell}^1 \\ \dot{y}_5 = \dot{y}_6 = 0. \end{cases} \tag{25}$$

If the nominal (simulated) trajectory stops at $\hat{\ell}^1$, we use the safety criterion $\hat{\varphi}^1$ in the form of (9) with the time horizon $[t^1[0], t^1[0] + T^1 + T_{\text{lag}}]$ replaced by $[t^1[0], T_{\text{end}}]$. If the nominal (simulated) trajectory stops at $\hat{\ell}^2$, in order to prevent the situation where I_2 is over I_{TAP} for over 0.1 s during the transition period at guard $\hat{g}^2 = \{y | y_5 = P_{\text{lim}}\}$, we modify Algorithm 2 by checking

Algorithm 4: Robust testing algorithm for remedial actions based on quick-start storage.

```
flag ← 0
while flag = 0 do
    Simulate ψ̂ and ρ̂_ψ of Ĥ from the initial point y^1
    if ρ̂_ψ is safe with respect to φ̂^i (i = 1, 2) then
        γ̂^1[0] ← RobustTube(Ĥ, ψ̂, ρ̂_ψ, r_0, a_s, T_lag, μ)
        if γ̂^1[0] ≥ η then flag ← 1
        else
            y_7^1 ← y_7^1 − h, y_8^1 ← y_8^1 + h
        end if
    else
        y_7^1 ← y_7^1 − h, y_8^1 ← y_8^1 + h
    end if
end while
return y^1, γ̂^1[0]
```

if the upper bound of I_2 is over I_{TAP} for $[t^2[0], t^2[0] + T'_{\text{d}}](0 \le T'_d \le 0.1)$. If the answer is no, we use $\hat{\varphi}^1$ in the form of (9); otherwise, $\hat{\varphi}^1$ is modified as the following STL form:

$$\begin{aligned} \hat{\varphi}^1 = \neg\Big(& \big(\Diamond_{[t^1[0],\, t^1[0]+T^1+T_{\text{lag}}]}(\Box_{[0,0.1]} I_2 > 1.05)\big) \vee \\ & \big(\Diamond_{[t^1[0]+T^1-T_{\text{lead}}-(0.1-T'_{\text{d}}),\, t^1[0]+T^1+T_{\text{lag}}]} \\ & (\Box_{[0,0.1-T'_{\text{d}}]} I_2 > 1.05)\big)\Big) \end{aligned} \tag{26}$$

where T_{lead} is the earliest time that any trajectory in the robust neighborhood could reach guard $\hat{g}^2$ (T_{lead} can be calculated or specified using the method in [8]). In both cases, the safety criterion $\hat{\varphi}^2$ for location $\hat{\ell}^2$ is in the form of (9) with the time horizon $[t^2[0], t^2[0] + T^2 + T_{\text{lag}}]$ replaced by $[t^2[0], T_{\text{end}}]$.

The robust testing algorithm for remedial generator power dispatch based on quick-start storage is shown in Algorithm 4. We set $h = 0.1, \eta = 0.00001, P_{\text{lim}} = 0.05$, and $T_{\text{end}} = 0.4$ s. We perform robust testing using storage as remedial actions when $P_{\text{G1}} = 0.71$ (1420 MW) and $P_{\text{G2}} = 0.79$ (1580 MW). As can be seen from Fig. 10, the simulated trajectory for storage starting rate of 0 (no storage starting) is not safe as I_2 is exceeding I_{TAP} for more than 0.1 s. When $-dP_{\text{ESS1}} = dP_{\text{ESS2}} = 0.1$, the nominal (simulated) trajectory is safe, but not every trajectory in the current guess of robust neighborhood (with $\hat{\gamma}^1[0] = 0.00075$) is safe as the upper bound current exceeds I_{TAP} for 0.1299 s. Then, $\gamma^1[0]$ shrinks to 0.000075 and the robust neighborhood is found.

We denote $\hat{V}$ as the transformation matrix in (5) for the remedial action scenario and $\hat{M} = \hat{V}^T\hat{V}$. By setting the elements in the $\hat{M}$ matrix that are not related to dP_{ESS1} and dP_{ESS2} to be 0, we have

$$\begin{aligned} & (dP_{\text{ESS}} + 0.1, dP_{\text{ESS2}} - 0.1) \begin{bmatrix} \hat{M}_{77} & \hat{M}_{78} \\ \hat{M}_{87} & \hat{M}_{88} \end{bmatrix} \\ & (dP_{\text{ESS1}} + 0.1, dP_{\text{ESS2}} - 0.1)^T \le 0.000075^2 \end{aligned} \tag{27}$$

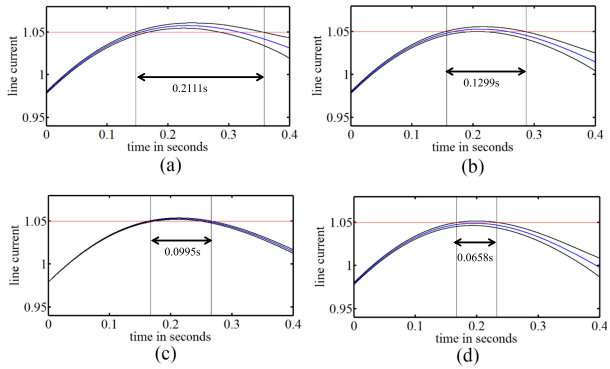


Fig. 10. Simulation of the line current I_2 with robustness bounds for different storage starting rates: (a) $dP_{\text{ESS1}} = dP_{\text{ESS2}} = 0$, $\hat{\gamma}^1[0] = 0.00075$, (b) $-dP_{\text{ESS1}} = dP_{\text{ESS2}} = 0.1$, $\hat{\gamma}^1[0] = 0.00075$, (c) $-dP_{\text{ESS1}} = dP_{\text{ESS2}} = 0.1, \hat{\gamma}^1[0] = 0.000075$, and (d) $-dP_{\text{ESS1}} = dP_{\text{ESS2}} = 0.2, \hat{\gamma}^1[0] = 0.00075$.

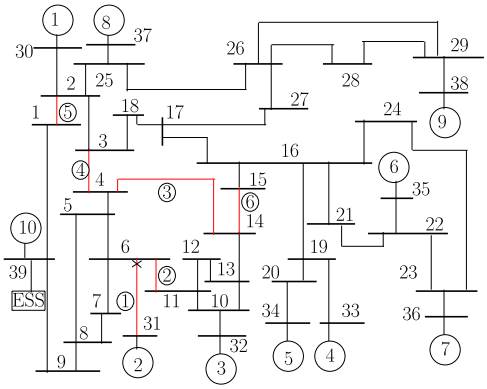


Fig. 11. IEEE 39-Bus System (the circled numbers next to the red lines are the orders in which the red lines are tripped after the fault near bus 6).

where $\hat{M}_{77} = 2.4416, \hat{M}_{78} = 1.2296, \hat{M}_{87} = 1.2296$, and $\hat{M}_{88} = 2.2795$. The above-obtained inequality describes the robust neighborhood of dP_{ESS1} and dP_{ESS2}, assuming all other state variables $y_i (i = 1, 2, \ldots, 6)$ remain the same. Therefore, it is verified that when $P_{\text{G1}} = 0.71$ (1420 MW) and $P_{\text{G2}} = 0.79$ (1580 MW), if $dP_{\text{ESS1}} = -0.1 \pm 0.00007 (-200$ MW/s $\pm$ 0.14 MW/s) and $dP_{\text{ESS2}} = 0.1 \mp 0.00007$ (200 MW/s $\mp$ 0.14 MW/s), then the cascading failure would not happen. To make $\hat{\gamma}^1[0]$ larger, we set $\eta = 0.0005$, and it is verified that if $dP_{\text{ESS1}} = -0.2 \pm 0.00052$ (-400 MW/s $\pm$ 1.04 MW/s) and $dP_{\text{ESS2}} = 0.2 \mp 0.00052$ (400 MW/s $\mp$ 1.04 MW/s), then the cascading failure would not happen.

B. IEEE 39-Bus System

In this section, we perform robust testing of the IEEE 39-Bus System (as shown in Fig. 11) in terms of mitigating cascading failures. We use the same relay and circuit breaker parameters as those in Section IV-A except that we set $I_{\text{TAP}} = 8.5$ for the lines that are connected directly to a generator bus (such as bus 6-31) and $I_{\text{TAP}} = 5.5$ for the other lines. We assume that a three-phase fault happens on line 6-31 near bus 6 and line 6-31 is tripped as a result. The tripping of line 6-31 leads to the isolation of generator G_2, and line 6-11 which connects the generators in the right part and the loads at bus 7,8,4 is suddenly

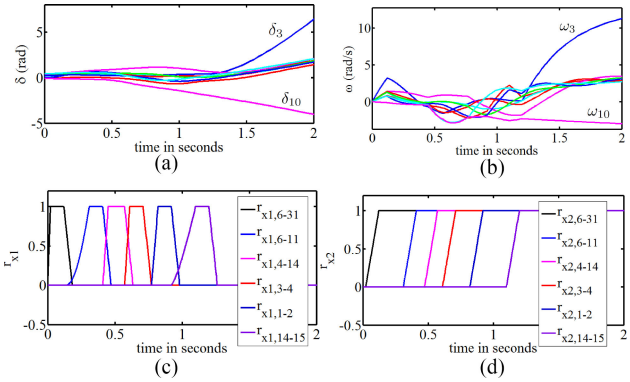


Fig. 12. Swing dynamics and protection operation in the IEEE 39-bus system model when $P_{\text{G3}} = 6.5$ (per unit value on a 100-MVA base) and $P_{\text{G10}} = 10$.

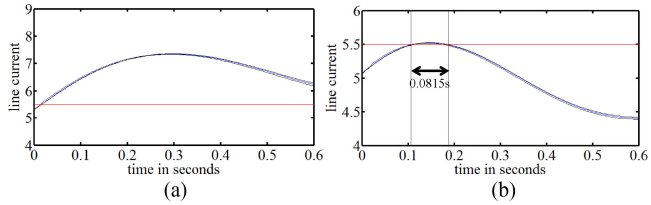
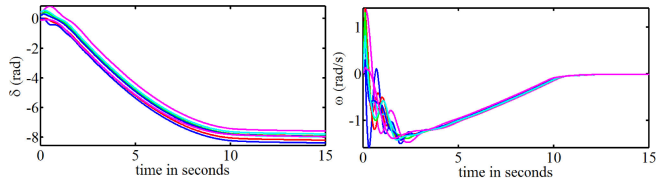


Fig. 13. Simulation of the line 6-11 current with robustness bounds at location ℓ^3 with $\gamma_1^0 = 0.00002$ and (a) $P_{\text{G3}} = 6.5$, $P_{\text{G10}} = 10$, and (b) $P_{\text{G3}} = 3.6$, $P_{\text{G10}} = 12.9$.

overloaded. When line 6-11 is tripped by the relay, line 4-14 is suddenly overloaded and tripped as it is one of the bridges that connect the left and right part of the remaining system. As shown in Fig. 12, lines 3-4, 1-2, and 14-15 will be tripped consecutively after the tripping of line 4-14 and the system soon becomes unstable. The locations of the hybrid system are similar to those in Fig. 4 except that we have 19 different locations corresponding to 6 line trippings. We use $r_{\text{x1},a\text{-}b}$ and $r_{\text{x2},a\text{-}b}$ to denote the internal state of the relay and the circuit breaker of line a-b. As G_3 at bus 32 and G_{10} at bus 39 are the two closely related generators for the tripping of line 6-11, we first test different power dispatch schedules of the two generators G_3 and G_{10} in terms of preventing the tripping of line 6-11 (which is essential for averting the cascading failure). The state $x = [\delta_1, \delta_2, \ldots, \delta_{10}, \omega_1, \omega_2, \ldots, \omega_{10}, P_{\text{G3}}, P_{\text{G10}}, r_{\text{x1,6-31}}, r_{\text{x2,6-31}}]^T$.

We set the safety criterion as $\varphi^3 = \neg\Diamond_{[t^3[0], T_{\text{end}}]}(\Box_{[0,0.1]} I_{\text{6-11}} > 5.5)$. Using a similar algorithm to Algorithm 3, it can be verified that if P_{G3} is 3.6 ± 0.00001 (360 MW $\pm$ 1 kW), P_{G10} is 12.9 ∓ 0.00001 (1290 MW $\mp$ 1 kW), then line 6-11 would not be tripped (as shown in Fig. 13). It should be, however, noted that although the tripping of line 6-11 is avoided, the supply and demand are still not balanced after line 6-31 is tripped (the generator at bus 31 is isolated). Therefore, we set a quick-start storage near bus 39 and after line 6-31 is tripped, the active power starts flowing from the storage to the grid and keeps increasing until the supply and demand are balanced. Using a similar algorithm to Algorithm 4, it can be verified that if $dP_{\text{ESS10}} = 1 \pm 0.00002$ (100 MW/s $\pm$ 2 kW/s), then line 6-11 would not be tripped and the cascading failure would not happen. As shown in Fig. 14, the machine angles first decrease and then gradually become steady with the rotor speeds returning to the



(a) (b)

Fig. 14. Machine angles and rotor speed deviations when $P_{G3} = 3.6$, $P_{G10} = 12.9$, and $dP_{ESS10} = 1$.

normal value. The robust testing for each simulation takes about 5 s on a Dell desktop computer with a 3.20-GHz Intel Xeon CPU and 8-GB RAM.

V. Conclusion

We presented a robust testing method for cascading failure mitigations based on power dispatch schedules and quick-start storage. We modified the approach in [13] to calculate the BDLDF, which bounds the growth of differences between trajectories considering the effects of disturbances. We modified the simulation model in [25] to model the 2003 Italian Blackout as a hybrid system, and it is different from their model as we consider not only different network topologies but also different relay states in defining different locations of the hybrid system. We also used the IEEE 39-bus system to test the scalability of our approach. Besides robust testing, our method can also provide a measure of system stability robustness in mitigating cascading failures, and this measure can be utilized in the economic dispatch to incorporate the stability aspect of the power dispatch.

Appendix

Proof of Theorem 1: The proof of uniform continuity of $\gamma(\tilde{x}^0, x^0, t)$ is not shown here as it is very similar with the proof of uniform continuity of $\beta(x_1, x_2, t)$ in [13, proof of Th. 3.9]. As $\tilde{S}[k]$ is an overapproximation of S in [13], utilizing [13, Lemma 3.6], it can be easily proven that $\tilde{J}_f(x) + \tilde{J}_f^T(x) \preceq 2\tilde{b}[k]I$ for any $x \in \tilde{S}[k]$, then from (5), for $t \in [t[k-1], t[k]]$,

$$\begin{aligned}\frac{d\left\|v(t)\right\|^2}{dt} &\le v^T(t)\left(\int_0^1 (2\tilde{b}[k]I)ds\right)v(t) + 2\left\|\zeta(t)\right\|\left\|V\right\|\left\|v(t)\right\| \\ &= 2\tilde{b}[k]\left\|v(t)\right\|^2 + 2\left\|\zeta(t)\right\|\left\|V\right\|\left\|v(t)\right\| \\ &\le 2\tilde{b}[k]\left\|v(t)\right\|^2 + 2\mu\left\|V\right\|\left\|v(t)\right\|. \end{aligned} \tag{28}$$

As the solution of the differential equation $\frac{du(t)}{dt} = 2bu(t) + 2a\sqrt{u(t)}$ with $u(0) = \|v(t[k-1])\|^2$ is $u(t) = (\sqrt{u(0)}e^{bt} + a/b(e^{bt} - 1))^2$, according to the comparison principle for the solutions of first-order ordinary differential equations [28], and considering the simulation error at time $t[k-1]$, we have for $t \in [t[k-1], t[k]]$,

$$\begin{aligned}\left\|v(t)\right\| \le (\left\|v(t[k-1])\right\| + \epsilon)e^{\tilde{b}[k](t-t[k-1])} \\ + \mu\left\|V\right\|/\tilde{b}[k](e^{\tilde{b}[k](t-t[k-1])} - 1).\end{aligned} \tag{29}$$

We use induction to prove $\phi_M(\hat{\xi}(t, \tilde{x}^0; \zeta), \xi(t, x^0)) \le \gamma(\tilde{x}^0, x^0, t)$. As $\phi_M(\hat{\xi}(t, \tilde{x}^0; \zeta), \xi(t, x^0)) = \|v(t)\|$, we only need to prove $\|v(t)\| \le \gamma(\tilde{x}^0, x^0, t)$. At $t[0]$, $\|v(t[0])\| = \phi_M(\hat{\xi}(t[0], \tilde{x}^0; \zeta), \xi(t[0], x^0)) = \gamma[0]$. By setting $k = 1$ in (29), for $t \in (t[0], t[1]]$, we have $\|v(t)\| \le (\gamma[0] + \epsilon)e^{\tilde{b}[k](t-t[0])} + \mu\|V\|/\tilde{b}[k](e^{\tilde{b}[k](t-t[0])} - 1) = \gamma(\tilde{x}^0, x^0, t)$.

Next, we show that if for $t \in (t[k-2], t[k-1]]$, $\|v(t)\| \le \gamma(\tilde{x}^0, x^0, t)$, then for $t \in (t[k-1], t[k]]$, $\|v(t)\| \le \gamma(\tilde{x}^0, x^0, t)$. To this end, we assume for $t \in (t[k-2], t[k-1]]$, $\|v(t)\| \le \gamma(\tilde{x}^0, x^0, t)$, then $\|v(t[k-1])\| \le \gamma[k-1]$; then, using (29), for $t \in (t[k-1], t[k]]$,

$$\begin{aligned}\left\|v(t)\right\| \le (\gamma[k-1] + \epsilon)e^{\tilde{b}[k](t-t[k-1])} \\ + \mu\left\|V\right\|/\tilde{b}[k](e^{\tilde{b}[k](t-t[k-1])} - 1) = \gamma(\tilde{x}^0, x^0, t).\end{aligned} \tag{30}$$

Thus, by induction, it is proven that $\gamma(\tilde{x}^0, x^0, t)$ is a BDLDF of the system.

Proof of Theorem 2:

$$\begin{aligned}\left\|E(x)\right\| &= \left\|\tilde{J}_f(x) + \tilde{J}_f^T(x) - \tilde{J} - \tilde{J}^T\right\| \\ &= \left\|(V^{-1})^T(J_f^T(x) - J^T)V^T + V(J_f(x) - J)V^{-1}\right\| \\ &\le \left\|(V^{-1})^T(J_f^T(x) - J^T)V^T\right\| + \left\|V(J_f(x) - J)V^{-1}\right\| \\ &\le \left\|(V^{-1})^T\right\|\left\|J_f^T(x) - J^T\right\|\left\|V^T\right\| \\ &\quad + \left\|V\right\|\left\|J_f(x) - J\right\|\left\|V^{-1}\right\| \\ &= 2\left\|V^{-1}\right\|\left\|J_f(x) - J\right\|\left\|V\right\| = 2\mathrm{cond}(V)\left\|J_f(x) - J\right\| \\ &\le 2\mathrm{cond}(V)\left\|J_f(x) - J\right\|_F. \end{aligned} \tag{31}$$

References

[1] A. Berizzi, "The Italian 2003 blackout," in *Proc. IEEE Power Eng. Soc. Gen. Meeting*, Jun. 2004, vol. 2, pp. 1673–1679.

[2] R. Yao, S. Huang, K. Sun, F. Liu, X. Zhang, and S. Mei, "A multi-timescale quasi-dynamic model for simulation of cascading outages," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 3189–3201, Jul. 2016.

[3] Z. Huang, C. Wang, M. Stojmenovi, and A. Nayak, "Characterization of cascading failures in interdependent cyber-physical systems," *IEEE Trans. Comput.*, vol. 64, no. 8, pp. 2158–2168, Aug. 2015.

[4] M. Khederzadeh and A. Beiranvand, "Identification and prevention of cascading failures in autonomous microgrid," *IEEE Syst. J.*, vol. 12, no. 1, pp. 308–315, Mar. 2018.

[5] T. Chaiyatham and I. Ngamroo, "Improvement of power system transient stability by PV farm with fuzzy gain scheduling of PID controller," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1684–1691, Sep. 2017.

[6] P. Varaiya, F. Wu, and R.-L. Chen, "Direct methods for transient stability analysis of power systems: Recent results," *Proc. IEEE*, vol. 73, no. 12, pp. 1703–1715, Dec. 1985.

[7] Y. Susuki *et al.*, "A hybrid system approach to the analysis and design of power grid dynamic performance," *Proc. IEEE*, vol. 100, no. 1, pp. 225–239, Jan. 2012.

[8] A. A. Julius, G. E. Fainekos, M. Anand, I. Lee, and G. J. Pappas, "Robust test generation and coverage for hybrid systems," in *Hybrid Systems: Computation and Control*. Berlin, Germany: Springer, 2007, pp. 329–342.

[9] Y. Deng, A. Rajhans, and A. A. Julius, "Strong: A trajectory-based verification toolbox for hybrid systems," in *Quantitative Evaluation of Systems* (ser. Lecture Notes Comput. Sci.), vol. 8054. Berlin, Germany: Springer, 2013, pp. 165–168.

[10] A. A. Julius and G. J. Pappas, "Trajectory based verification using local finite-time invariance," in *Hybrid Systems: Computation and Control*. Berlin, Germany: Springer-Verlag, 2009, pp. 223–236.

[11] M. Althoff, "Formal and compositional analysis of power systems using reachable sets," *IEEE Trans. Power Syst.*, vol. 29, no. 5, pp. 2270–2280, Sep. 2014.
[12] P. Duggirala, S. Mitra, and M. Viswanathan, "Verification of annotated models from executions," in *Proc. Int. Conf. Embedded Softw.*, Sep. 2013, pp. 1–10.
[13] C. Fan and S. Mitra, "Bounded verification with on-the-fly discrepancy computation," in *Proc. Int. Symp. Automat. Technol. Verificat. Analysis*, Lecture Notes in Computer Science, vol 9364, Cham, Switzerland: Springer, 2015, pp. 446–463.
[14] Z. Xu, C. Belta, and A. A. Julius, "Temporal logic inference with prior information: An application to robot arm movements," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 141–146, 2015.
[15] Z. Xu and A. A. Julius, "Census signal temporal logic inference for multiagent group behavior analysis," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 1, pp. 264–277, Jan. 2018.
[16] Z. Xu, M. Birtwistle, C. Belta, and A. Julius, "A temporal logic inference approach for model discrimination," *IEEE Life Sci. Lett.*, vol. 2, no. 3, pp. 19–22, Sep. 2016.
[17] B. Krogh, S. H. Javid, and J. H. Chow, "Multi-stage rescheduling of generation, load shedding and short-term transmission capacity for emergency state control," *IEEE Trans. Power Appar. Syst.*, vol. PAS-102, no. 5, pp. 1466–1472, May 1983.
[18] S. W. Mohod and M. V. Aware, "Micro wind power generator with battery energy storage for critical load," *IEEE Syst. J.*, vol. 6, no. 1, pp. 118–125, Mar. 2012.
[19] A. Mondal, S. Misra, and M. S. Obaidat, "Distributed home energy management system with storage in smart grid using game theory," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1857–1866, Sep. 2017.
[20] R. Alur *et al.*, "The algorithmic analysis of hybrid systems," *Theoret. Comput. Sci.*, vol. 138, pp. 3–34, 1995.
[21] Y. Deng, A. DInnocenzo, M. D. Di Benedetto, S. Di Gennaro, and A. A. Julius, "Verification of hybrid automata diagnosability with measurement uncertainty," *IEEE Trans. Autom. Control*, vol. 61, no. 4, pp. 982–993, Apr. 2016.
[22] N. S. Nedialkov, "VNODE-LP—A validated solver for initial value problems in ordinary differential equations," McMaster Univ., Hamilton, ON, Canada, 2006.
[23] *CAPD-Computer Assisted Proofs in Dynamics, a Package for Rigorous Numerics*, 2002. [Online]. Available: http://capd.wsb-nlu.edu.pl/
[24] N. S. Nedialkov, K. R. Jackson, and J. D. Pryce, "An effective high-order interval method for validating existence and uniqueness of the solution of an IVP for an ODE," *Rel. Comput.*, vol. 7, no. 6, pp. 449–465, 2001.
[25] Y. Susuki, Y. Takatsuji, and T. Hikihara, "Hybrid model for cascading outage in a power system: A numerical study," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. 92, pp. 871–879, 2009.
[26] L. Perez, A. Flechsig, and V. Venkatasubramanian, "Modeling the protective system for power system dynamic analysis," *IEEE Trans. Power Syst.*, vol. 9, no. 4, pp. 1963–1973, Nov. 1994.
[27] A. Donzé and O. Maler, "Robust satisfaction of temporal logic over real-valued signals," in *Proc. 8th Int. Conf. Form. Model. Anal. Timed Syst.*, 2010, pp. 92–106.
[28] H. K. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996. [Online]. Available: http://opac.inria.fr/record=b1091137

Zhe Xu (S'16) received the B.S. and M.S. degrees in electrical engineering from Tianjin University, Tianjin, China, in 2011 and 2014, respectively. He is currently working toward the Ph.D. degree in electrical engineering at the Rensselaer Polytechnic Institute, Troy, NY, USA.

His research interests include temporal logic, systems and control, hybrid systems, and power systems.

A. Agung Julius (M'06) received the Ph.D. degree in applied mathematics from the University of Twente, Twente, The Netherlands, in 2005.

From 2005 to 2008, he was a Postdoctoral Researcher with the University of Pennsylvania. Since 2008, he has been with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA, where he is currently an Associate Professor. His research interests include systems and control, systems biology, stochastic models in systems biology, control of biological systems, hybrid systems, and mathematical systems theory.

Joe H. Chow (F'92) received the B.S. degree from the University of Minnesota, Minneapolis, MN, USA, and the M.S. and Ph.D. degrees from the University of Illinois Urbana–Champaign, Champaign, IL, USA.

After working within the General Electric power system business in Schenectady, NY, USA, in 1987, he joined the Rensselaer Polytechnic Institute, Troy, NY, where he is a Professor with the Department of Electrical, Computer, and Systems Engineering and the Campus Director of the NSF/DOE CURENT ERC. His research interests include power system dynamics and control, renewable resources, voltage stability, and synchronized phasor data.

Dr. Chow was the recipient of the Donald Eckman Award from the American Automatic Control Council, the Control System Technology Award from the IEEE Control System Society, and the Charles Concordia Power System Engineering Award from the IEEE Power and Energy Society. He is a member of the National Academy of Engineering.