Spatio-Temporal Failure Propagation in Cyber-Physical Power Systems

Osman Boyaci
Electrical Engineering
Texas A&M University
College Station, TX, 77843
osman.boyaci@tamu.edu

M. Rasoul Narimani
College of Engineering
Arkansas State University
Jonesboro, AR, 72404
mnarimani@astate.edu

Katherine Davis
Electrical Engineering
Texas A&M University
College Station, TX, 77843
katedavis@tamu.edu

Erchin Serpedin
Electrical Engineering
Texas A&M University
College Station, TX, 77843
eserpedin@tamu.edu

Abstract—Cascading failure in power systems is triggered by a small perturbation that leads to a sequence of failures spreading through the system. The interconnection between different components in a power system causes failures to easily propagate across the system. The situation gets worse by considering the interconnection between cyber and physical layers in power systems. A plethora of work has studied the cascading failure in power systems to calculate its impact on the system. Understanding how failures propagate into the system in time and space can help the system operator to take preventive actions and upgrade the system accordingly. Due to the nonlinearity of the power flow equation as well as the engineering constraints in the power system, it is essential to understand the spatio-temporal failure propagation in cyber-physical power systems (CPPS). This paper proposes an asynchronous algorithm for investigating failure propagation in CPPS. The physics of the power system is addressed by the full AC power flow equations. Various practical constraints including load shedding, load-generation balance, and island operation are considered to address practical constraints in power system operation. The propagation of various random initial attacks of different sizes is analyzed and visualized to elaborate on the applicability of the proposed approach. Our findings shed light on the cascading failure evolution in CPPS.

I. INTRODUCTION

The interaction between different components in cyber-physical power systems (CPPS) makes these systems more vulnerable to cascading failures as failure in one component can easily propagate into other components [1]. Electric Power Systems incorporate numerous communication devices, that enable full observability, controllability, and flexibility of power system operation. Furthermore, utilizing communication components in power systems increases the interaction surface between different components and paves the way for initiating cascading failures in CPPS. Cascading failure starts from a local failure and propagates into the system by overloading other components. For instance, the U.S. 2003 blackout was initiated to a large extent by the failure that initially occurred in the information and communications technology (ICT) system [2]. Studying how the failures propagate in time and space is crucial for making the power system more resilient. A plethora of research has been dedicated to studying the impact of cascading failures on the power system. However, the dynamics of cascading failures and how

they propagate into the system are less studied. To predict and mitigate failure spreading in a CPPS, understanding their spatio-temporal propagation properties is crucial. To bridge this gap, we model the propagation of cascading failure and analyze how it propagates into the CPPS. To this end, we first randomly attack different lines and force them to be out of service. Then, we study the impact of these outages on the system.

The mechanism of failure propagation in power systems strongly depends on the physics of the power systems, which is dictated by the nonconvex nonlinear power flow equations. Numerous studies have neglected the power flow equations in studying cascading failure propagation in power systems [3]. These studies focus on the topology of the system and leverage complex network theory to model failure propagation [4]–[6]. Although methods based on the complex network theory require a much lower computational time, it is difficult to incorporate the physical mechanisms of cascading failure into the topological metrics. Sandpile model has been employed to study the cascading failure characteristics in different types of complex networks [7]. However, this model is not able to address the physics of power systems. In power grids, flows are driven by Kirchoff's laws, and cannot be described by a network flow model. Accordingly, when a failure occurs in a power grid, the power flow is redistributed on the rest of the network, and some elements could overload and fail which leads to cascading failures. Thus, without detailed power grid information, the results of the methods based on complex network theory yield differ greatly and could result in misleading conclusions about the grid vulnerability [8]. Various studies consider load redistribution of a failed node among the in-service nodes, i.e., when a node fails, the load it was carrying (right before the failure) is redistributed equally among the remaining nodes [9]. This is not the case in the power grid, where the physics of the system is governed by the nonlinear power flow equations and thus a more sophisticated model needs to address the load redistribution after a contingency. There are numerous studies that have addressed the physics of the power system in cascading failure problems via linearized power flow equations [10], [11]. Although DC power flow equations have been utilized to reasonably solve various problems in power systems, it has been shown that

cascading failure simulations that underlie the DC model (e.g., ignoring power losses, reactive power flow, and voltage magnitude variations) can lead to inaccurate and overly optimistic cascade predictions [10]. More specifically, utilizing DC power flow to study cascading failure in large networks tends to overestimate the number of in-service components after stopping the failure propagation [10] as DC power flow cannot properly capture nonlinear mechanisms like voltage collapse or dynamic instability in power systems. Hence, using the DC model for cascade prediction may result in a misrepresentation of the gravity of a cascade [10]. Multiple studies developed statistical models that use data from simulations [12], [13], or historical cascades [14] to represent the general features of cascading. Statistical models are useful but cannot replace detailed simulations to fully understand cascading mechanisms.

To address these drawbacks, our method proposes an asynchronous algorithm that periodically propagates the failures between cyber and power layers. Furthermore, the full AC power flow equations along with the engineering constraints in power grids, e.g., load-generation balance, island operation criteria, etc. have been utilized to properly address the physics of power systems. Moreover, a realistic graph generation algorithm [15] is employed to construct the underlying cyber graph in CPPS.

II. PROPAGATION OF CASCADING FAILURE

Before delving into the proposed approach for failure propagation, we need a realistic framework for cyber graph generation which is essential for modeling the failure propagation in CPPS. We recently proposed a realistic model for graph generation that mimics the features of a real-world communication system of a smart grid [15]. The model leverages the Hungarian algorithm [16] to minimize the total cross edge distance in the graph which reduces the distance between corresponding cyber and power system nodes. We applied our approach in [15] to generate realistic cyber graphs in this paper.

To model the failure propagation in cyber-physical power system, we model the interconnected CPPS with a graph. In this connection, we first consider an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ as a model of a power grid network. Here $\mathcal{V}$ is a set of nodes, and $\mathcal{E}$ is a set of edges. Similarly, the cyber layer is modeled with a graph with the same number of vertices that the corresponding power system has, i.e, $\mathcal{V}$. The topology of the cyber graph, i.e., edge position, is determined by our proposed algorithm in [15]. For sake of simplicity, we assume one connection between each node in the power systems with the corresponding node in the cyber layer. Note that the proposed model can be easily generalized to consider different typologies of the cyber graph, i.e., a cyber graph with a different number of nodes.

The proposed model is an asynchronous model in which faults propagate in only one subsystem, i.e., power or cyber graph, at a specific time step. Once the fault propagation is finished in one subsystem, it can propagate into other subsystems through the linking edges that connect cyber and power graph nodes. The fault propagation in power and cyber layers acts differently as the physics that governs these systems are totally different. For instance, in the cyber layer, after creation of islands, the largest island will be considered in-service and the other islands will be deleted from the system. Conversely, different islands in the power system can operate simultaneously as long as they satisfy the power system constraints. Algorithm 1 demonstrates the different phases of the proposed cascading failure approach where the relevant method of the class 'Grid' is called from Algorithm 2. Phase *A* of the proposed cascading failure approach models the failure propagation in the power system after a random attack. Then phase *B* reflects the impact of the topology changes in phase *A* into the cyber layer by removing the cyber nodes that are connected to the deleted nodes in the power system. Next, phase *C* propagates the failure into the cyber layer based on the node removals in Phase *B* and updates the topology of the cyber layer accordingly.

Algorithm 1: Main of Cascading Failure

Input : power network $\mathbb{PN}$, cyber network $\mathbb{CN}$, attacked buses $A_{\mathcal{V}}$, attacked branches $A_{\mathcal{E}}$
Output: binary flag indicating the blackout
1 $grid \leftarrow Grid(\mathbb{PN}, \mathbb{CN})$ // create the Grid obj.
2 grid.trigger_failure_in_power($A_{\mathcal{V}}, A_{\mathcal{E}}$)
3 **while True do**
4   grid.prop_failure_in_power()   // phase A
5   grid.prop_failure_to_cyber()   // phase B
6   **if** $grid.stopped$ **break**
7   grid.prop_failure_in_cyber()   // phase C
8   grid.prop_failure_to_power()   // phase D
9   **if** $grid.stopped$ **break**
10 $blackout \leftarrow$ grid.check_blackout()
11 **return** $blackout$

Finally, phase *D* reflects the impact of the topology changes in phase *C* into the power layer by removing the power nodes that are connected to the deleted nodes in the cyber layer. Phases *A* to *D* are continuously executed until no further removal happens in either the cyber or power layers. The cascading failure can be terminated if the number of removed nodes exceeds a threshold.

To make the proposed work more practical, the full AC power flow equations (1) are considered in this paper.

$$P_i^g - P_i^d = g_{sh,i} V_i^2 + \sum_{\substack{(l,m)\in\mathcal{L}, \\ \text{s.t. } l=i}} P_{lm} + \sum_{\substack{(l,m)\in\mathcal{L}, \\ \text{s.t. } m=i}} P_{ml}, \quad (1a)$$

$$Q_i^g - Q_i^d = -b_{sh,i} V_i^2 + \sum_{\substack{(l,m)\in\mathcal{L}, \\ \text{s.t. } l=i}} Q_{lm} + \sum_{\substack{(l,m)\in\mathcal{L}, \\ \text{s.t. } m=i}} Q_{ml}, \quad (1b)$$

$$P_{lm} = g_{lm} V_l^2 - g_{lm} V_l V_m \cos(\theta_{lm}) - b_{lm} V_l V_m \sin(\theta_{lm}), \quad (1c)$$

$$Q_{lm} = -(b_{lm} + b_{c,lm}/2) V_l^2 + b_{lm} V_l V_m \cos(\theta_{lm})$$
$$- g_{lm} V_l V_m \sin(\theta_{lm}), \quad (1d)$$

$$P_{ml} = g_{lm} V_m^2 - g_{lm} V_l V_m \cos(\theta_{lm}) + b_{lm} V_l V_m \sin(\theta_{lm}), \quad (1e)$$

$$Q_{ml} = -(b_{lm} + b_{c,lm}/2) V_m^2 + b_{lm} V_l V_m \cos(\theta_{lm})$$
$$+ g_{lm} V_l V_m \sin(\theta_{lm}), \quad (1f)$$

In eq. (1), l and m are both ends of each line $(l,m) \in \mathcal{L}$ in the system, and $g_{lm} + jb_{lm}$ and $jb_{c,lm}$ are the mutual admittance and shunt admittance in the Π model of power lines, respectively. $P_{lm} + jQ_{lm}$ denotes the complex power flow and θ_{lm} is equal to $\theta_l - \theta_m$ for each line terminal $(l,m) \in \mathcal{L}$.

Furthermore, other practical constraints including generation-load balance are considered for each island in the power system. Step-wise load shedding is also incorporated into the model to keep the islands that do not have enough generation to satisfy their loads. This makes the power system not lose an island because of a small imbalance between generation and load and consequently withstand fault propagation for a longer period of time.

The cascading failure is initially triggered by the outage of certain nodes or lines in the power system (Alg. 2, line 9-13). These line outages can change the network topology and might divide the power system network into separate components [17]. If there is enough generation in each component to satisfy the load, then the component can operate. If there is no generation in a separated component, the component will be removed by force outage its nodes and branches. Likewise, if there is no load in a separated component, the component will be removed by force outage its nodes and branches (Alg. 3, line 15-17).

For the components with both supply and demand, we first check the generation-demand balance. If the total demand was greater than the generation, the demand would be reduced at all nodes by a common factor, i.e., load shedding. Accordingly, if total generation was greater than the load, the generation would be reduced at all generation nodes by a common factor, i.e., generation curtailment (Alg. 3, line 22-23). We next check the line flows to determine the overflowed lines in the system. These lines will be removed from the system using a selected line outage rule (Alg. 3, line 8-11). The cascade continues with the removal of those lines. If there is no new line failure in any of the remaining components, the cascade ends in the power system layer (Alg. 2, line 20-21).
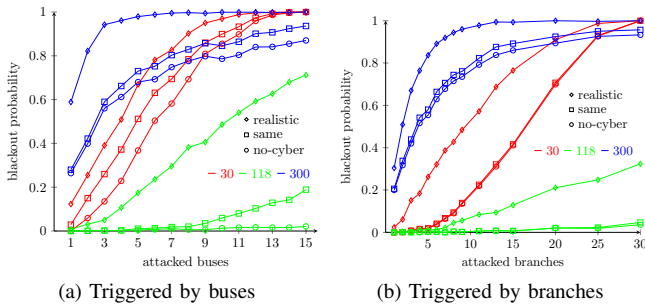

Fig. 1. Blackout probabilities of the cascading failure triggered by the number of initial attacked buses (a) and branches (b) for various systems and cyber networks. Each power network having 30- (red), 118- (green), and 300-bus (blue) simulated with three different cyber networks where there is no cyber graph (circle), cyber graph with the same topology as power system's graph (square), and cyber graph with a realistic topology (diamond), respectively. Blackout probabilities are calculated as the mean values of 1000 runs in which attacked buses or branches selected randomly.

Algorithm 2: Class Grid

```
   // Constructer method of the class
1  def Grid::Grid(PN, CN)
2      PN ← PN
3      CN ← CN
4      n, m ← |PN.V|, |PN.E|      // # of components
5      l ← ⟨PN⟩          // total active load in PN
6      V_p^f, E_p^f ← ∅, ∅    // Failed power components
7      V_c^f, E_c^f ← ∅, ∅    // Failed cyber components
8      stopped ← False        // Failure stopped
9  def Grid::trigger_failure_in_power(A_V, A_E)
10     foreach v ∈ A_V do
11         mark bus v in PN as out of service
12     foreach e ∈ A_E do
13         mark branch e in PN as out of service
   // Phase A
14 def Grid::prop_power_failure_in_power()
15     while True do
16         foreach island P in PN do
17             run_power_flow(P)
18             remove_overloaded_branches(P)
19             extract_islands(P)
20         if there is no new failure
21             break
22     V_p^f, E_p^f ← determine failed components in PN
   // Phase B
23 def Grid::prop_power_failure_to_cyber()
24     if V_p^f = ∅
25         stopped ← True
26     else
27         foreach u in V_p^f do
28             mark correspoding node v as failed in CN
   // Phase C
29 def Grid::prop_cyber_failure_in_cyber()
30     determine the connected components in CN
31     keep only the giant component of CN
32     mark other islands as failed in CN
33     V_c^f, E_c^f ← determine failed components in CN
   // Phase D
34 def Grid::prop_cyber_failure_to_P(P)
35     if V_c^f = ∅
36         stopped ← True
37     else
38         foreach u in V_c^f do
39             mark correspoding node v as failed in PN
40 def Grid::check_blackout(P)
41     if |PN.V| < n/2 or |PN.E| < m/2 or ⟨PN⟩ < l/2
42         return True
43     else
44         return False
```

III. RESULTS AND DISCUSSION

This section demonstrates the effectiveness of the proposed approach using IEEE 30-, 118-, and 300-bus test cases. All implementation was carried out in Python 3.8 on Intel i9-8950 HK CPU 2.90GHz with NVIDIA GeForce RTX 2070 GPU.

Fig. 1 demonstrates the blackout probability with respect to the number of the attacked nodes and branches for 30-, 118-, and 300-bus test cases. The blackout is defined as the loss of fifty percent of either the number of nodes, the number of branches, or the total load in the system (Alg. 2, line 40-44).

Algorithm 3: Power utility functions

```
1  def run_power_flow(ℙ)
2      foreach s ∈ {1.0, 0.95, ..., 0.05} do
3          scale loads of ℙ with s
4          result ← runpf(ℙ)
5          if result
6              mark island ℙ as succeded
7              return
8      mark island ℙ as failed
9  def remove_overloaded_branches(ℙ)
10     foreach e ∈ ℙ.ℰ do
11         if e's loading > 100%
12             mark e as out of service
13 def extract_islands(ℙ)
14     determine the connected components in ℙ
15     foreach island ∈ ℙ do
16         if ℙ has no generation or load
17             mark ℙ's component as out of service
18             continue
19         if ℙ has no slack bus
20             g ← highest generation capacity bus
21             set g as the slack bus
22         if total generation > total load
23             apply generation curtailment
```

It is clear that the probability of having blackout increases by the initial attack size as expected. There are three different plots in Fig. 1 for each test case that are corresponding to the different topology of the underlying cyber graph of the power systems. The lines with circles, squares, and diamonds in Fig. 1 represent the probability of occurring blackout in CPPS where there is no cyber graph, cyber graph with the same topology as the power system's graph, and cyber graph with a realistic topology, respectively. From Fig. 1 it is clear that the topology of the underlying cyber graph in CPPS plays an important role in the dynamics of failure propagation.

Figs. 2 and 3 visualize different phases of failure propagation after random attacks for IEEE 118- and 300-bus test cases, respectively. Comparing Fig. 3b and Fig. 3c demonstrate that the failures propagate nonlocally in power systems.

IV. CONCLUSION

To better understand the dynamics of cascading failure in cyber-physical power systems, we proposed an asynchronous algorithm to model failure propagation in cyber-physical power systems. In contrast to most previous works that employ DC power flow model for cascading failure analysis, we implement the full AC power flow equation in our model to accurately address the physics of power systems. Moreover, to make the proposed approach more general, the interdependency between power and cyber layers is taken into consideration in the proposed model. We illustrated via simulations the failure propagation in cyber-physical power systems. We showed that the failures in power systems propagate globally, i.e., the consecutive failures can be far away from each other, rather than locally. We also show that taking cyber layer impacts into account can expedite failure propagation process, as it was expected. The outcome of this paper can be used to analyze the resiliency of cyber-physical systems against failure propagation. Furthermore, the proposed model can shed light on identifying critical components and analyzing contingency analysis in cyber-physical systems. This work is under study to model the interaction between power and cyber layers via graph theory metrics. More specifically, dependability and importance of a specific node in one layer to the corresponding node in another layer will be incorporated in the proposed model to more accurately simulate the interaction between layers in cyber-physical power systems.

REFERENCES

[1] Z. Jichang, L. Daqing, S. Hillel, C. Reuven, and H. Shlomo, "Spatio-temporal propagation of cascading overload failures in spatially embedded networks," *Nature Communications*, vol. 7, no. 10094, 2016.

[2] J. V. Milanovi and W. Zhu, "Modeling of interconnected critical infrastructure systems using complex network theory," *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 4637–4648, 2018.

[3] P. Crucitti, V. Latora, and M. Marchiori, "A topological analysis of the italian electric power grid," *Physica A: Statistical Mechanics and its Applications*, vol. 338, no. 1, pp. 92–97, 2004, proceedings of the conference A Nonlinear World: the Real World, 2nd International Conference on Frontier Science. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378437104002249

[4] R. Albert, I. Albert, and G. L. Nakarado, "Structural vulnerability of the north american power grid," *Phys. Rev. E*, vol. 69, p. 025103, Feb 2004. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.69.025103

[5] P. Dey, R. Mehra, F. Kazi, S. Wagh, and N. M. Singh, "Impact of topology on the propagation of cascading failure in power grid," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 1970–1978, 2016.

[6] J. Beyza, E. Garcia-Paricio, H. F. Ruiz, and J. M. Yusta, "Geodesic vulnerability approach for identification of critical buses in power systems," *Journal of Modern Power Systems and Clean Energy*, vol. 9, no. 1, pp. 37–45, 2021.

[7] K.-I. Goh, D.-S. Lee, B. Kahng, and D. Kim, "Sandpile on scale-free networks," *Phys. Rev. Lett.*, vol. 91, p. 148701, Oct 2003. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.91.148701

[8] J. Song, E. Cotilla-Sanchez, G. Ghanavati, and P. D. H. Hines, "Dynamic modeling of cascading failure in power systems," *IEEE Transactions on Power Systems*, vol. 31, no. 3, pp. 2085–2095, 2016.

[9] S. Pahwa, C. Scoglio, and A. Scala, "Sabruptness of cascade failures in power grids," *Scientific reports*, vol. 4, no. 3694, 2014.

[10] H. Cetinay, S. Soltan, F. A. Kuipers, G. Zussman, and P. Van Mieghem, "Comparing the effects of failures in power grids under the ac and dc power flow models," *IEEE Transactions on Network Science and Engineering*, vol. 5, no. 4, pp. 301–312, 2018.

[11] R. Spiewak, S. Soltan, Y. Forman, S. V. Buldyrev, and G. Zussman, "A study of cascading failures in real and synthetic power grid topologies," *Network Science*, vol. 6, no. 4, p. 448468, 2018.

[12] M. Rahnamay-Naeini, Z. Wang, N. Ghani, A. Mammoli, and M. M. Hayat, "Stochastic analysis of cascading-failure dynamics in power grids," *IEEE Transactions on Power Systems*, vol. 29, no. 4, pp. 1767–1779, 2014.

[13] P. D. Hines, I. Dobson, E. Cotilla-Sanchez, and M. Eppstein, ""dual graph" and "random chemistry" methods for cascading failure analysis," in *2013 46th Hawaii International Conference on System Sciences*, 2013, pp. 2141–2150.

[14] I. Dobson, "Estimating the extent of cascading transmission line outages using standard utility data and a branching process," in *2011 IEEE Power and Energy Society General Meeting*, 2011, pp. 1–3.

[15] O. Boyaci, M. R. Narimani, K. Davis, and E. Serpedin, "Generating connected, simple, and realistic cyber graphs for smart grids," *arXiv preprint arXiv:2104.02012*, 2022.

[16] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval research logistics quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955.

[17] Y. Ko, T. Verma, N. A. M. Araujo, and M. Warnier, "Matcasc: A tool to analyse cascading line outages in power grids," in *2013 IEEE International Workshop on Inteligent Energy Systems (IWIES)*, 2013, pp. 143–148.
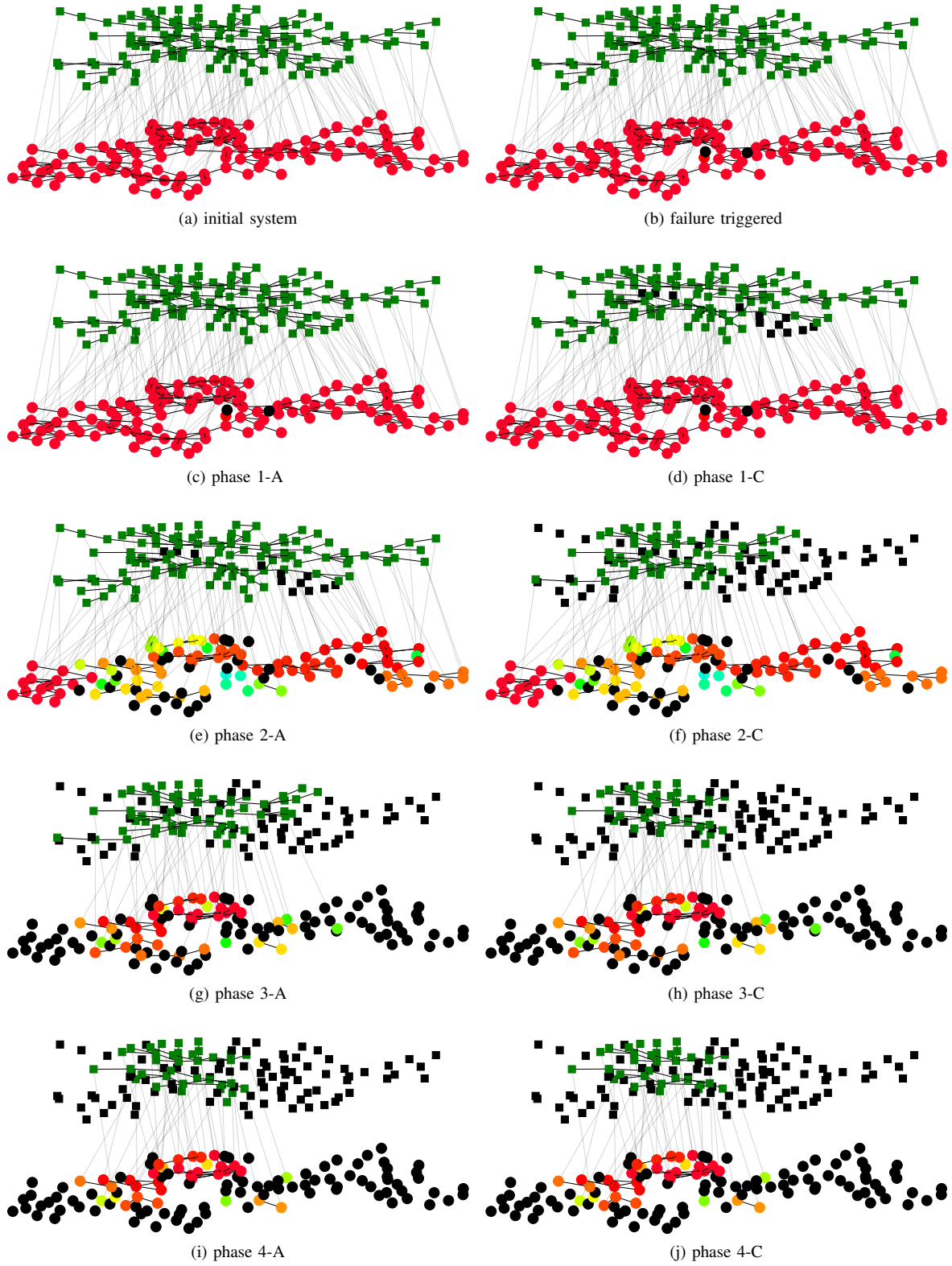
Fig. 2. Iterations of a cascading failure for the IEEE 118-bus test system where the power and cyber nodes are depicted with circles at the bottom layer and squares at the top layer, respectively. Different islands are plotted with different colors and failed nodes are always given in black. Note that failure triggered in two power buses (b) eventually stopped after four iterations (j). Due to the space limitation, only phases A and C are depicted.

(a) initial system

(b) failure triggered

(c) phase 1-A

(d) phase 1-C

(e) phase 2-A

(f) phase 2-C

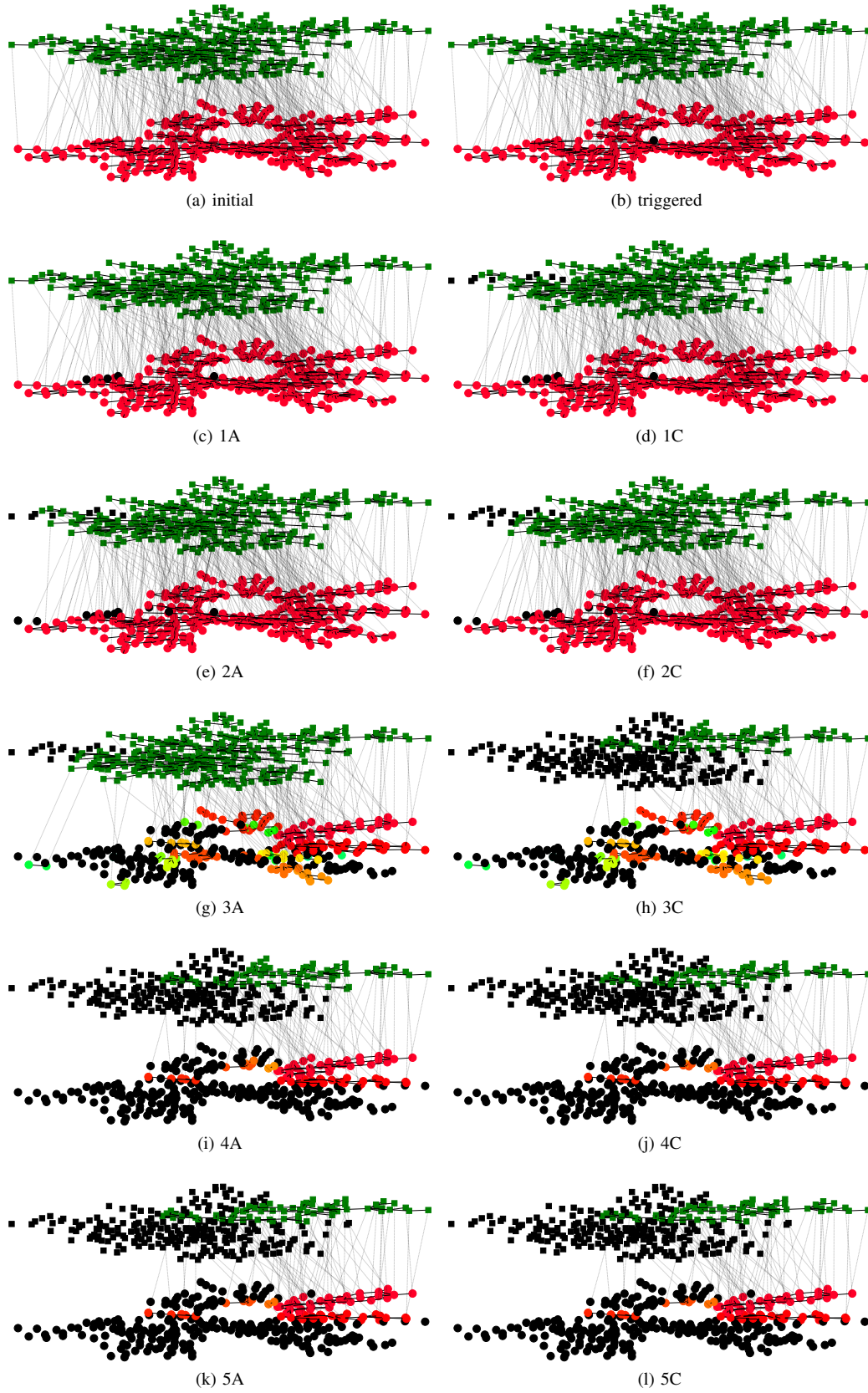(g) phase 3-A

(h) phase 3-C

(i) phase 4-A

(j) phase 4-C

Fig. 3. Different iterations of the proposed model for failure propagation in the IEEE 300-bus test system for a random attack. Note that the failure triggered in the power system (b) propagates in the power system which causes three additional bus failures (c). It demonstrates that a failure can propagate non-locally due to the power system's physics.