Preferential Cyber Defense for Power Grids

Mohammadamin Moradi,[1] Yang Weng,[1] John Dirkman,[2] and Ying-Cheng Lai[1,3,*]

[1] *School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, Arizona 85287, USA*
[2] *Resource Innovations, 719 Main Street, Half Moon Bay, California 94019, USA*
[3] *Department of Physics, Arizona State University, Tempe, Arizona 85287, USA*

(Received 10 March 2023; revised 27 July 2023; accepted 26 September 2023; published 23 October 2023)

The integration of computing and communication capabilities into the power grid has led to vulnerabilities enabling attackers to launch cyberattacks on the grid. The resources that can be deployed to protect a power grid are limited, rendering the need to impose preferences and priorities in optimal resource allocation. Due to the complexity of modern power grids, exploitation of machine learning is desired for developing optimal preferential cybersecurity defense strategies, where choosing a suitable mathematical framework to describe preference satisfaction and articulating a specific machine-learning method are key. We develop a reinforcement-learning approach with the objective of satisfying the preferences as quantitatively described by linear temporal logic. To characterize the preferences, we exploit a probabilistic planning approach that transforms preference satisfaction into a mixed-integer programming (MIP) problem, incorporate MIP into the resource-allocation problem, and use reinforcement learning to obtain the optimal policy. Due to the time-varying nature of the problem, the transformation needs to be carried out and MIP is to be solved at each time step. Utilizing the benchmark W&W 6-bus power-grid network, we validate our preferential machine-learning framework to defend the system against attacks under limited resources. Although our framework is computationally intensive at the present, it provides a stepping stone toward developing more efficient machine-learning frameworks to preferentially defend large cyberphysical systems.

DOI: 10.1103/PRXEnergy.2.043007

I. INTRODUCTION

To meet the ever-growing need for reliable and clean energies, smart power grids incorporating a variety of renewable power sources into the existing electrical infrastructure have become widespread. The increasing complexity of smart grids renders necessary the deployment of cyber control and communications, making the grids a major class of cyberphysical systems. A smart grid aims not only to increase electricity generation but also to deliver enhanced transmission and distribution capabilities [1]. To meet the future demand, the U.S. Department of Energy (DOE) has identified seven principal characteristics for smart grids: anticipating and responding to system disturbances, optimization of asset utilization, power quality for digital economy, enabling new products and services, accommodation of all generations, consumer participation, and attack and natural-disaster resilience [2]. Underlying the day-to-day operation and functionality of the grid is a vastly complex network of cyber infrastructure composed of layers of computers and communication systems, which constitute the "hidden" backbone of power-system operations.

The integration of computing and communication capabilities into the power grid has led to numerous vulnerabilities. Cyber capabilities open doors for attackers to access a power grid and cause disruptions to the normal operation of the grid [3,4], which are commonly referred to as cyberattacks. The major blackouts of 2003 on the U.S.-Canadian border [5], the 2015 Ukraine blackout [6], and Stuxnet in 2010 [7] were caused by cyberattacks. The three key subsystems of a power grid are generation, transmission, and distribution. Among them, the transmission subsystem consists of widely distributed substations and high-voltage pylons and is therefore most vulnerable as it is more susceptible to attacks than the other two subsystems [8]. In accordance with the DOE criteria of asset optimization and attack resilience, it is critical to allocate optimal (but finite) resources to protecting the transmission lines against cyberattacks.

[*] Ying-Cheng.Lai@asu.edu

In the real world, the defenders usually have incomplete or no knowledge of the attacker's strategies and resources. Yet, it is possible to make certain assumptions or estimates based on available information and historical data. In practice, the determination of the attack preferences can be approached through various methods, such as analyzing the past cyberattacks, considering known attack patterns or strategies, and leveraging expertise from cybersecurity professionals. An example is the Stuxnet cyberattack [7]. In 2010, Stuxnet, a sophisticated malware program, targeted Iran's nuclear facilities. This attack specifically focused on compromising industrial control systems, such as programmable logic controllers (PLCs), with the aim of sabotaging the centrifuges used for uranium enrichment. The attack demonstrated a clear preference for targeting specific components and exploiting vulnerabilities unique to the targeted infrastructure. In addition to known attacks such as Stuxnet, the determination of attack preferences can also leverage broader trends and patterns observed in the landscape of cyber threats. For example, there is a recurring preference among attackers to exploit common vulnerabilities in widely used software or to target high-value assets, such as financial institutions or critical infrastructure systems. By considering these historical trends and common attacker objectives, defenders can develop proactive defense strategies that prioritize the protection of critical assets and vulnerabilities commonly targeted by attackers. However, it is important to note that the determination of attack preferences is always an ongoing challenge and that the evolving nature of cyber threats requires constant adaptation. While real-world examples such as the Stuxnet attack provide valuable insights, each attack is unique and attackers can change their strategies over time. Continuous monitoring, threat-intelligence sharing, and collaboration among cybersecurity professionals remain crucial to staying updated on emerging attack techniques and preferences.

In this paper, we address the problem of allocating finite resources for preferential defense of power grids quantitatively through machine learning. To accomplish this, there are two prerequisites: choosing a suitable mathematical framework to describe preference satisfaction and articulating a specific machine-learning method. Our respective solutions are linear temporal logic (LTL) and reinforcement learning (RL), explained as follows.

In a real situation, the deployable resources are limited and it is not possible to protect all transmission lines in the grid. As a result, preference becomes an important factor of consideration in the articulation of defense strategies. In fact, preferences are generally one of the most important factors in human decision making. For example, in a problem consisting of a number of tasks, it is likely that not all the task goals can be achieved simultaneously and it is necessary to assign preferences to certain tasks. Under limited resources, preference-based planning is of fundamental importance to the security of cyberphysical systems. Mathematically, a decision-theoretic planning problem can be formulated as a Markov decision process (MDP), with the objective of obtaining an optimal policy to achieve maximum-preference satisfaction [9]. Generally, in the power-grid industry, decision making frequently entails bringing about compromises between conflicting priorities and preferences. The operators of the power grid, for instance, must strike a trade-off between the requirement to give customers access to dependable electricity and the need to do so at the lowest feasible cost. A previous technique for assessing options based on many criteria, termed multicriterion decision analysis (MCDA) [10], is effective for making decisions in this situation. For power grids, the analysis entails assessing potential power-generating technologies or operational approaches considering factors such as cost, dependability, and environmental impact. In the power-grid industry, decision making needs to be carried out with incomplete or uncertain information, as the operators must make decisions in real time based on often noisy and incomplete data.

In view of the uncertainties, the key to effective decision making in the power-grid industry is to carefully weigh the trade-offs among competing objectives and to use a systematic approach to evaluate and compare options and preferences. In this regard, LTL stands out as an effective task-specification language [11]. In particular, LTL can be exploited as a quantitative tool to bridge the task specification designed by the user with the designated objectives of the problem. Previously, optimal control for a system subject to LTL constraints has been studied [12]. It has also been used to design a system that maximally realizes its partial specifications [13]. In addition, achieving the maximum satisfaction of a given LTL formula while minimizing the steady-state average cost in a Markovian optimal control setting has been investigated [14,15] and a similar approach has been used to develop a transient analysis to find a policy that minimizes the accumulated cost on a finite horizon [16]. We note that a necessary step toward realizing preferential defense of a cyberphysical system is to find an optimal allocation policy for a fixed amount of resources to achieve a predetermined goal—the resource-allocation problem. A previous solution has been based on negotiation theory [17]. The problem has also been studied in the field of cybersecurity [18]. Alternative solution approaches include game theory [19,20] and MDP [21].

The dynamics of a power grid are generally nonlinear. Another complication is the occurrence of cascading failures [22] in power grids. Because of the complexity, RL—which is a branch of artificial intelligence that enables a dynamical system to learn from experiences gathered from interacting with its environment—stands out as a viable strategy to solve the preferential cyber-defense problem for power grids. As a main derivative of machine learning, RL has proven to be useful for solving cybersecurity problems. There have been numerous studies using RL to solve smart-grid cybersecurity problems, such as in false-data-injection attacks [23], topology attacks [24,25], attack mitigation [26–28], attack detection [29,30], and persistent attacks [31]. Traditionally, RL dealt with problems in which the state-action space was small and could be tabulated—the so-called Q function or Q table. However, when the state-action space is large, as in typical smart grids, the Q-table approach is not viable. In this case, neural networks can be used in RL to approximate the Q function, leading to deep Q learning [32,33]. In the field of smart-grid security, deep RL has been implemented [25,26,28,30]. LTL formulas as task-specification tools have been transferred into the RL framework [34]. Multiagent RL with LTL specifications has also been studied [35]. In robotics, the complex rules that the robot should follow have been articulated using temporal logic and an RL algorithm has been developed to learn tasks expressed as truncated LTL formulas [36]. Algorithms combining RL and LTL have also been widely studied in control engineering. For example, synthesizing the controller required to produce a specified closed-loop response given the model of a feedback loop has been achieved using combined RL and LTL [37–39].

The main challenge associated with the resource-allocation problem in cybersecurity of power grids lies in allocating limited resources, such as budget and personnel, to different security measures in order to protect the grid from potential cyberattacks. The problem is extraordinarily complex because of the many different types of potential threats to the grid, each requiring a different set of resources for effective mitigation. In this paper, we solve the optimal resource-allocation problem by developing optimal preferential cybersecurity defense strategies for cyberphysical systems, using power grids as a concrete setting. In the case of limited resources in terms of, e.g., funding, available security guards, or security cameras, realizing optimal resource allocation while taking into account the user preferences is of the utmost importance. Consider the power grid in Fig. 1 and assume that the government has some preferences based on the cultural and political attributes of the cities represented. For example, when an attack occurs, a blackout in cities A and B may be preferred to a blackout in cities B and C. The question is how to allocate the available resources to defend the transmission lines so that the preference is maximally satisfied. We articulate an RL approach with the objective of satisfying the preferences specified by the LTL formulas. To model the preference problem, we exploit the probabilistic planning approach [40], which transforms preference satisfaction into a mixed-integer programming (MIP) problem. Due to the time variations, the transformation needs to be carried out at each time step to enable a new MIP problem to be formulated. Once MIP is incorporated into the resource-allocation problem, RL can be executed to obtain the optimal policy, at each time step. The result is a kind of adaptivity: the optimal policy needs to be varied from time to time to offer the best protection of the power grid subject to preference satisfaction.

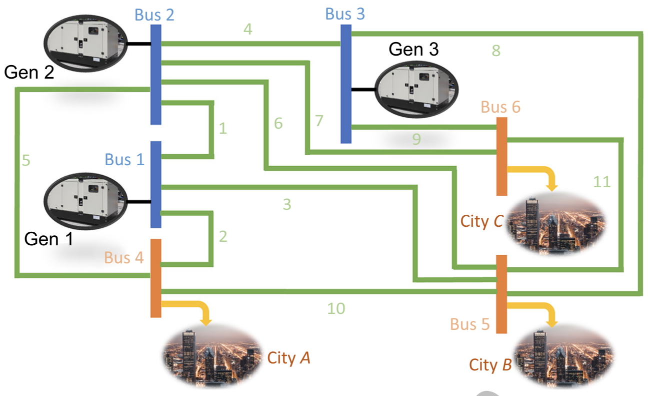

FIG. 1. A representation of a power grid, modeled as the W&W 6-bus system. The system has six buses, three generators (denoted as "Gen"), three loads (here depicted as cities), and 11 transmission lines. The simulation of the power grid is performed using the DCSIMSEP software package, a simulator of cascading failures in power systems. DCSIMSEP does not use any specific stress-mitigating controls under the assumption that the cascades propagate too fast for the operators to react, so it is suitable for cybersecurity problems.

In Sec. II, we quantify preferences using temporal logic formulas, formulate the resource-allocation problem for preferential cybersecurity defense, and articulate an RL approach to the optimal solution. The results are presented in Sec. III and a discussion is offered in Sec. IV.

II. MIXED-INTEGER PROGRAMMING FORMULATION OF PREFERENCE AND REINFORCEMENT LEARNING

Temporal logic is a type of formal logic that can be used to reason about events and their relationships over time. In the context of power grids, temporal logic can be used to model and analyze the dynamic behavior of the grid over time. This can be useful for tasks such as verifying the correctness of the control systems, predicting the behavior of the grid under different scenarios, and identifying potential vulnerabilities or weaknesses in the design of the grid. Several different formal methods for temporal logic exist, each with its own syntax and semantics for representing and reasoning about time. Some common examples include LTL and computation-tree logic (CTL). These methods can be used to express statements about the behavior of the power grid over time, such as "the power grid will always remain stable" or "the power grid will eventually reach a state where demand is equal to supply," etc. These statements can then be evaluated using automated theorem-proving or model-checking techniques to determine whether they are true or false. Overall, temporal logic is an effective tool for describing the dynamic evolution of the power grid and can be used to improve the reliability and robustness of the grid. In this section, we introduce the concept of the finite automaton and illustrate, by using an example, how preferences can be quantified by the LTL formulas. Based on the constraints from the LTL formulas, we demonstrate that the preferential cyber-defense problem can be formulated as an MIP problem.

A. Quantifying preferences with LTL formulas

A deterministic finite automaton (DFA) is a finite-state machine that accepts or rejects a given string of symbols by running through a state sequence deterministically specified by the string [41,42]. Based on DFA, preferences over accepting conditions can be modeled [40]. A DFA is a 5-tuple $\langle \tilde{S}, \Sigma, \delta, \tilde{s}_0, \phi \rangle$, where $\tilde{S}$ represents the set of automaton states, Σ is the set of possible automaton symbols or actions, δ is the transition function, $\tilde{s}_0$ is the initial automaton state, and ϕ represents the preference formula(s). To incorporate preferences into an MDP, we define the *preference-induced MDP*, which is a 4-tuple, $\langle \tilde{S} \times S, A, d, \Delta \rangle$, where S denotes the set of MDP states, A is the set of MDP actions, and d is the probability of the initial state distribution. The transition function Δ is defined as

$$\Delta((\tilde{s}', s')|(\tilde{s}, s), a) = P(s'|s, a) * \mathbf{1}_{\{\tilde{s}'\}}\{\delta(L(s'), \tilde{s})\}, \quad (1)$$

which means that from MDP state s and automaton state $\tilde{s}$, under action a, the probability of going into MDP state s' and automaton state $\tilde{s}'$ is equal to the sum of the probabilities of going into MDP state s' from MDP state s, taking action a for all possible transitions. In Eq. (1), L is the labeling function that maps MDP states to their corresponding automaton actions.

To explain the definition in Eq. (1), we consider a concrete example of a power grid to describe the preferences from the attacker's perspective [40]: the W&W 6-bus system—denoted as *Example 1*. As illustrated in Fig. 1), this benchmark power grid consists of three generators, each susceptible to attacks. Assume that the attacker can attack the generators in any order. Considering different uncertainties such as the available attack window, the defender's capabilities, and the generation cost, the attacker can specify the following possible policies as preferences (ordered by priority):

(i) *PA*: the more generators being attacked, the better.
(ii) *PB*: if attacking all generators is possible, attacking Gen 3 in the end is preferred.
(iii) *PC*: attacking Gen 1 first is preferred to attacking other generators first.

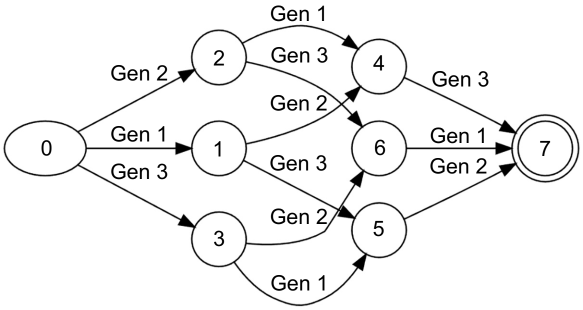

FIG. 2. An illustration of an automation of preferences. For *Example 1* described in the text, the automation comprises eight states, with no. 0 being the starting state and no. 7 being the final state. For preference *PA*, from the attacker's point of view, attacking three generators is preferred to attacking two generators, which in turn is preferred to attacking only one generator. Similarly, attacking one generator is preferred to attacking no generators. The preference can be described by the formula $\{7\} \succeq \{4, 5, 6\} \succeq \{1, 2, 3\} \succeq \{0\}$. The same logic applies to preference *PC*. Since the automaton is not able to describe preference *PB*, an improvised automaton is constructed, as shown in Fig. 3.

The resulting automaton is illustrated in Fig. 2. Note that the constructed standard automaton has eight states, where *PA* and *PC* can be implemented as the following preference formulas:

(i) $\{7\} \succeq \{4, 5, 6\} \succeq \{1, 2, 3\} \succeq \{0\}$
(ii) $\{1\} \succeq \{2, 3\}$

To derive the preference formulas for *PB*, the automaton in Fig. 2 cannot be used, because the states ending with attacking Gen 3 are not distinguishable from the states ending with attacking the other generators. Modifying the automaton to the one in Fig. 3 and adding an extra automaton state allows us to express the preference *PB* explicitly as

(i) $\phi_1 : \{8\} \succeq \{7\} \succeq \{4, 5, 6\} \succeq \{1, 2, 3\} \succeq \{0\}$
(ii) $\phi_2 : \{1\} \succeq \{2, 3\}$

More specifically, for the example considered, we have the following:

(i) $\tilde{S} = \{0, 1, \ldots, 8\}$ is the set of automaton states.
(ii) $S = \{[s_1, s_2, s_3]\}$, where s_i is a binary variable, which is 0 if Gen i is attacked, and S is the set of MDP states. (As will be demonstrated in Sec. III, natural numbers can be used instead of vectors for representing the MDP states, for convenience.) The transformation follows the conversion of the logical bit-wise inverse of the state, which is binary to its decimal counterpart. For example, the states $s = [1, 1, 1]$, $s = [0, 0, 0]$, and $s = [0, 0, 1]$ are equivalent to nos. 0, 7, and 6, respectively. (In Sec. III, S is the set representing the condition of the transmission

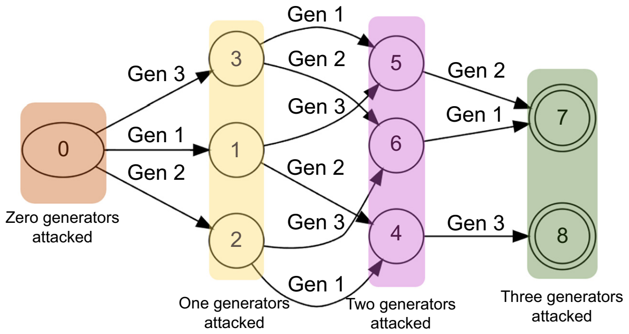

FIG. 3. A modified automaton of *Example 1*. There are now nine states, with no. 0 being the starting state and nos. 7 and 8 being the final states. Since the automaton in Fig. 2 is not able to describe preference *PB*, an improvised automaton is constructed. Preference *PB* stipulates that, from the attacker's point of view, attacking Gen 3 in the end is preferred to attacking other generators if attacking all generators is possible. Using the refined automaton, this preference can be described by adding state no. 8 to the formula illustrated in Fig. 2: $\{8\} \succeq \{7\} \succeq \{4,5,6\} \succeq \{1,2,3\} \succeq \{0\}$.

lines following the same binary logic as is used here.)

(iii) $\Sigma = A = \{Gen\ 1, Gen\ 2, Gen\ 3\}$ is the set consisting of the automaton and MDP actions, where Σ is usually called the alphabet in the theory of automata [41,42].

(iv) The transition function δ is represented by the graph in Fig. 3.

(v) $\tilde{s}_0 = \{0\}$ is the set containing the initial state(s) of the automaton.

Transforming preferences into temporal logic formulas in the form $\{P\} \preceq \{P'\}$ enables us to formulate an MIP problem to find an optimal policy to maximally satisfy the preferences.

B. Maximum-preference satisfaction as a mixed-integer programming problem

In an MIP problem [43], some variables of the system to be optimized are integers with a linear objective function, subject to linear constraints. In our work, the objective function is the preference-satisfaction value and there are two sets of constraints: one representing the preference-satisfaction value while the other denotes the preferences defined by the user.

The definition of the value of the preference satisfaction (V_{PS}) is closely related to the probability of occurrence of the corresponding preference, where V_{PS} for a preference formula $X_0 \preceq X_1 \preceq \cdots \preceq X_n$ is defined as [40] $P(X_i)$ if there exists some i such that $P(X_i) \geq P(X_{i-1})$, while for all $k \geq i$, $P(X_k) < P(X_{k-1})$ holds and is zero otherwise. The definition of V_{PS} can be illustrated using the example discussed in Sec. II A. Assume that for the preference formula ϕ_1, the two derived policy samples induce the probabilities as listed in Table I. From the definition of V_{PS}, policy π_1 satisfies the preference ϕ_1 by 50%, whereas policy π_2 satisfies the preference ϕ_1 by 80%. This problem is denoted as *Example 2*.

TABLE I. The probability of the automaton ending in specific states for two policies in *Example 1*.

Automaton state set	P_{π_1}	P_{π_2}
{8}	0.05	0.8
{7}	0.15	0.05
{4, 5, 6}	0.5	0
{1, 2, 3}	0.1	0.05
{0}	0.2	0.1

To obtain the constraints for the MIP problem, we employ a supporting variable $y(t, (\tilde{s}, s), a)$, defined as the probability of visiting the state pair $(\tilde{s}, s)$ at time t and taking action a, so that $y(T, P)$ is the probability of the automaton being at state P at time $t = T$ (regarded as the ending point). Using the supporting variable and from the definition of V_{PS} for the preference formula, we have the first set of MIP constraints $\{P\} \preceq \{P'\}$ as [40]

$$0 \leq V_{\mathrm{PS}} \leq B, \tag{2}$$

$$B - 1 \leq V_{\mathrm{PS}} - y(T, P') \leq 0, \tag{3}$$

$$B(1+\epsilon) + 1 \leq y(T, P') - y(T, P) \leq B(1+\epsilon) - \epsilon, \tag{4}$$

$$B \text{ is a binary}, \tag{5}$$

$$\text{all } y \text{ are non-negative}, \tag{6}$$

where ϵ is a small positive number and B is a binary variable: it is 1 if there exists a policy that satisfies the preference and 0 otherwise. From Eqs. (2) and (3), it can be seen that if there exists no such policy satisfying the preference, then $V_{\mathrm{PS}} = 0$; otherwise its value will be between 0 and 1 and is equal to $y(T, P')$. We enforce the probabilities of the events and their difference to be between 0 and 1 using Eqs. (4) and (6). The second set of MIP constraints is responsible for maintaining the consistency between the supporting variable $y(t, (\tilde{s}, s), a)$ and MDP elements d and Δ. For all possible state pairs and for all time steps, we have

$$\sum_{a \in A} y(0, (\tilde{s}, s), a) = d(\tilde{s}, s), \tag{7}$$

$$\sum_{a \in A} y(t, (\tilde{s}', s'), a) = \sum_{a \in A} \sum_{(\tilde{s}, s) \in \tilde{S} \times S} \Delta((\tilde{s}', s') | (\tilde{s}, s), a) y(t-1, (\tilde{s}, s), a). \tag{8}$$

Equation (7) ensures that the probabilities denoted by the supporting variable y at time $t = 0$ are consistent with the distribution of initial state pair d. Equation (8) asserts that the probability of getting to a state pair at time t is equal to the sum of probabilities that other possible state pairs take the specific action to reach the target state pair at time $t-1$. Taking these constraints into consideration, we formulate the maximum-preference-satisfaction problem for the preference formula $\{P\} \preceq \{P'\}$ as

$$\max_{B,y,V_{\text{PS}}} \; V_{\text{PS}} \text{ subject to Eqs. (2)} - \text{(8)}, \qquad (9)$$

where B, y, and V_{PS} are the optimization variables and V_{PS} is the objective function. Equation (9) represents an MIP problem because it includes an integer constraint [Eq. (5)]; otherwise it is a standard linear-programming problem. Our main idea is to integrate RL and MIP to solve the preferential cyber-defense problem for power grids.

C. Reinforcement-learning solution to the preferential resource-allocation problem

Reinforcement learning is a decision-making tool, where the "agent" explores the "environment," interacts with it, and collects observations to find an optimal behavior in order to maximize a long-term "reward." While RL is capable of directly solving certain cyber-security problems [28], here we exploit it to find the preferential optimal resource allocation. In particular, suppose that, in a power grid (e.g., the one shown in Fig. 1), the security decision maker intends to distribute a number H of defensive resources among L transmission lines. To prevent a disastrous blackout, the defender must consider the potential attacker's policy and the criticality of the transmission lines along with the political and technical preferences. From Fig. 4, it can be seen that the defender (agent) starts with the initial allocation state $h_0 = [0, 0, \ldots, 0]$, which means that no resources have been assigned to any of the transmission lines as of yet. At this step, an action is selected and a single resource unit is assigned to the respective transmission line. The next allocation state, h', can be determined based on the chosen action. We use the epsilon greedy method [44] to select a proper action, where the action with the largest Q value is chosen with the probability of $1-\epsilon$ and a random action is performed with the probability ϵ. Because the transition functions and the attacker's policy have changed in response to the new allocation state h', a new MIP problem needs to be formulated and solved. The maximum-preference satisfaction, the solution of Eq. (9), is taken as the reward of the current time step. The agent will observe the reward of its action and will "learn" from the acquired information. The epsilon greedy algorithm gives a new choice of the action, resulting in a new MIP problem, the solution of which will lead to a reward. This process continues until a total of H actions have been taken and the RL "episode" is regarded as being over, initiating another episode. The whole process stops when the agent's learning objective has been achieved. The optimal policy is thus a sequence of allocation actions, which represents the optimal solution of the resource-allocation problem.

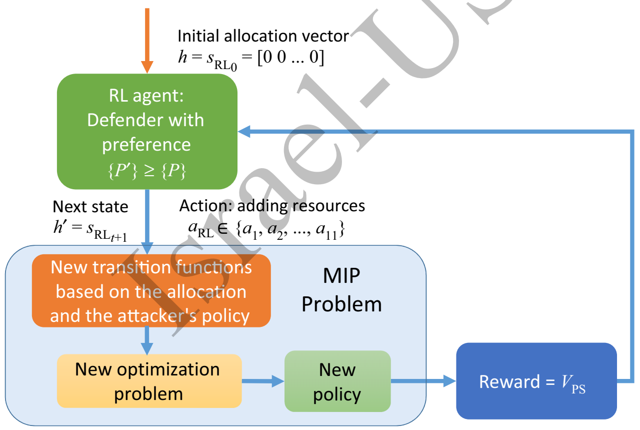
Initial allocation vector
$h = s_{\text{RL}_0} = [0\ 0\ \ldots\ 0]$
RL agent: Defender with preference $\{P'\} \geq \{P\}$
Next state
$h' = s_{\text{RL}_{t+1}}$
Action: adding resources
$a_{\text{RL}} \in \{a_1, a_2, \ldots, a_{11}\}$
New transition functions based on the allocation and the attacker's policy
MIP Problem
New optimization problem
New policy
Reward = V_{PS}

FIG. 4. The proposed RL-based method to solve the preferential optimal resource-allocation problem. At each time step, application of the epsilon greedy algorithm results in a new choice of the action, a new MIP problem the solution of which leads to the reward. One RL episode consists of a total of H time steps.

Depending on the setting of the environment, a wide array of RL methods exists. In the setting of our resource-allocation problem, both the state (the allocation state) and action are discrete. While several RL methods are available, such as the actor-critic (AC) method [45], the policy-gradient (PG) method [46], and the proximal policy optimization (PPO) method [47], which are suitable for our setting, we prefer deep Q learning as it is efficient for power grids. In Q learning, the Q function is a mapping of all possible state-action pairs to a scalar value and represents the expected total discounted reward that an agent anticipates obtaining through starting from a determined state and taking a specified action. The optimal Q function can be defined as

$$Q^*(s_{\text{RL}}, a_{\text{RL}}) = r(s_{\text{RL}}, a_{\text{RL}}) + \gamma \sum_{s'_{\text{RL}}=1}^{N} p(s'_{\text{RL}}|s_{\text{RL}}, a_{\text{RL}}) v(s'_{\text{RL}}, \pi), \qquad (10)$$

where $s'_{\text{RL}} (or\ h')$ is the next state evolving from state $s_{\text{RL}} (or\ h)$, taking action a_{RL}. In order to efficiently approximate the Q function, we employ a deep RL method as a replacement for the usual tabular Q learning. The approximator in deep Q learning is a multilayered neural network [32]. For any given state s_{RL}, the network outputs a vector of action values $Q(s_{\text{RL}}, ., .; \theta)$, where θ denotes the set of parameters of the online network. The target network with the parameter set θ^* is the same as the online network except that for every c episode, its parameters are copied from the online network: $\theta_t^* = \theta_t$, which remains unchanged during the c episodes. The target used in deep Q learning can be described as

$$Q^{*t} = r^{t+1} + \gamma \max_{a_{\rm RL}} Q_t(s_{\rm RL}^{t+1}, a_{\rm RL}; \theta^{*t}). \tag{11}$$

The agent receives the initial allocation and calculates the Q-function values for all possible actions, which in our problem are the transmission lines to which a single resource unit is allocated. The allocation vector, action, the next allocation vector derived from the stochastic transition function, and the computed maximum value of preference satisfaction ($V_{\rm PS}$) are stored. The data are then sampled uniformly from the memory bank to update the network—the so-called experience replay—as some random batches of transitions are sampled. The error between the target and predicted Q functions is calculated as

$$e^t = Q^{*t} - Q^t(s_{\rm RL}^{t+1}, a_{\rm RL}; \theta^t), \tag{12}$$

where a small error signifies a well-trained algorithm. Typically, a gradient-descent algorithm can be used to optimize the online-network parameter values to minimize the error. The parameters of the target network are updated periodically to match those of the online network. Both the target network and experience replay can dramatically improve the performance of the algorithm [48]. Using the Q function defined Eq. (10), we determine the optimal resource-allocation for the power-grid security problem.

III. RESULTS

We solve the preferential resource-allocation problem in the power-grid cyber-security setting for different types of preferences. The simulations are carried out using the MATLAB R2021b Reinforcement Learning Toolbox on a desktop PC with an Intel Core i7-6850K CPU and 128 GB of RAM. We find that it is useful to employ an external optimizer for more complex problems rather than the MATLAB built-in solver. In this work, we use the GUROBI optimizer [49], due to its advanced algorithms, cutting-edge heuristics, and strong integer-feasibility checks, to solve the MIP problem to improve speed over the regular MATLAB solver. Specifically, the GUROBI branch-and-bound approach efficiently searches for the global optimum, while its parallel-processing capabilities utilize multicore processors effectively. These advantages make GUROBI a powerful choice for handling complex and large-scale MIP problems, offering significant speed improvements and a higher solution quality compared to the MATLAB built-in solver.

The power-grid model is the benchmark W&W 6-bus system shown in Fig. 1, with the assumption that the power supply is provided by three generators. A generator is out if all the transmission lines connected to it are in an outage state. To simulate the power grid, we use a dc load-flow simulator of cascading (separation) in power systems, named DCSIMSEP [50,51], and we incorporate the power grid into our preferential resource-allocation problem.

Transient effects and the growing share of renewables on the dynamics of power grids are at the forefront of issues in cyberphysical systems. It is important to consider these effects and to address their potential implications on the proposed preferential cyber-defense strategy, as they can significantly affect the stability and reaction of a power grid. A power grid, because of its intrinsic non-linear dynamics, is susceptible to cascading failures. The power-grid simulation framework DCSIMSEP used in our work takes into account nonlinear dynamics and cascading failures. In particular, the framework captures the complex interdependencies in the power grid and cascading effects that can arise during the propagation of failures [52]. In addition, the framework accounts for the transient effects and fluctuations (e.g., those induced by the growing share of renewable) in an implicit manner. DCSIMSEP simulations have allowed us to demonstrate the feasibility of incorporating preferential cyber-defense strategies into power-grid systems with cascading dynamics.

In the simulations, we assume that an attack on a specific line is successful with a probability that depends on the defender's resource allocation, which is updated during the learning process. If the attacker attacks line i, the probability of an outage on that line will be [53]

$$p(i) = \frac{1}{1+h(i)}, \tag{13}$$

where $h(i)$ is the ith component of the resource-allocation vector h. The attacker constructs an "attack pool," which is a mixture of random and maximally secured lines, so the attacker only attacks the lines belonging to the preselected attack pool. The attack-pool length is taken to be 3 (arbitrarily), so the maximal attack time is $T = 2$. Initial state pairs $d(\tilde{s}, s)$ follow the normal distribution, with mean $(0, 0)$ and standard deviation 2.5. Table II lists the simulation-parameter values for the deep Q-learning algorithm. The results from two examples of the

TABLE II. The deep Q-learning parameters.

Parameters	Values
Episodes	100–2000
Episode steps	H (available resource units)
Epsilon	0.5
Epsilon decay	0.001
Epsilon minimum	0.01
Learning rate	0.001
Disc. factor	1
Experience buffer length	10 000
Minibatch size	256
F.C. layer neurons	50

Disc., discount; F.C., fully connected.

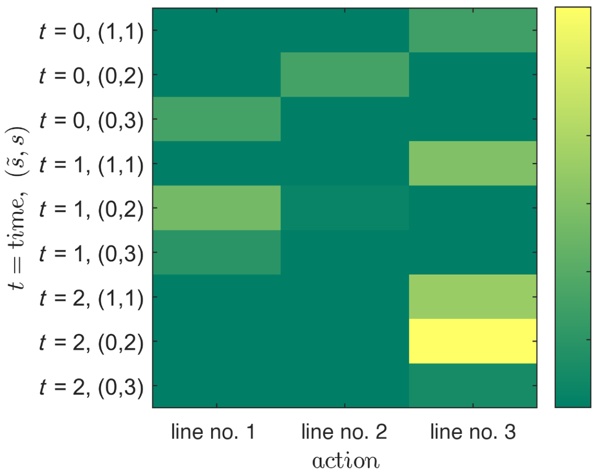

FIG. 5. An example of an MIP solution. The allocation vector is $[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]$. The supporting variables y are shown via a color map, $V_{\mathrm{PS}} = 1$, and $B = 1$. The color map describes the attacker's optimal policy (the defender's worst case), which maximizes preference satisfaction in the worst-case scenario. The derived policy is stochastic, where the color map indicates the probability of attack in each state pair of the power grid at each time step. For example, at time $t = 2$, if the attacker follows an optimal policy (the worst case for the defender), line no. 5 will be attacked about 60% of the time while the system is state pair $(0, 2)$. Overall, at time $t = 0$, all lines are equally in danger, whereas at times $t = 1$ and $t = 2$, lines nos. 1 and 5 are those in most danger, respectively.

preferential resource-allocation problem are presented in Figs. 5 and 6.

In the first case, there is only a single resource unit available ($H = 1$) and the maximal attack time is $T = 2$. The preference is set to be $\{0, 1, 2, 3, 4\} \succeq \{5, 6, 7, 8\}$, as shown in Fig. 3. The optimal allocation vector derived from deep Q learning is $h = [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]$, indicating that the single unit should be allocated to line no. 5. Figure 5 depicts the optimal attacker policy, which is the solution of the corresponding MIP problem defined in Eq. (9). There are three unique possible state pairs $(\tilde{s}, s)$, as in the state definitions of *Example 1*. For instance, from Fig. 5, we see that it is optimal for the attacker to attack line no. 1 rather than line no. 2 at time $t = 1$ when the power grid is in the automaton state $\tilde{s} = 0$ and MDP state $s = 2$. It is also more effective to attack line no. 2 than line no. 5 in the same situation. This optimal attacker policy, which is the worst-case scenario for the defender, satisfies the defender preference by 100% ($V_{\mathrm{PS}} = 1$). As a result, the allocation vector $h = [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]$, derived from the deep Q-learning algorithm, is the optimal preferential resource allocation for this problem for $H = 1$.

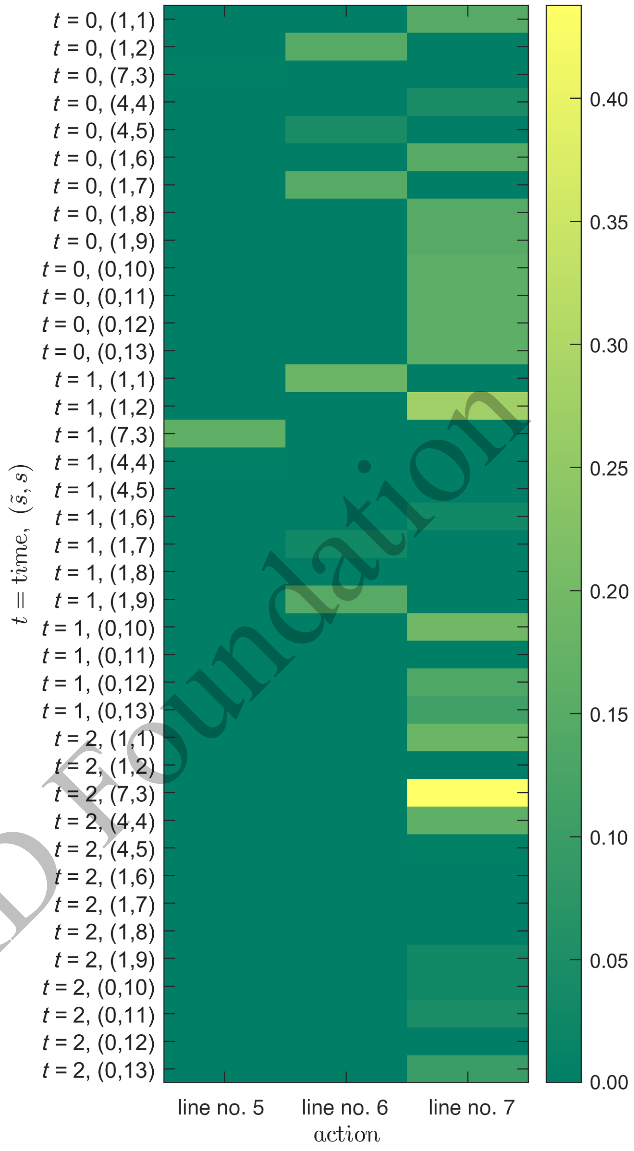

FIG. 6. An additional example illustrating the solution of an MIP problem. The allocation vector is $= [0, 0, 1, 0, 4, 0, 2, 3, 0, 0, 0]$. The supporting variables y are shown via a color map describing the attacker's optimal policy. The parameter values are $V_{\mathrm{PS}} = 0.6057$ and $B = 1$. At time $t = 1$, if the attacker follows an optimal policy (the worst-case scenario for the defender), line no. 8 will be attacked about 35% of the time while the system is state pair $(1, 2)$. For this policy, at almost all times, line no. 8 is the most vulnerable. However, the optimal allocation vector allocates more resources to defending line no. 5 since this line is the most critical in this system (see Ref. [28]), suggesting that the derived optimal allocation is applicable even when the attacker does not follow the optimal policy, since the algorithm also considers the intrinsic dynamics of the power-grid system.

In the second case, ten resource units are available ($H = 10$) and the preference is set to be $\{0, 1, 2, 3\} \succeq \{4, 5, 6, 7, 8\}$, as shown in Fig. 3. The application of deep Q learning gives the optimal allocation vector based on this preference as $h = [0, 0, 1, 0, 4, 0, 2, 3, 0, 0, 0]$. Figure 6 shows the optimal attacker policy, which is the solution of the MIP problem in Eq. (9). This time, there

TABLE III. The optimal preferential resource-allocation results derived from the proposed deep Q-learning-based method.

Resources H	Preference	Allocation vector h	V_{PS}
0	1	$[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$	0.3192
0	2	$[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$	0.3192
0	3	$[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]$	0
1	2	$[0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]$	1
2	2	$[0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0]$	1
3	2	$[0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0]$	1
4	1	$[0, 0, 0, 2, 1, 1, 0, 0, 0, 0, 0]$	1
7	3	$[0, 0, 0, 0, 3, 0, 2, 2, 0, 0, 0]$	0.5941
10	3	$[0, 0, 1, 0, 4, 0, 2, 3, 0, 0, 0]$	0.6057
20	3	$[0, 0, 0, 0, 10, 0, 1, 9, 0, 0, 0]$	0.6475

are 13 unique possible state pairs $(\tilde{s}, s)$ (*Example 1*). For instance, Fig. 6 indicates that it is optimal for the attacker to attack line no. 8 rather than line no. 5 or line no. 7 at time $t = 2$ when the power grid is in the automaton state $\tilde{s} = 7$ and MDP state $s = 3$. Similar to the first case, the optimal attacker policy, which is the worst-case scenario for the defender, satisfies the defender preference by 60.57% ($V_{\text{PS}} = 0.6057$). Consequently, the allocation vector $h = [0, 0, 1, 0, 4, 0, 2, 3, 0, 0, 0]$ is the optimal preferential resource allocation for this $H = 10$ case.

Table III summarizes the results for different preferences and various amounts of available resource units. For clarity of presentation, the preferences $\{6\} \preceq \{0, 1, 2, 3, 4, 5\}$, $\{0, 1, 2, 3, 4\} \succeq \{5, 6, 7, 8\}$, and $\{0, 1, 2, 3\} \succeq \{4, 5, 6, 7, 8\}$ are denoted as preferences 1, 2, and 3, respectively.

While our results generate logical and consistent outcomes achieved through the proposed framework, it would be useful to compare our results with benchmarks. However, to our knowledge, there are no previous works in the literature on resource allocation with preferences in cyber defense for power grids, making a benchmark study infeasible at the present. Nonetheless, we have demonstrated the effectiveness of our framework in satisfying preferences and optimizing resource allocation.

Remark 1.—In our study, only the preference formulas of length 1 have been considered. For preferences with length more than 1, such as $\{P_1\} \preceq \{P_2\} \preceq \{P_3\}$, further modification to the MIP constraints is required. If the transformation of longer-preference formulas to the logical combination (and/or) of length-1 formulas is feasible, the same approach is applicable. Otherwise, the constraints must be handcrafted according to the specific formulas.

Remark 2.—The number of the equality constraints of the MIP optimization problem from Eqs. (7) and (8) is equal to $(T + 1)$ times the number of unique possible state pairs. For example, in the second case, there are 13 unique state pairs, so the number of equality constraints is 39. For a more complex preference, a longer attack time, or a larger power grid, this number will grow quickly, causing the MIP problem to be computationally intractable. In fact, as the constraints grow to the order of more than a couple of hundred, MIP solutions may not be feasible. As practical guidance, the MIP approach requires that the preferences and consequently the state pairs be simple.

Remark 3.—Reinforcement-learning algorithms, like all other learning methods (optimization algorithms in general), are in some cases prone to the emergence of a suboptimal solution. This can happen occasionally during the simulations, yet a better training phase with tuned learning parameters can often solve the problem, requiring proper modifications of the reward function, the neural network structure, and/or the learning parameters.

Remark 4.—To address the limitations of our current framework with respect to the representation of the power-grid dynamics and the effects of transients and renewable fluctuations, a more advanced modeling approach would be needed. In the present study, we have utilized the DCSIMSEP package implemented on MATLAB, which captures interdependencies and cascading failures within the power grid. To account for renewable fluctuations, an alternative simulation environment, such as that offered by Grid2Op [54], an open-source PYTHON package specifically designed to facilitate the development and evaluation of RL algorithms for power systems, would be needed. More specifically, Grid2Op provides a simulation environment that models the operation of an electrical grid and integrates with the OpenAI Gym interface, offering a wide range of actions and rewards for training and testing RL agents. The Grid2op framework enables renewables to be incorporated into the power grid under a wider range of scenarios, providing a platform to study both the transient effects and fluctuations introduced by renewable energy sources. It is possible that investigating the applicability of Grid2Op for the preferential cyber defense of power grids can provide insights into the interplay among grid dynamics, stability, and cyber-defense strategies.

Remark 5.—While our results demonstrate the effectiveness of the proposed framework in optimizing resource allocation and satisfying specified preferences on the benchmark W&W 6-bus power-grid network, scaling up to large systems remains a significant challenge. In particular, utilizing MIP in the preference-satisfaction formulation and incorporating the MIP problem into the RL training process are computationally complex. Alternative mathematical formulations and optimization techniques should be explored to address this challenge. In addition, advancements in RL algorithms, such as sample-efficient algorithms or approaches that leverage distributed computing, could help alleviate the training burden associated with incorporating the MIP problem. We hope that our present work can serve as a starting point toward developing a preferential cyber-defense framework for large power grids.

IV. DISCUSSION AND CONCLUSIONS

In an ideal world where an infinite number of resources are available to protect a cyberphysical system, full security against cyberattacks can be guaranteed. In the real world, the resources that can be deployed to protect a power grid are limited, so it is practically impossible to protect all components, including each and every transmission line, especially when the system is large. To develop practical and effective cyber-defense strategies, preferences must be taken into account to offer the maximally possible protection of the power grid. The considerations used to determine certain preferences vary and often depend on a variety of technological, financial, and even political factors. To our knowledge, prior to our work, there has been little work that has addressed the problem of developing defense strategies for power grids against cyberattacks by allocating optimal resources according to specific preferences.

A necessary step in solving the problem of preferential optimal resource allocation for power grids is to identify a mathematical tool to quantify the preferences. Our approach is to exploit automata theory to transform an everyday-language preference into an MIP problem. We then develop an RL-based framework to solve the preferential cyber-defense problem, where MIP is employed as the system dynamics for the RL-based framework to generate the optimal resource allocation to best protect the power grid under the preference constraint. Utilizing the benchmark W&W 6-bus power-grid network, we have validated our preferential machine-learning framework to maximally defend the system against attacks using limited resources.

We have carried out new simulations to address the impact of fluctuations induced by renewables on the power grid. In general, introducing renewable sources into a power grid will lead to a new set of challenges for grid management due to their inherent intermittency and variability. In our simulations, we first establish a fixed allocation of resources for the power grid under consideration and then employ a determined preference. The inherent variability of renewable energy generation, driven by factors such as weather patterns and diurnal cycles, results in fluctuations in the power output. Our goal is to understand how these fluctuations affect the transition function, denoted as $\Delta((\tilde{s}', s')|(\tilde{s}, s), a)$, which characterizes the probability of transition from one state to another, given specific actions. We model the fluctuations as random perturbations added to the probabilities, leading to reconstructing the MIP and finding a new solution for preference satisfaction. Through this model, we are able to reconstruct the MIP formulation and find a novel solution that accounts for preference satisfaction in the presence of fluctuations induced by renewables. Our analysis indicates that the impact of these random fluctuations in the power grid is analogous to the effect of changing resource allocations in the MIP equations. This intriguing discovery suggests that the same temporal-language MIP strategy, previously applied to conventional power grids, can be seamlessly deployed in power grids incorporating renewables. As a result, the same temporal-language MIP strategy could be deployed to power grids that include renewables. Details of the simulation setting, results, and analysis are presented in Appendix B.

To make our framework meaningful for real-world problems, one shortcoming must be overcome. In particular, when the preferences are complex and the power grid is large, the number of MIP constraints tends to grow exponentially. At present, no effective methods exist for solving large-scale MIP problems. To develop methods to reduce the number of MIP constraints without violating the preference constraints and without deviating from the original solution is key to making our MIP and/or machine-learning cyber-defense framework realistic.

While methodologies incorporating preferences as temporal logic have been explored in other research fields such as robotics, our study presents a unique contribution by formulating temporal logic for preferences specifically for the resource-allocation problem in the cybersecurity of power grids. This distinction arises from the complex nature of power grids as cyberphysical systems and the specific challenges associated with defending them against cyberattacks. By integrating LTL preference specifications with RL, our framework provides a novel approach to addressing the optimization of resource allocation under limited resources for power-grid cybersecurity. By leveraging domain-specific insights and modeling the dynamics of power grids, our methodology offers a tailored solution that addresses some of the specific cybersecurity concerns of power grids. Through experiments on a benchmark power-grid network, we have demonstrated the potential impact of our framework in enhancing the resilience of power grids against cyber threats.

In principle, a multiagent adversarial-game framework represents a more comprehensive approach to modeling the adaptiveness of both the attacker and defender.

We focus on the defender's adaptiveness under limited resources, so as to prioritize the defender's adaptiveness so that it becomes feasible to develop a framework that allows the defender to dynamically adjust its defense strategy in response to changing circumstances and evolving cyber threats. This approach is particularly important in scenarios where the defenders have limited resources and need to efficiently allocate those resources to counter various attack strategies. The main contribution of our work is a quantitative machine-learning framework to address the practical challenge of resource allocation and the prioritization of the defender's actions in an evolving cyber-threat landscape.

Taken together, to overcome the limitations of our current framework to address the effects of transient and renewable fluctuations, the integration of dynamic modeling tools simulating power-grid dynamics under transient conditions is necessary. A possible approach is to leverage advanced simulation techniques, such as time-domain simulations or agent-based models, so that the complex dynamics and interdependencies within the power grid can be captured. Factors such as load balancing, voltage stability, and system response to disturbances can then be studied, providing a more accurate representation of the dynamical behavior of the power grid. Additionally, incorporating data-driven approaches and real-time monitoring can help better capture the fluctuations introduced by renewable energy sources. Integrating the dynamic modeling techniques and data-driven approaches into the framework can lead to a more comprehensive and realistic approach to preferential cyber defense for power grids.

ACKNOWLEDGMENTS

This work was supported by the U.S.-Israel Energy Center managed by the Israel-U.S. Binational Industrial Research and Development (BIRD) Foundation. This work was also supported by the Air Force Office of Scientific Research (AFOSR) under Grant No. FA9550-21-1-0438.

APPENDIX A: FORMULATION OF MIXED-INTEGER PROGRAMMING THROUGH A CONCRETE EXAMPLE

We use a concrete example (denoted as *Example 3*) to illustrate the implementation of the MIP formulation of optimization. The specific finite automaton is illustrated in Fig. 7. The setting is as follows:

(i) Preference:

$$p := (P' =)\{1\} \succeq (P =)\{2\}$$

(ii) Maximum attack time: $T = 3$

(iii) MDP states:

$$S = \{[1,1] := 0, [1,0] := 1, [0,1] := 2, [0,0] := 3\}$$

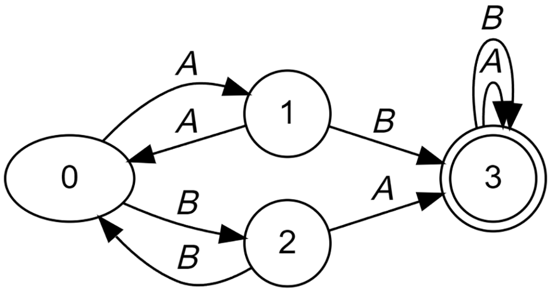

FIG. 7. The automaton of *Example 3*. There are four states, with no. 0 being the starting state and no. 3 being the final state. The preference $p := \{1\} \succeq \{2\}$ stipulates that ending in state no. 1 is preferred to ending in state no. 2. Table V shows that, using the derived policy, 76.89% of the time the system will end in state no. 1 compared with ending in state no. 2, which takes place 0.48% of the time, indicating that the preference is optimally satisfied.

(iv) Automaton states: $\tilde{S} = \{0, 1, 2, 3\}$

(v) Actions: $A = \{A, B\}$

(vi) Initial state-pair probability distribution:

$$d(\tilde{s}, s) = \begin{cases} 0.9 & \text{if } (\tilde{s}, s) = (0,0) \\ 0.05 & \text{if } (\tilde{s}, s) = (1,1) \\ 0.05 & \text{if } (\tilde{s}, s) = (2,2) \\ 0, & \text{else} \end{cases}$$

(vii) State-pair–transition probability distribution:

$$\Delta((1,1)|(0,0),A) = 0.8$$
$$\Delta((2,2)|(0,0),A) = 0.1$$
$$\Delta((3,3)|(0,0),A) = 0.05$$
$$\Delta((0,0)|(0,0),A) = 0.05$$
$$\Delta((1,1)|(0,0),B) = 0.1$$
$$\Delta((2,2)|(0,0),B) = 0.8$$
$$\Delta((3,3)|(0,0),B) = 0.05$$
$$\Delta((0,0)|(0,0),B) = 0.05$$
$$\Delta((0,0)|(1,1),A) = 0.05$$
$$\Delta((1,1)|(1,1),A) = 0.95$$
$$\Delta((3,3)|(1,1),B) = 1$$
$$\Delta((3,3)|(2,2),A) = 1$$
$$\Delta((0,0)|(2,2),B) = 0.05$$
$$\Delta((2,2)|(2,2),B) = 0.95$$
$$\Delta((3,3)|(3,3),A) = 1$$
$$\Delta((3,3)|(3,3),B) = 1$$

As discussed in Sec. II, the supporting variable $y(t, (\tilde{s}, s), a)$ is defined as the probability of visiting the

TABLE IV. The definition of the supporting variables $y(t, (\tilde{s}, s), a)$.

t	$(\tilde{s}, s)$	A	B
0	(0,0)	y_1	y_2
0	(1,1)	y_3	y_4
0	(2,2)	y_5	y_6
0	(3,3)	y_7	y_8
1	(0,0)	y_9	y_{10}
1	(1,1)	y_{11}	y_{12}
1	(2,2)	y_{13}	y_{14}
1	(3,3)	y_{15}	y_{16}
2	(0,0)	y_{17}	y_{18}
2	(1,1)	y_{19}	y_{20}
2	(2,2)	y_{21}	y_{22}
2	(3,3)	y_{23}	y_{24}
3	(0,0)	y_{25}	y_{26}
3	(1,1)	y_{27}	y_{28}
3	(2,2)	y_{29}	y_{30}
3	(3,3)	y_{31}	y_{32}

state pair $(\tilde{s}, s)$ at time t and taking action a. We use the variables listed in Table IV with the purpose of keeping the equations concise. For example, y_{11} is the probability of being in the state pair $(1, 1)$ at time $t = 1$ under action A. With these definitions and the initial state-pair probability distribution $(d(\tilde{s}, s))$, we construct the following constraints from Eq. (7):

$$y_1 + y_2 = 0.9, \tag{A1}$$

$$y_3 + y_4 = 0.05, \tag{A2}$$

$$y_5 + y_6 = 0.05, \tag{A3}$$

$$y_7 + y_8 = 0. \tag{A4}$$

Moreover, using the state-pair transition probability distribution defined above $[\Delta((\tilde{s}', s')|(\tilde{s}, s), a)]$, we obtain the following constraints from Eq. (8):

$$y_9 + y_{10} = 0.05y_1 + 0.05y_2 + 0.05y_6 + 0.05y_3, \tag{A5}$$

$$y_{11} + y_{12} = 0.8y_1 + 0.1y_2 + 0.95y_3, \tag{A6}$$

$$y_{13} + y_{14} = 0.1y_1 + 0.8y_2 + 0.95y_6, \tag{A7}$$

$$y_{15} + y_{16} = 0.05y_1 + 0.05y_2 + y_4 + y_5 + y_7 + y_8, \tag{A8}$$

$$y_{17} + y_{18} = 0.05y_9 + 0.05y_{10} + 0.05y_{14} + 0.05y_{11}, \tag{A9}$$

$$y_{19} + y_{20} = 0.8y_9 + 0.1y_{10} + 0.95y_{11}, \tag{A10}$$

$$y_{21} + y_{22} = 0.1y_9 + 0.8y_{10} + 0.95y_{14}, \tag{A11}$$

$$y_{23} + y_{24} = 0.05y_9 + 0.05y_{10} + y_{12} + y_{13} + y_{15} + y_{16}, \tag{A12}$$

$$y_{25} + y_{26} = 0.05y_{17} + 0.05_{18} + 0.05_{22} + 0.05y_{19}, \tag{A13}$$

$$y_{27} + y_{28} = 0.8y_{17} + 0.1y_{18} + 0.95y_{19}, \tag{A14}$$

$$y_{29} + y_{30} = 0.1y_{17} + 0.8y_{18} + 0.95y_{22}, \tag{A15}$$

$$y_{31} + y_{32} = 0.05y_{17} + 0.05y_{18} + y_{20} + y_{21} + y_{23} + y_{24}. \tag{A16}$$

The following relations are useful:

$$y(T, P) = y_{29} + y_{30},$$
$$y(T, P') = y_{27} + y_{28}.$$

Equations (2)–(6) then lead to the following constraints:

$$0 \leq V_{\text{PS}} \leq B, \tag{A17}$$

$$B - 1 \leq V_{\text{PS}} - y_{27} - y_{28} \leq 0, \tag{A18}$$

$$B(1 + \epsilon) + 1 \leq y_{27} + y_{28} - y_{29} - y_{30} \leq B(1 + \epsilon) - \epsilon, \tag{A19}$$

$$B \text{ is a binary}, \tag{A20}$$

$$\text{all } y \text{ are non-negative}. \tag{A21}$$

The standard form of the MIP optimization problem for this example is

$$\max_{B, y, V_{\text{PS}}} V_{\text{PS}} \text{ subject to Eqs. (A1)–(A21)}.$$

Using the GUROBI optimizer MIP solver, we obtain the solution to this optimization problem as $V_{\text{PS}} = 0.7689$, $B = 1$, with the values of the y parameters listed in Table V. This solution describes the best possible way to satisfy the preference $p := \{1\} \succeq \{2\}$. Following this policy for each state pair at each time step, it can be ensured that 76.89% of the time the preference will be satisfied—the highest satisfaction possible for this problem.

TABLE V. The solution for the MIP problem of *Example 3*: $V_{\text{PS}} = 0.7689$ and $B = 1$.

t	$(\tilde{s}, s)$	A	B
0	(0,0)	$y_1 = 0.9$	$y_2 = 0$
0	(1,1)	$y_3 = 0.05$	$y_4 = 0$
0	(2,2)	$y_5 = 0$	$y_6 = 0.05$
0	(3,3)	$y_7 = 0$	$y_8 = 0$
1	(0,0)	$y_9 = 0.05$	$y_{10} = 0$
1	(1,1)	$y_{11} = 0.7675$	$y_{12} = 0$
1	(2,2)	$y_{13} = 0$	$y_{14} = 0.1375$
1	(3,3)	$y_{15} = 0$	$y_{16} = 0.0450$
2	(0,0)	$y_{17} = 0.0478$	$y_{18} = 0$
2	(1,1)	$y_{19} = 0.7691$	$y_{20} = 0$
2	(2,2)	$y_{21} = 0.1356$	$y_{22} = 0$
2	(3,3)	$y_{23} = 0$	$y_{24} = 0.0475$
3	(0,0)	$y_{25} = 0$	$y_{26} = 0.0408$
3	(1,1)	$y_{27} = 0$	$y_{28} = 0.7689$
3	(2,2)	$y_{29} = 0$	$y_{30} = 0.0048$
3	(3,3)	$y_{31} = 0$	$y_{32} = 0.1855$

Note that, at each RL time step, based on the attacker's policy and the defender's resource allocation, a new MIP problem is constructed and solved in order to find the optimal resource allocation.

APPENDIX B: EFFECTS OF FLUCTUATIONS INDUCED BY RENEWABLES ON PREFERENTIAL DEFENSE OF POWER GRIDS

Recent years have witnessed increasing integration of renewable energy sources, such as solar, wind, and hydroelectric power, into power grids worldwide, in efforts to reduce greenhouse-gas emissions and combat climate change. From the point of view of grid management, the incorporation of the renewable sources into a power grid introduces a new set of challenges due to their inherent intermittency and variability. It is thus important to study the effects of fluctuations induced by renewables on the power-grid dynamics. In the context of preferential defense here, we focus on elucidating the specifics of these fluctuations and their influence on the temporal-language MIP strategy for preference satisfaction.

In general, for a fixed allocation of resources for the power grid under consideration under certain preference, the inherent variability of renewable energy generation, driven by factors such as weather patterns and diurnal cycles, will result in fluctuations in power output. Our goal is to understand, given specific actions, how these fluctuations affect the transition function, denoted as $\Delta((\tilde{s}', s')|(\tilde{s}, s), a)$, which characterizes the probability of transitioning from one state to another. To effectively model the fluctuations in the power grid, we use random perturbations by treating these fluctuations as probabilistic deviations from the expected values. This probabilistic framework has allowed us to incorporate the inherent uncertainty associated with renewable energy sources and better mimic real-world scenarios. An illustrative example of the new transition function affected by the added perturbations for the automaton of *Example 3* is

$$\begin{aligned}
&\Delta((1,1)|(0,0),A) = 0.7,\\
&\Delta((2,2)|(0,0),A) = 0.2,\\
&\Delta((3,3)|(0,0),A) = 0.15,\\
&\Delta((0,0)|(0,0),A) = 0.05,\\
&\Delta((1,1)|(0,0),B) = 0.2,\\
&\Delta((2,2)|(0,0),B) = 0.7,\\
&\Delta((3,3)|(0,0),B) = 0.08,\\
&\Delta((0,0)|(0,0),B) = 0.02,\\
&\Delta((0,0)|(1,1),A) = 0.15,\\
&\Delta((1,1)|(1,1),A) = 0.85,\\
&\Delta((3,3)|(1,1),B) = 1,
\end{aligned}$$

$$\begin{aligned}
&\Delta((3,3)|(2,2),A) = 1,\\
&\Delta((0,0)|(2,2),B) = 0.15,\\
&\Delta((2,2)|(2,2),B) = 0.85,\\
&\Delta((3,3)|(3,3),A) = 1,\\
&\Delta((3,3)|(3,3),B) = 1.
\end{aligned}$$

Based on the transition function, we reconstruct the MIP formulation to find the new solution that accounts for preference satisfaction in the presence of fluctuating perturbations, as displayed in Table VI.

TABLE VI. The solution for the extended problem of *Example 3*: $V_{\mathrm{PS}} = 0.6177$ and $B = 1$.

t	$(\tilde{s}, s)$	A	B
0	(0,0)	$y_1 = 0.9$	$y_2 = 0$
0	(1,1)	$y_3 = 0.05$	$y_4 = 0$
0	(2,2)	$y_5 = 0$	$y_6 = 0.05$
0	(3,3)	$y_7 = 0$	$y_8 = 0$
1	(0,0)	$y_9 = 0.06$	$y_{10} = 0$
1	(1,1)	$y_{11} = 0.6725$	$y_{12} = 0$
1	(2,2)	$y_{13} = 0$	$y_{14} = 0.2225$
1	(3,3)	$y_{15} = 0$	$y_{16} = 0.1350$
2	(0,0)	$y_{17} = 0.1373$	$y_{18} = 0$
2	(1,1)	$y_{19} = 0.6136$	$y_{20} = 0$
2	(2,2)	$y_{21} = 0$	$y_{22} = 0.2011$
2	(3,3)	$y_{23} = 0$	$y_{24} = 0.1440$
3	(0,0)	$y_{25} = 0$	$y_{26} = 0.1291$
3	(1,1)	$y_{27} = 0$	$y_{28} = 0.6177$
3	(2,2)	$y_{29} = 0$	$y_{30} = 0.1984$
3	(3,3)	$y_{31} = 0$	$y_{32} = 0.1646$

Our analysis indicates that the impact of the random fluctuations on the power-grid dynamics is analogous to the effect of changing resource allocations in the MIP equations. This suggests that the same temporal-language MIP strategy, previously applied to conventional power grids, can be seamlessly deployed in power grids incorporating renewables.

[1] S. Sridhar, A. Hahn, and M. Govindarasu, Cyber-physical system security for the electric power grid, Proc. IEEE **100**, 210 (2012).

[2] U.S. Department of Energy (DOE), A systems view of the modern grid, National Energy Technology Laboratory (NETL) (2007).

[3] C.-C. Sun, A. Hahn, and C.-C. Liu, Cyber security of a power grid: State-of-the-art, Int. J. Elec. Power Energy Sys. **99**, 45 (2018).

[4] K. R. Davis, C. M. Davis, S. A. Zonouz, R. B. Bobba, R. Berthier, L. Garcia, and P. W. Sauer, A cyber-physical modeling and assessment framework for power grid infrastructures, IEEE Trans. Smart Grid **6**, 2464 (2015).

[5] P. Pourbeik, P. S. Kundur, and C. W. Taylor, The anatomy of a power grid blackout—root causes and dynamics of recent major blackouts, IEEE Power Energy Mag. **4**, 22 (2006).

[6] G. Liang, S. R. Weller, J. Zhao, F. Luo, and Z. Y. Dong, The 2015 Ukraine blackout: Implications for false data injection attacks, IEEE Trans. Power Sys. **32**, 3317 (2017).

[7] R. Langner, Stuxnet: Dissecting a cyberwarfare weapon, IEEE Secu. Priv. **9**, 49 (2011).

[8] J. Xie, A. Stefanov, and C.-C. Liu, Physical and cyber security in a smart grid environment, Wiley Interdiscip. Rev.: Energy Environ. **5**, n/a (2016).

[9] J. A. Baier and S. A. McIlraith, Planning with preferences, AI Magazine **29**, 25 (2008).

[10] J.-J. Wang, Y.-Y. Jing, C.-F. Zhang, and J.-H. Zhao, Review on multi-criteria decision analysis aid in sustainable energy decision-making, Renewable Sustainable Energy Rev. **13**, 2263 (2009).

[11] M. Y. Vardi, in *Logics for Concurrency: Structure versus Automata*, edited by F. Moller and G. Birtwistle (Springer-Verlag, Berlin, 1996), p. 238.

[12] A. Priyanshu, K. Suren, R. Julian, C. Jason, K. Venkat, and A. Nisar, in *Robotics: Science and Systems VIII* (The MIT Press, Cambridge, Massachusetts, 2013), p. 449.

[13] T. Tomita, A. Ueno, M. Shimakawa, S. Hagihara, and N. Yonezaki, Safraless LTL synthesis considering maximal realizability, Acta Informatica **54**, 655 (2017).

[14] X. C. Ding, S. L. Smith, C. Belta, and D. Rus, MDP optimal control under temporal logic constraints, arXiv:1103.4342 (2011).

[15] M. Svorenova, I. Cerna, and C. Belta, Optimal control of mdps with temporal logic constraints, arXiv:1303.1942 (2013).

[16] B. Lacerda, D. Parker, and N. Hawes, in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Institute of Electrical and Electronics Engineers (IEEE), Chicago, Illinois, 2014), p. 1511.

[17] A. Sathi, Cooperation through constraint directed negotiation: Study of resource reallocation problems (1988).

[18] B. Srinidhi, J. Yan, and G. K. Tayi, Allocation of resources to cyber-security: The effect of misalignment of interest between managers and investors, Decision Supp. Sys. **75**, 49 (2015).

[19] A. Sokri, Optimal resource allocation in cyber-security: A game theoretic approach, Procedia Comput. Sci. **134**, 283 (2018).

[20] M. Wang, B. Liu, and H. Xu, in *2019 28th Wireless and Optical Communications Conference (WOCC)* (Institute of Electrical and Electronics Engineers (IEEE), Beijing, China, 2019), p. 1.

[21] L. L. Njilla, C. A. Kamhoua, K. A. Kwiat, P. Hurley, and N. Pissinou, in *2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE)* (Institute of Electrical and Electronics Engineers (IEEE), Singapore, Singapore, 2017), p. 49.

[22] A. E. Motter and Y.-C. Lai, Cascade-based attacks on complex networks, Phys. Rev. E **66**, 065102 (2002).

[23] Y. Chen, S. Huang, F. Liu, Z. Wang, and X. Sun, Evaluation of reinforcement learning-based false data injection attack to automatic voltage control, IEEE Trans. Smart Grid **10**, 2158 (2019).

[24] J. Yan, H. He, X. Zhong, and Y. Tang, Q-learning-based vulnerability analysis of smart grid against sequential topology attacks, IEEE Trans. Inf. Forensics Secur. **12**, 200 (2017).

[25] Z. Wang, H. He, Z. Wan, and Y. Sun, Coordinated topology attacks in smart grid using deep reinforcement learning, IEEE Trans. Ind. Inform. **17**, 1407 (2020).

[26] C. Roberts, S.-T. Ngo, A. Milesi, S. Peisert, D. Arnold, S. Saha, A. Scaglione, N. Johnson, A. Kocheturov, and D. Fradkin, in *2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)* (IEEE, Virtual Conference, 2020), p. 1.

[27] N. I. Haque, M. H. Shahriar, M. G. Dastgir, A. Debnath, I. Parvez, A. Sarwat, and M. A. Rahman, Machine learning in generation, detection, and mitigation of cyberattacks in smart grid: A survey, arXiv preprint arXiv:2010.00661 (2020).

[28] M. Moradi, Y. Weng, and Y.-C. Lai, Defending smart electrical power grids against cyberattacks with deep *Q*-learning, PRX Energy **1**, 033005 (2022).

[29] M. N. Kurt, O. Ogundijo, C. Li, and X. Wang, Online cyber-attack detection in smart grid: A reinforcement learning approach, IEEE Trans. Smart Grid **10**, 5174 (2019).

[30] Y. Li and J. Wu, Low latency cyberattack detection in smart grids with deep reinforcement learning, Available at SSRN 4019864 (2022).

[31] B. Ning and L. Xiao, in *2021 40th Chinese Control Conference (CCC)* (IEEE, Shanghai, China, 2021), p. 8598.

[32] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, Human-level control through deep reinforcement learning, Nature **518**, 529 (2015).

[33] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, Playing Atari with deep reinforcement learning, arXiv:abs/1312.5602 (2013).

[34] Z. Xu and U. Topcu, Transfer of temporal logic formulas in reinforcement learning, IJCAI Proc. **28**, 4010 (2019).

[35] L. Hammond, A. Abate, J. Gutierrez, and M. Wooldridge, Multi-agent reinforcement learning with temporal logic specifications, arXiv:2102.00582 (2021).

[36] X. Li, C. Vasile, and C. Belta, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Institute of Electrical and Electronics Engineers (IEEE), Vancouver, Canada, 2017), p. 3834.

[37] M. Wen, R. Ehlers, and U. Topcu, Correct-by-synthesis reinforcement learning with temporal logic constraints, arXiv:1503.01793 (2015).

[38] A. K. Bozkurt, Y. Wang, M. M. Zavlanos, and M. Pajic, in *2020 IEEE International Conference on Robotics and Automation (ICRA)* (Institute of Electrical and Electronics Engineers (IEEE), Virtual Conference, 2020), p. 10349.

[39] M. Hasanbeig, Y. Kantaros, A. Abate, D. Kroening, G. J. Pappas, and I. Lee, Reinforcement learning for temporal logic control synthesis with probabilistic satisfaction guarantees, arXiv:1909.05304 (2019).

[40] J. Fu, in *2021 American Control Conference (ACC)* (IEEE, Virtual Conference, 2021), p. 4854.

[41] M. O. Rabin and D. Scott, Finite automata and their decision problems, IBM J. Res. Develop. **3**, 114 (1959).

[42] J. E. Hopcroft, R. Motwani, and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation (3rd Edition)* (Addison-Wesley Longman Publishing Co., Inc., Boston, United States, 2006).

[43] N. Pisaruk, *Mixed Integer Programming: Models and Methods* (Belarus State University, 2019).

[44] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (The MIT Press, Cambridge, Massachusetts, 2018), 2nd ed.

[45] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, in *Proceeddings of the 33rd International Conference on Machine Learning Research 2016*, Vol. 48 (PMLR, New York, New York, 2016), p. 1928.

[46] R. J. Williams, Simple statistical gradient-following algorithms for connectionist reinforcement learning, Mach. Learn. **8**, 229 (1992).

[47] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, Proximal policy optimization algorithms, arXiv:1707.06347 (2017).

[48] D. S. H. van Hasselt and A. Guez, in *Proceeddings of the thirtieth AAAI Conference on Artifical Intelligence* (AAAI Press, Phoenix, Arizona, 2016), p. 1928.

[49] Gurobi Optimization, LLC, GUROBI Optimizer Reference Manual (2023).

[50] P. Rezaei, P. D. H. Hines, and M. J. Eppstein, Estimating cascading failure risk with random chemistry, IEEE Trans. Power Syst. **30**, 2726 (2015).

[51] M. J. Eppstein and P. D. H. Hines, A "random chemistry" algorithm for identifying collections of multiple contingencies that initiate cascading failure, IEEE Trans. Power Syst. **27**, 1698 (2012).

[52] D. Witthaut, F. Hellmann, J. Kurths, S. Kettemann, H. Meyer-Ortmanns, and M. Timme, Collective nonlinear dynamics and self-organization in decentralized power grids, Rev. Mod. Phys. **94**, 015005 (2022).

[53] L. Wei, A. H. Moghadasi, A. Sundararajan, and A. I. Sarwat, in *2015 10th System of Systems Engineering Conference (SoSE)* (Institute of Electrical and Electronics Engineers (IEEE), San Antonio, Texas, 2015), p. 12.

[54] A. Marot, B. Donnot, G. Dulac-Arnold, A. Kelly, A. O'Sullivan, J. Viebahn, M. Awad, I. Guyon, P. Panciatici, and C. Romero, in *NeurIPS 2020 Competition and Demonstration Track* (PMLR, Virtual Conference, 2021), p. 112.