Physics-informed Evolutionary Strategy for Adaptive Real-time Voltage Stability Control

Yan Du, Qiuhua Huang, Renke Huang, Tianzhixi Yin, Jie Tan, Wenhao Yu, Xinya Li

***Abstract*—In this work we propose a novel data-driven, real-time power system voltage stability control method based on the physics-informed guided meta evolutionary strategy (ES). The main objective is to quickly provide an adaptive control strategy to secure system voltage stability. The problem is challenging due to the high-dimensional feature of the power system model and the fast-changing and uncertain nature of power system operation scenarios. To this end, a model-free and derivative-free guided ES method is applied. The method is further combined with a meta-learning strategy to make the learnt control policy automatically adapted to unseen operation conditions and fault scenarios, which is highly desired for real-time emergency control. Last but not least, physical knowledge is embedded in the above method through a trainable action mask technique to rule out unnecessary load shedding actions for better learning and control performance. Case studies on the IEEE 300-bus system and comparisons with other state-of-the-art benchmark methods verify the superiority of the proposed physics-informed guided meta ES method in realizing fast and adaptive power system voltage stability control.**

***Index Terms*—Action mask, evolutionary strategy, meta optimization strategy, physics-informed, voltage control**

I. Introduction

INITIALLY brought into the spotlight by the unprecedented success of AlphaGo in year 2016, the deep reinforcement learning (deep RL) technique [1] has been motivating breakthroughs in a broad range of areas including games, robotics, and autonomous driving. In the field of power systems, the deep RL technique has been leveraged for solving complex grid control and optimization problems, such as autonomous voltage regulation [2], residential HVAC control [3], electricity market bidding [4], and power system stability and emergency control [5]–[7]. The major advantages of the deep RL method over the conventional model-based method, as has been discussed thoroughly in the above existing research works, lie in that it is model-free and requires no prerequisite knowledge; it can conduct efficient exploration in high-dimensional continuous search space; and it has generalization to unseen instances.

Nevertheless, there still exist a number of critical factors that prohibit the full adoption of the deep RL technique in the physical systems: the method has a costly training and fine-tuning process due to its numerous embedded parameters; it cannot be easily scaled up for high-dimensional state and action spaces, which fails for large-scale test systems; the increasing assortment and complexity of the algorithm makes it difficult to choose and to deploy.

Driven by the above concerns, in RL community there has been a rising trend for applying evolutionary strategy (ES) as a scalable alternative to the RL method [8], [9]. Unlike the well-known deep RL methods that rely on chain-rules and backpropagation to update the parameters of the neural networks, in ES the parameters are randomly sampled and then optimized along these random directions. The key advantages of the ES algorithm, if compared with the deep RL method, are that the former is derivative-free and can be easily parallelized, both of which lead to remarkable computational efficiency improvement without sabotaging the learning performance.

Most recently, a novel guided ES method is proposed to enhance the exploration efficiency of the algorithm in high-dimensional parameter spaces [10]. Instead of conducting a complete random search, the guided ES method leverages the guidance from a surrogate gradient, which is correlated to the true gradient but gets unbiased in some unknown fashions. By coordinating the random search with the gradient-guided search, the guided ES method demonstrates a faster learning speed and achieves better solutions.

Inspired by the above work, also building upon our previous work in leveraging ES for power system emergency control [11], in this paper we propose a model-free guided ES-based control strategy for securing power system voltage stability with high exploration efficiency. The free from computationally intensive back-propagation process and easy parallelization of the guided ES method makes it possible to overcome the extremely high-dimensional state/action complexities introduced by large-scale power systems.

One key issue with the power system voltage stability control problem is that how well a developed control policy can be adapted to the ever-changing grid operation conditions [12]. To achieve a fast adaptation of the learnt control strategy to unseen fault scenarios to meet with the real-time control requirement, we further combine the above guided ES method with a meta-learning strategy introduced in our previous work [13], namely the meta strategy optimization (MSO), which leads to guided meta ES. The core idea behind MSO is to learn a latent variable as a representation of the variations of the training environments. The latent variable can then be fine-tuned when a new fault scenario is presented, and the control policy is adjusted accordingly. The robustness and adaptability of the proposed guided meta ES method makes it practical for real-world implementations.

The final highlight of our proposed method is the introduction of the physics knowledge for further accelerating the learning process. Physics-informed machine learning has received growing attention lately due to the challenges encountered by the pure data-driven machine learning methods, including high cost of data acquisition, data incompleteness,

TABLE I
TECHNICAL ROADMAP OF THE PHYSICS-INFORMED ES

Key challenges of deep RL	Proposed technique
Costly back-propagation process; Laborious parameter fine-tuning; Lack of scalability; Increasing algorithm complexity	Derivative-free ES for easy parallelization and low computational burden (in our previous work[11])
Lack of adaptability and generalization to unseen test cases	Meta strategy optimization for fast policy adaptation (in our previous work [13])
Time-consuming random action-space exploration	Guiding exploration in the parameter space with surrogate gradient to focus on promising directions (developed in this paper)
High-dimensional action domains leading to exhaustive searching	Introducing physical knowledge through TAM for effective action filtering (developed in this paper)

and extremely high search spaces, etc [14]. In the problem of power system voltage stability control, the bulk grid with hundreds of thousands of buses induces a vast control action space and an unduly burdensome searching process. To overcome the above difficulty, we import a physics-informed module called trainable action mask (TAM), which utilizes the prior physical knowledge to filter out improper control actions and to avoid unnecessary explorations. Previous studies in the dialog system [15] have validated the effectiveness of TAM in boosting the learning performance of RL in large domains. In this work, we initiatively combined the physical knowledge with TAM for the voltage stability control problem and got promising results. More details of the action mask will be revealed in the following sections.

In summary, the key contributions of our work can be outlined as follows:

1) A novel model-free guided meta ES method is developed to intelligently conduct load shedding to stabilize the bulk power system voltage level after fault occurrence. The combination of random search with surrogate gradient information endows the guided ES method with superior computational efficiency than the standard ES algorithm. The generalization and adaptability of the learnt control policy to unseen fault scenarios is further enhanced through a meta learning strategy.

2) To further improve the exploration efficiency of the algorithm, physical knowledge is introduced through a TAM technique, which eliminates improper load shedding actions and spares the exploration efforts. The embedding of physics awareness considerably promotes the algorithm performance in large action domains.

3) The adaptability and efficiency of the proposed voltage stability control method based on the physics-informed guided meta ES algorithm is fully verified by testing on a large-scale power system under multiple unseen fault scenarios and by comparing with state-of-the-art benchmark methods, which implies its great promises for real-time applications.

Table I further presents an overview of the proposed technical roadmap in our work.

The rest of the paper is organized as follows: Section II describes the problem formulation of the power system voltage stability control. Section III introduces the proposed adaptive model-free control method based on guided meta ES algorithm. In Section IV, the physics knowledge is further combined with the proposed control method through trainable action mask. Case studies are shown in Section V. Finally, Section VI concludes the work and the future directions.

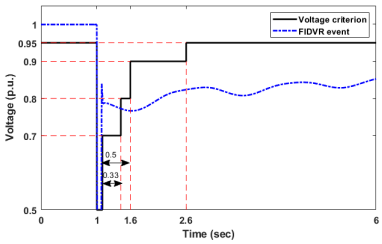

Fig. 1. Transient voltage recovery criterion

II. PROBLEM FORMULATION

One widely applied power system emergency control measure in industries for maintaining the voltage stability after the fault occurs is through load shedding. An ideal load shedding strategy should be able to bring the system voltage magnitude to a certain level with minimum amount of load shedding. A standard transient voltage recovery criterion is shown in Fig. 1 [16]. As shown in the figure, the voltage should return to at least 0.7, 0.8, and 0.95 p.u. within 0.33s, 0.5s and 1.5s after the fault is cleared. Deciding the optimal load shedding strategy is not a trivial task since three crucial problems must be considered: when to conduct load shedding, at which bus the load should be shed, and how much of the load should be shed, which leads to a high-dimensional non-convex decision-making problem [6] and fails many model-based solutions in the case of real-time implementation. Also, the model-based solution is extremely sensitive to the model inputs and cannot be adapted to unseen fault scenarios.

Based on the above discussions, in this work we propose to formulate the power system voltage stability control problem as a Markov Decision Process (MDP) and further apply a model-free guided meta ES algorithm with trainable action mask (TAM) to obtain the optimal load shedding strategies under different fault scenarios, which is adaptive, highly scalable, and computationally efficient. The MDP-based problem formulation is first presented as follows.

The MDP is represented as a 5-tuple$(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$, and their definitions under the context of power system voltage stability control are provided as follows:

1) **state**: the state $\mathcal{S}$ is defined as a vector that contains the latest observations from the power system, including the voltage magnitude at the monitored buses and the percentage of remaining load that can be shed at the controllable buses: $s_t = [V_{1,t}, ..., V_{M,t}, P^d_{1,t}, ..., P^d_{N,t}]$

2) **action**: the action $\mathcal{A}$ is defined a vector that contains the normalized load shedding actions for all the controllable buses. The normalized load shedding action is a scalar between -1 and 1, where -1 indicates that 20% of the remaining load will be shed, and 1 indicates no load shedding actions.

3) **state transition**: the state transition $\mathcal{P}$ describes the power system dynamics and is deterministically governed by a set of differential and algebraic equations:

$$\dot{x}_t = f(x_t, y_t, d_t, a_t) \tag{1}$$

$$0 = g(x_t, y_t, d_t, a_t) \tag{2}$$

In (1)-(2), x_t represents the system dynamic state variables, such as the generator rotor angle and speed; y_t represents the system algebraic state variables, which are usually bus voltage magnitudes and bus voltage angles; d_t refers to the system perturbation or contingency; and a_t is the emergency control action.

4) **reward**: in the power system emergency control problem, the main objective is to restore the voltage magnitude to the normal level after fault clearance with the least amount of load shedding. To reach this objective, we adopted the same reward design as in our previous work [6]:

$$r_t = \begin{cases} -10000,\, if\, V_{i,t} < 0.95,\, T_{pf} + 4 < t \\ c_1 \sum_i \Delta V_{i,t} - c_2 \sum_j \Delta P_{j,t}(p.u.) - c_3 u_{ivld}, \\ \text{otherwise} \end{cases} \tag{3}$$

where

$$\Delta V_{i,t} = \begin{cases} \min\left\{V_{i,t} - V_{th,1}, 0\right\},\, if\, T_{pf} < t < T_{pf} + t_1 \\ \min\left\{V_{i,t} - V_{th,2}, 0\right\},\, if\, T_{pf} + t_1 < t < T_{pf} + t_2 \\ \min\left\{V_{i,t} - V_{th,3}, 0\right\},\, if\, T_{pf} + t_2 < t \end{cases} \tag{4}$$

In (3)-(4), t is the current time step;$V_{i,t}$ is the voltage magnitude at bus i;T_{pf} is the time instant for fault clearance; $\Delta P_{j,t}$ is the amount of shed load at bus j in p.u.; u_{ivld} is the penalty for invalid action if there is still a load shedding action at buses with zero remaining load; c_1,c_2, and c_3 are the weight factors; $V_{th,1}$, $V_{th,2}$,$V_{th,3}$, t_1,and t_2 constitute the voltage recovery criterion. One example of their values has been shown in Fig. 1.

5) **discount factor**: the objective of MDP is to maximize the following total reward $R_t = \sum_{t'=t}^{T} \gamma^{t'-t} r_{t'}$,where γ is a discount factor between 0 and 1. The reason for adding the discounted factor is to avoid an infinite sum of future rewards.

III. Guided Meta Evolutionary Strategy

In this section, we will first give a brief review of the evolutionary strategy (ES). Then we will introduce the guided ES algorithm that combines the surrogate gradient with random search to achieve higher sampling efficiency. Lastly, we will present an adaptive guided ES algorithm by utilizing meta strategy optimization (MSO) to obtain more flexible control strategies.

A. An introduction to ES

The ES is a type of heuristic search algorithm inspired by the evolution theory: at each iteration, a population of parameters that need to be optimized are randomly perturbed and their objective function values are calculated. The parameters with the highest values are then recombined to formulate the population for the next iteration. The process repeats until the objective meets the convergence criterion.

In the context of RL, given the reward function r and the policy $\pi(s|\theta)$, the goal is to find the optimal θ that maximizes the expected total discounted reward $E\{\sum_{t=0}^{T} \gamma^t r_t(s_t, \pi(s_t|\theta))\}$. **Algorithm 1** shows the implementation of ES [8]:

Algorithm 1 Evolutionary Strategy (ES)

1: Initialize the learning rate η, noise standard deviation σ, the number of perturbation directions N, and policy parameter θ
2: **for** iteration t = 1 **to** M **do**
3: Sample perturbation directions $\epsilon_1, ..., \epsilon_N$ from $\mathcal{N}$(0,I)
4: **for** i = 1 **to** N **do**
5: Generate action $a_i = \pi(s|\theta_t + \sigma\epsilon_i)$
6: Execute a_i and receive reward r_i
7: **end for**
8: Update the policy parameter:
9: $\theta_{t+1} = \theta_t + \eta \frac{1}{N\sigma} \sum_{i=1}^{N} r_i \epsilon_i$
10: **end for**

As shown in the above pseudo code, the algorithm consists of two repeated phases: first, the policy parameter is randomly perturbed by noises derived from a standard normal distribution, and the associated actions are executed and evaluated based on their reward values for an entire episode (line 3-line 7); second, the policy parameter is updated by an estimated stochastic gradient (line 9), which comes from the following derivation: assuming our objective is to optimize θ over a distribution $p_\psi(\theta)$ to maximize the expected reward $\mathbb{E}_{\theta \sim p_\psi(\theta)} r(\theta)$, when the parameter distribution $p_\psi(\theta)$ follows a Guassian distribution, the expected reward can be directly written as $\mathbb{E}_{\theta \sim p_\psi(\theta)} r(\theta) = \mathbb{E}_{\epsilon \sim N(0,I)} r(\theta + \sigma\epsilon)$. With the objective defined in terms of θ, the gradient can be calculated as follows:

$$\nabla \mathbb{E}_{\epsilon \sim N(0,I)} r(\theta + \sigma\epsilon) = \frac{1}{\sigma} \mathbb{E}_{\epsilon \sim N(0,I)} \left\{ r(\theta + \sigma\epsilon)\epsilon \right\} \tag{5}$$

The expectation term in (5) can be achieved through sampling, as shown by line 3 in the algorithm. Note that line 4-line 7 can be naturally deployed in a parallel fashion to speed up the training, since each perturbation direction ϵ_i is independent from each other. The simple way of sampling instead of back-propagation for parameter update makes the ES algorithm more scalable to distributed computer systems than the gradient-based RL methods.

B. Guiding ES search with surrogate gradient

In the above ES algorithm, the policy parameter θ is randomly perturbed following a Gaussian distribution. While this random search is easy to implement, it can introduce high variance and results in unnecessary explorations. The guided ES algorithm is thus proposed to handle this challenge. The core idea behind the guided ES algorithm is to refer to the surrogate gradient to guide the algorithm search toward the most promising directions instead of conducting a completely random search.

A surrogate gradient is correlated with the true gradient, but somehow biased or corrupted due to the model unobservability. An illustration of the surrogate gradient is shown in Fig. 2. The guided ES algorithm takes advantage of the surrogate gradient in the following way [10]: suppose we can get a vector of surrogate gradient for the policy parameters at each iteration, then by collecting the surrogate gradients from the previous k iterations, we can generate a subspace $U^T U = I_k$, where U is an $n \times k$ orthogonal basis for this subspace, and n is the

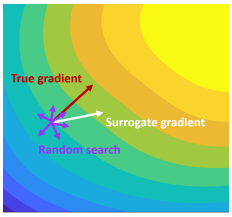

Fig. 2. Schematic of surrogate gradient

dimension of the policy parameters. The gradient information can be further embedded in the ES algorithm by changing the distribution of the perturbation ϵ_i from $\mathcal{N}(0,\mathrm{I})$ to $\mathcal{N}(0,\sum)$, where $\sum$ is calculated as follows:

$$\sum = \alpha^2 I_n + (1-\alpha)^2 U U^T \tag{6}$$

In (6), α is a weight factor that makes a trade-off between the random search (exploration) and the guided search with surrogate gradient (exploitation). Setting $\alpha = 1$ will lead to the ES algorithm. In our case, we set α to 0.5 to balance the exploration with exploitation. With the modified distribution, the perturbation direction ϵ_i can be calculated as follows:

$$\epsilon_i = \alpha\epsilon' + (1-\alpha)\epsilon'' \tag{7}$$

where $\epsilon' \sim \mathcal{N}(0,I_n)$, and $\epsilon'' \sim \mathcal{N}(0,I_k)$. The complete guided ES algorithm is shown in **Algorithm 2**. The algorithm basically follows the same framework as the ES method. One difference is that at the initialization state, a surrogate gradient buffer B is defined to store the surrogate gradients from the previous k steps for generating perturbations. In addition, an antithetic sampling is applied, where for each perturbation direction, a pair of evaluations for $\theta_t + \sigma\epsilon_i$ and $\theta_t - \sigma\epsilon_i$ are conducted to reduce variance, as shown by line 6. The evaluations are later used to calculate the surrogate gradient for policy parameter update, as shown by line 11. Finally, the surrogate gradient is stored in the buffer B for generating the new perturbation distribution, as shown by line 12 in the algorithm.

Algorithm 2 Guided Evolutionary Strategy (guided ES)

1: Initialize the learning rate η, the weight factor α, the scale factor β, the noise standard deviation σ, the number of samples N, the number of surrogate gradients to use k, and the policy parameter θ
2: Initialize the surrogate gradient buffer $B \in \mathbb{R}^{k\times n}$
3: **for** iteration t = 1 **to** M **do**
4: Sample perturbation directions $\epsilon_i, ..., \epsilon_N$ from $\mathcal{N}(0,\sum)$
5: **for** i = 1 **to** N **do**
6: Generate action $a_i = \pi(s|\theta_t + \sigma\epsilon_i), a_i = \pi(s|\theta_t - \sigma\epsilon_i)$
7: Execute a_i and receive reward r_i
8: **end for**
9: Update θ with surrogate gradient g:
10: $g = \frac{\beta}{2\sigma N}\sum_{i=1}^{N}\epsilon_i[r(\theta_t + \sigma\epsilon_i) - r(\theta_t - \sigma\epsilon_i)]$
11: $\theta_{t+1} = \theta_t + \eta g$
12: Store g to the buffer B and update the surrogate gradient subspace U and pertubation distribution $\sum$
13: **end for**

C. Enhancing algorithm adaptability with MSO

The power system has a fast-changing and uncertain nature, which requires that an emergency control strategy should have sufficient robustness and be adaptive to unseen fault scenarios. To enhance the adaptability of the above data-driven guided ES-based control policy to new environment dynamics, in this subsection we propose to integrate the idea of meta learning, namely learning to learn, into the guided ES method, which leads to guided meta ES.

We apply a specific meta-learning technique, the meta strategy optimization (MSO) [17], to realize the above objective. MSO adapts a learnt control policy to unseen scenarios through latent space representation. For each operation scenario encountered during the training, a latent variable is defined for this scenario to encode its hidden features. The latent variable is later combined with the direct observations of the scenario and sent to the policy function for decision-making. The latent variable optimization and the policy parameter update can be expressed by the following two equations:

$$c_{\mu,t} = \arg\max_{c} J_\mu(c,\theta_t) \tag{8}$$

$$\theta_{t+1} = \arg\max_{\theta} E_\mu[J_\mu(c_{\mu,t},\theta)] \tag{9}$$

In (8), $c_{\mu,t}$ is the latent variable associated with scenario μ at the t^{th} iteration; $J_\mu(c,\theta_t)$ is a performance measurement, e.g., the reward function. The policy parameter θ is then optimized by maximizing the expected performance measurement with the learnt latent variable $c_{\mu,t}$, as shown by (9).

When unseen operation scenarios occur during the testing, new latent variables can be calculated through the above process for fine-tuning the policy, making it adapted to the new environment dynamics. This adaptation can be realized through only a few iterations with the environment, which is highly time-efficient. More technical details of MSO application in power system emergency control can be found in our previous work [13].

D. Advantages of Guided Meta ES algorithm over RL

Under the context of power system voltage stability control, the above guided meta ES algorithm exceeds the gradient-based RL methods in the following three aspects:

1) The algorithm does not require the computationally intensive back-propagation process for gradient calculation and parameter update. For large-scale power systems, usually deep neural networks are constructed for better policy approximation or action value approximation, which results in tedious gradient calculation. The proposed algorithm overcomes this difficulty with the idea of surrogate gradient and greatly spares computational efforts.

2) The algorithm is well suited to scale up to parallel computing: the algorithm operates on complete power system dynamic simulations, which indicates infrequent communications among the parallel workers. Considering the variety of power system operation scenarios, a parallel simulation greatly facilitates the training process.

3) In the case of unexpected fault scenarios, the learnt control policy can be quickly adjusted through MSO to suit to new

environment dynamics and to mitigate the negative impacts, which is highly desired for real-time voltage stability control.

IV. Physics-informed Guided Meta ES with Trainable Action Mask

In this section, we aim to further improve the exploration efficiency of the guided meta ES method by introducing a novel trainable action mask (TAM) technique, which brings in the physical knowledge of power systems to pinpoint the optimal control actions.

A. Embedding physics knowledge through TAM

while the guided ES algorithm is much better than basic ES algorithms, it still suffers from exploration inefficiency issues when applied to high-dimensional control problems. One way to overcome this obstacle is to incorporate a physics-informed action mask component into the algorithm, which will filter out impossible or unfavorable actions and prevent the algorithm from conducting unnecessary explorations [18].

The action mask makes use of existing physical knowledge. In the case of power system voltage stability control, the voltage stability criterion (such as the one in Fig. 1) can be regarded as prior knowledge, and be used to accelerate the training through a simple hand-crafted action-mask. The action mask can be constructed as a vector that has the same dimension as the control action. Then, at each time step, for each controllable bus, if its observed voltage magnitude is above the stability criterion, no action is required and a zero element will be added to the corresponding position in the mask, and vice versa. Next, the action generated by the policy network will be multiplied by this mask, where the positions with zero elements will eliminate the corresponding load shedding actions since it is unnecessary, and the positions with one will keep the load shedding actions. The physical information introduced by the action mask helps exclude redundant actions.

Note that in the above hand-crafted action mask, the mask settings are set according to a predefined, fixed stability criterion and generally remain fixed for all scenarios. However, considering that the power system operation scenarios can vary significantly from one to another, for instance, with different loading conditions, a fixed mask is unlikely to be the optimal solution for a wide range of operation scenarios.

Based on the above discussions, we propose a TAM technique to develop a learnable, adaptive criterion and to obtain a more flexible and generalized control strategy. An illustrative explanation of the TAM technique is shown in Fig. 3, and it can be described by the following mathematical expressions:

$$[a_t, cr_t] = \pi(s_t, c_t|\theta_t) \tag{10}$$

$$TAM_{i,t} = \begin{cases} 1, \; if \; s_{i,t} < cr_t \\ 0, \; elsewise \end{cases} \quad \forall i \in observable\, buses \tag{11}$$

$$a_t = a_t \odot TAM \tag{12}$$

where a_t and cr_t are the action and the learned criterion based on the current state s_t and the current latent variable c_t. The TAM is generated by comparing the voltage magnitude at each bus i with the voltage criterion cr_t, as shown by (11). The action is filtered by conducting a element-wise multiplication with TAM, as shown by (12).

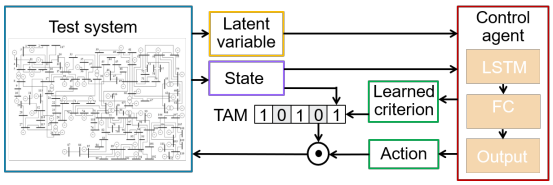

Fig. 3. Illustration of the trainable action mask technique

As can be seen from the above process, at each time step, a specified voltage criterion is generated based on the current state and the operation scenario information provided by the latent variable. Compared with the fixed action mask, the TAM is flexibly adjusted as the states vary, and the control actions are filtered accordingly. In the TAM method, the physical knowledge is introduced by defining an upper bound and a lower bound for the learnable criterion, which reasonably reduces the search space and facilitates the training process.

B. Physics-informed Guided Meta ES with TAM

The complete physics-informed guided meta ES method with TAM for power system voltage stability control is shown in **Algorithm 3**. The algorithm is composed of three major procedures: 1) generate the latent variable (line 3-5); 2) perturb the policy parameters and get the associated rewards (line 6-17); 3) update the policy parameter and the surrogate gradient subspace (line 18-21). The major difference of the above physics-informed guided meta ES method with TAM from the guided meta ES method lies in that in the former method, the policy network will output not only the action but also the voltage criterion, which is later used to mask unnecessary actions, as shown by line 11-13. This simple tweak of the algorithm leads to a remarkable learning performance improvement with negligible extra computational efforts, since the voltage criterion cr_{ijt-} only adds a few additional output dimensions to the policy network. In the next section, the comparative studies will further validate the superiority of the proposed physics-informed ES method.

V. Case Studies

In this section, we will first introduce the test environment for implementing and testing the proposed methods, then we will present the simulation results and comparisons with other state-of-the-art benchmark methods to demonstrate the performance of the proposed methods in terms of training efficiency, RL agent generalization capability, control performance and optimality.

A. Test environment and deployment details

The proposed physics-informed guided meta ES-based learning framework is deployed on a local high performance computing cluster with a Linux operation system of 520 nodes. Each node has a dual-socket Intel Haswell E5-2670V3 CPU with 64 GB DDR4 memory and 12 cores per socket running at 2.3 GHz. The training and testing of the algorithm are performed with IEEE 300-bus system [19]. The power system

Algorithm 3 Physics-informed Guided Meta ES for Power System Emergency Control

Initialize the learning rate η, the decay rate ξ, the weight factor α, the noise standard deviation σ, the total number of perturbation directions N, the number of surrogate gradients to use k, the number of top-performing directions b, the number of power flow cases to simulate for each iteration m, and the policy parameter θ
Initialize the latent variable $\boldsymbol{c}$ for each training power flow case
Initialize the surrogate gradient buffer $B \in \mathbb{R}^{k\times n}$
for iteration t = 1 **to** M **do**
  Sample m power flow cases $\{\mu_j | j = 1, ..., m\}$
  if mod(t, t_{inner})==0 **then**
    Update the latent variable $c_{\mu,t}$ by maximizing J_μ
  end if
  Generate the perturbation direction $\epsilon_1, \epsilon_2, ..., \epsilon_N$:
  $\epsilon_i = \alpha\epsilon^{'} + (1-\alpha)\epsilon^{''}, \epsilon^{'} \sim \mathcal{N}(0, I_n), \epsilon^{''} \sim \mathcal{N}(0, I_k)$
  for i = 1 **to** N **do**
    Generate a pair of policy parameters $\theta_t + \sigma\epsilon_i$ and $\theta_t + \sigma\epsilon_i$
    for j = 1 **to** m **do**
      Generate control action and the learnt criterion $a_{ijt+}, cr_{ijt+} = \pi(s_{ijt+}, c_{j,t}|\theta_t + \sigma\epsilon_i)$ and $a_{ijt-}, cr_{ijt-} = \pi(s_{ijt-}, c_{j,t}|\theta_t - \sigma\epsilon_i)$
      Generate the binary action mask vector based on the observed voltage level in s_{ijt+} and s_{ijt-}
      Apply the action mask to the control actions and collect the rewards r_{tij+} and r_{tij-}
    end for
    Calculate the average reward:
    $r_{ti+} = \frac{1}{m}\sum_{j=1}^{m} r_{tij+}$, $r_{ti-} = \frac{1}{m}\sum_{j=1}^{m} r_{tij-}$
  end for
  Select top b rewards with the largest values and update θ_t with the surrogate gradient:
  $g = \frac{1}{b\sigma}\sum_{i=1}^{b}(r_{ti+} - r_{ti-})\epsilon_i$
  $\theta_{t+1} = \theta_t + \eta g$
  Store g to the buffer B and update the surrogate gradient subspace $UU^T = I_k$, where $U \in \mathbb{R}^{n\times k}$
  Update the learning rate and the noise standard deviation with the decay rate: $\eta = \xi\eta$, $\sigma = \xi\sigma$
end for

dynamic simulation is completed by the open-source platform RLGC [6], [20]. A summary of the hyper-parameters of the algorithm is shown in Table II. Note that the policy network is constructed as a neural network with two hidden layers, one LSTM layer and one fully-connected layer, with each having 32 neurons. The state is defined as a vector with 154 elements, where the first 108 elements are the bus voltage magnitudes, and the last 46 elements are the remaining load levels at the buses with controllable loads. The state vector is further concatenated with a latent context vector with 16 latent variables as the input to the policy network. The output from the policy network is an action vector with 51 elements, where the first 46 elements are the amount of shed load, and the last 5 elements define the learnt voltage criterion, namely $V_{th,1}, V_{th,2}, V_{th,3}, t_1, t_2$ in (4). Based on the physical knowledge, an upper bound and a lower bound are defined for the above 5 fixation points as follows: $V_{th,1}$ ∈[0.7., 0.85] p.u., $V_{th,2}$ ∈[0.85., 0.92] p.u., $V_{th,3}$ ∈[0.92 0.96] p.u., t_1 ∈[0.25, 0.4] s, t_2 ∈[0.4, 0.6] s.

TABLE II
HYPERPARAMETERS FOR GUIDED META ES WITH TAM

Parameters	**300-Bus**
Policy Model	LSTM+FC
Policy Network Size (Hidden Layers)	[32,32]
Weight factor (α)	0.5
Number of Disturbances (N)	128
Top Directions (b)	64
Step Size (η)	1
Std. Dev. of Exploration Noise (σ)	2
Decay Rate (ξ)	0.998

For training the algorithm, 36 operation scenarios are generated, which combines 4 power flow cases (scenarios) with 9 fault scenarios. The power flow scenarios vary in their generation levels and loading levels, and the fault scenarios vary in their fault buses. The fault is assumed to start at 1.0s and ends after 0.1s. For testing the algorithm, 136 operation scenarios are generated, which combines 4 power flow scenarios with 34 fault scenarios. More fault buses are considered during testing. In addition, the fault is assumed to start at 0.5s and ends after 0.08s. The reason for applying new test scenarios is to validate the adaptability of the proposed data-driven control policy. More details of the training cases and test cases can be found at [13].

B. Comparative studies

In this subsection, we will analyze the efficiency and adaptability of the proposed physics-informed guided meta ES method for power system voltage stability control by making comparisons with several benchmark algorithms.

1) Evaluating the guiding gradient

We first compare the performance of the guided ES method with the ES method to evaluate the function of the surrogate gradient in leading the explorations. Fig. 4 presents the training results and the testing results for both methods. For the ES method, we applied the augmented random search (ARS) from our previous work [11], which is an improved version of ES. The figure on the left shows the average reward for 500 training iterations, where the shaded area stands for the standard deviation over 3 random seeds. The reward curve of the guided ES method increases faster than that of the ARS method and reaches a higher converged value with smaller deviation. The two figures on the right show the reward gained in each of the 136 test cases from the two methods. Table III lists the average test reward for the two methods. As shown in the table, the guided ES method improved the average reward by more than 50% compared with the ARS method. We further counted the number of test cases in which the two methods failed to recover the system voltage level, which is indicated by a reward smaller than -10^4. As shown in the third column of Table III, the number of failed cases of the guided ES method is less than 1/4 of that of the ARS method. Therefore, we can safely conclude that the implementation of the surrogate gradient helps improve the exploration efficiency and lead to better control policies.

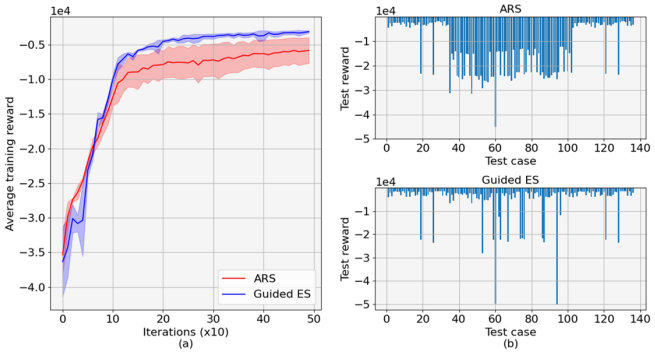

Fig. 4. Comparison of ARS with guided ES:(a) average training reward over 3 random seeds; (b) test reward for 136 new cases.

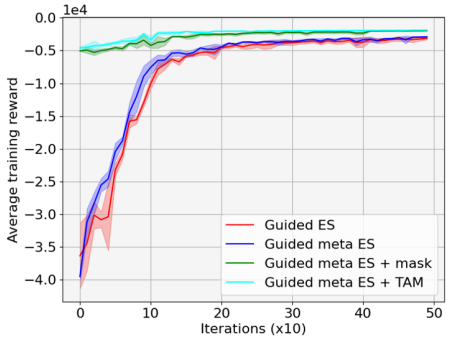

Fig. 5. Comparison of training curves of guided ES methods

TABLE III
COMPARISON OF TEST RESULTS

Method	Average test reward	No. of failed cases
ARS	-1.27×10^4	72
Guided ES	-5.6×10^3	17
Guided meta ES	-4.3×10^3	12
Guided meta ES + mask	-2.8×10^3	8
Guided meta ES + TAM	-1.9×10^3	3

2) Discussion on meta learning and physics awareness

The proposed physics-informed guided meta ES with TAM method is further compared with three other benchmark methods, namely the guided ES method, the guided meta ES method, and the guided meta ES method with mask derived from the fixed voltage criterion. Fig. 5 shows the training curves of the four methods. As shown in the figure, the last two methods with action mask outperform the first two methods with a much higher starting point and also a higher final reward, which validates the function of action mask in speeding up the training process and the importance of utilizing physics knowledge. Fig. 6 shows the test results of the four methods. The average reward for 136 test cases and the associated number of failed cases for the four methods are also listed in Table III. As shown in the table, the proposed guided meta ES method with TAM has the highest average reward and the fewest failed cases. The following two conclusions can be further drawn from the observations of the test results: 1) compared with the pure guided ES method, the guided meta ES method can quickly adapt the learnt control policy to unseen test cases through MSO and achieve better performance; 2) a learnt voltage criterion from TAM works more efficiently in generating masks for action selection than a fixed voltage criterion.

To look deeper into how the TAM helps improve the learning performance, we study one test case in which the first three methods failed and only the guided meta ES with TAM succeeded in restoring the voltage stability. The voltage magnitudes of all observable buses under the four methods are shown in Fig. 7. In total there are 108 voltage curves in each figure. The dashed black voltage envelope stands for the lower security bound of the voltage. For the first three methods, the simulation ends at around 6 seconds. This is because the control policies failed to restore the voltage level and the system went collapsed. For the last method, the simulation

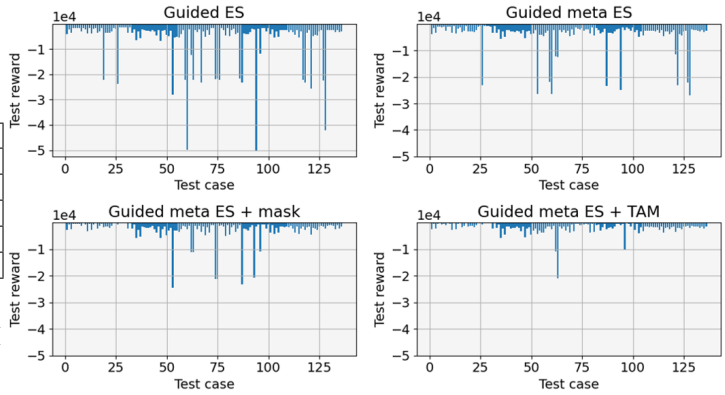

Fig. 6. Test results of the proposed methods and the benchmark methods

lasts for 10 seconds, which is the predefined time length of an entire simulation, and all the bus voltage magnitudes are above the voltage envelope.

We further unveiled the learnt voltage criterion and the control actions from the guided meta ES method. In Fig. 8, the left figure shows the total remaining load for the four methods. The first two methods have relatively lower remaining load, which implies that unnecessary load shedding is conducted without guidance from the physical knowledge. The third method, which utilizes a mask from a fixed voltage criterion, has the highest remaining load level after shedding. Nevertheless, the system voltage level cannot be fully recovered in this case.

The figure on the right compares the learnt voltage criteria from TAM and the fixed voltage criteria. In TAM, for each time step, a voltage criterion is generated. For a complete simulation with a time length of 10s and a time step of 0.1s, there are 100 learnt voltage criteria. We compare the average value of the learnt voltage criteria with the fixed voltage criterion. As can be seen from the figure, following the immediate occurrence of the fault (between 0.6s and 1s), the learnt voltage criterion is higher than the fixed voltage criterion, which explains why a larger amount of load is shed in the guided meta ES method with TAM. In Fig. 7, comparing the two lower figures, it can be observed that with TAM, the voltage rises faster and higher than that with mask after the fault takes place (from 0.6s to 2s), due to the larger amount of load shedding. Therefore, we can safely conclude that a learnt voltage criterion from TAM can lead to more reliable load shedding strategies.

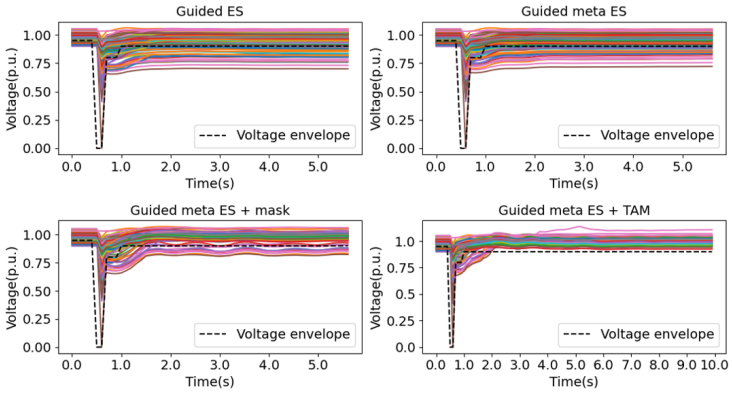

Fig. 7. Comparison of bus voltage under the four control methods

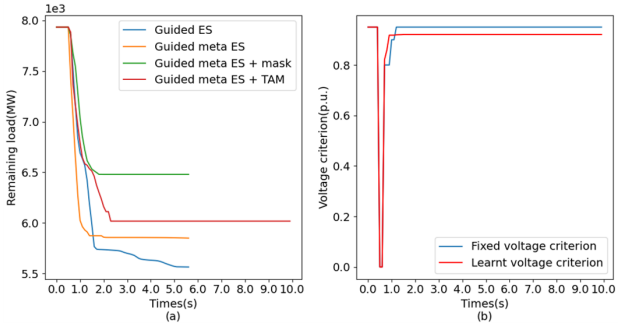

Fig. 8. Comparison of load shedding strategy and mask voltage criterion: (a) total remaining load of the system; (b) voltage criterion.

3) Computation time for implementation

Finally, we compare the computation time of the four methods for determining control actions, and results are shown in Table IV. With total decision-making time of 0.63s for more than 80 action steps, the proposed physics-informed guided meta ES method meets the real-time operation requirement and can provide timely remedial control strategies against fast, short-term voltage stability problems.

VI. Conclusions

In this paper, we propose a novel model-free power system emergency control method built upon the physics-informed guided meta ES algorithm. The proposed algorithm makes use of the surrogate gradient and guidance from physics knowledge to conduct more efficient exploration during the training for better solutions. In addition, the algorithm is combined with meta-learning to gain adaptability and robustness. Simulation results on the IEEE 300-bus system show that the proposed algorithm outperforms other state-of-the-art model-free control algorithms with a faster convergence and a more adaptive control strategy to unseen fault scenarios, and also meets the real-time requirement.

For future research, we will dive deeper into the trade-off between the guided direction and the random search in the guided ES method for better learning performance; also, we will investigate the transferability of the learnt control policy among different test systems, which will lead to more general and practical algorithm implementation.

TABLE IV
Comparison of Testing Time

Method	Guided ES	Guided meta ES	Guided meta ES + mask	Guided meta ES + TAM
Testing time(s)	0.51	0.49	0.62	0.63

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[2] S. Wang, J. Duan, D. Shi, C. Xu, H. Li, R. Diao, and Z. Wang, "A data-driven multi-agent autonomous voltage control framework using deep reinforcement learning," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4644–4654, 2020.

[3] L. Yu, Y. Sun, Z. Xu, C. Shen, D. Yue, T. Jiang, and X. Guan, "Multi-agent deep reinforcement learning for hvac control in commercial buildings," *IEEE Transactions on Smart Grid*, vol. 12, no. 1, pp. 407–419, 2021.

[4] Y. Liang, C. Guo, and H. Hua, "Agent-based modeling in electricity market using deep deterministic policy gradient algorithm," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4180–4192, 2020.

[5] Z. Yan and Y. Xu, "Data-driven load frequency control for stochastic power systems: A deep reinforcement learning method with continuous action search," *IEEE Transactions on Power Systems*, vol. 34, no. 2, pp. 1653–1656, 2018.

[6] Q. Huang, R. Huang, W. Hao, J. Tan, R. Fan, and Z. Huang, "Adaptive power system emergency control using deep reinforcement learning," *IEEE Transactions on Smart Grid*, vol. 11, no. 2, pp. 1171–1182, 2020.

[7] G. Zhang, W. Hu, D. Cao, Q. Huang, J. Yi, Z. Chen, and F. Blaabjerg, "Deep reinforcement learning based approach for proportional resonance power system stabilizer to prevent ultra-low-frequency oscillations," *IEEE Transactions on Smart Grid*, 2020.

[8] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever, "Evolution strategies as a scalable alternative to reinforcement learning," *arXiv preprint arXiv:1703.03864*, 2017.

[9] H. Mania, A. Guy, and B. Recht, "Simple random search provides a competitive approach to reinforcement learning," *arXiv preprint arXiv:1803.07055*, 2018.

[10] N. Maheswaranathan, L. Metz, G. Tucker, D. Choi, and J. Sohl-Dickstein, "Guided evolutionary strategies: Augmenting random search with surrogate gradients," in *International Conference on Machine Learning*, 2019, pp. 4264–4273.

[11] R. Huang, Y. Chen, T. Yin, X. Li, A. Li, J. Tan, W. Yu, Y. Liu, and Q. Huang, "Accelerated deep reinforcement learning based load shedding for emergency voltage control," *arXiv preprint arXiv:2006.12667*, 2020.

[12] B. Park and M. M. Olama, "A model-free voltage control approach to mitigate motor stalling and fidvr for smart grids," *IEEE Transactions on Smart Grid*, 2020.

[13] R. Huang, Y. Chen, T. Yin, Q. Huang, J. Tan, W. Yu, X. Li, A. Li, and Y. Du, "Learning and fast adaptation for grid emergency control via deep meta reinforcement learning," *arXiv preprint arXiv:2101.05317*, 2021.

[14] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations," *arXiv preprint arXiv:1711.10561*, 2017.

[15] Y.-C. Wu, B.-H. Tseng, and C. E. Rasmussen, "Tam: Using trainable-action-mask to improve sample-efficiency in reinforcement learning for dialogue systems," in *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 2019.

[16] PJM Transmission Planning Department, "Exelon transmission planning criteria," 2009.

[17] W. Yu, J. Tan, Y. Bai, E. Coumans, and S. Ha, "Learning fast adaptation with meta strategy optimization," *IEEE Robotics and Automation Letters*, 2020.

[18] J. D. Williams, K. Asadi, and G. Zweig, "Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning," *arXiv preprint arXiv:1702.03274*, 2017.

[19] Q. Huang, R. Huang, B. J. Palmer, Y. Liu, S. Jin, R. Diao, Y. Chen, and Y. Zhang, "A generic modeling and development approach for WECC composite load model," *Electric Power Systems Research*, vol. 172, pp. 1–10, 2019.

[20] Q. Huang, R. Huang, and W. Hao, "An open-source platform for applying reinforcement learning for grid control," https://github.com/RLGC-Project/RLGC, accessed: 2020-12-10.