Distribution Network Reconfiguration Using Deep Reinforcement Learning

Mukesh Gautam, *Graduate Student Member, IEEE*, Mohammed Benidris, *Senior Member, IEEE*
Department of Electrical & Biomedical Engineering, University of Nevada, Reno
(emails: mukesh.gautam@nevada.unr.edu, mbenidris@unr.edu)

***Abstract*—This paper proposes a deep reinforcement learning (DRL)-based framework for distribution network reconfiguration (DNR). The objective of the proposed framework is to minimize power losses in the network and various reliability indices including System Average Interruption Frequency Index (SAIFI), System Average Interruption Duration Index (SAIDI), and Average Curtailed Power (ACP). Constraints of the optimization problem are radial topology constraint and all nodes traversing constraint. The distribution network is modeled as a graph and the optimal network configuration is determined by searching for an optimal spanning tree. Contrary to existing analytical and population-based approaches, where the entire analysis and computation is to be repeated to find the optimal network configuration for each system operating state, DRL-based DNR, if properly trained, can determine optimal or near-optimal configuration quickly even with changes in system states. The Q-learning, a model-free reinforcement learning algorithm, is used by the proposed DRL-based framework to learn the action-value function. The effectiveness and efficacy of the proposed framework for DNR is demonstrated through a case study performed on 33-node distribution test system.**

***Index Terms*—Deep Q Network, distribution system reliability, network reconfiguration, reinforcement learning, and spanning trees.**

I. Introduction

The goal of electric utilities is to provide efficient, reliable, and affordable electricity service to their customers through utilization of available resources. Since power interruptions are mostly due to failure of distribution system components [1], enhancing the reliability of distribution systems is inevitable to provide uninterruptible electric power supply to customers. Reliability of distribution systems can be enhanced in two ways: (a) optimal utilization of available resources using smart grid technologies and (b) installation of redundant resources. The option of installing redundant resources is not economical and a waste of resources. Therefore, sophisticated smart grid technologies should be developed to optimally utilize distribution system resources in an optimum manner. In this context, distribution network reconfiguration (DNR) is one of such smart grid technologies to provide efficient, economical, and reliable supply of electricity. DNR can optimize existing resources by modifying the configuration of distribution networks through changing status of sectionalizing and tie-switches.

Several analytical and population-based intelligent search approaches have been proposed in the literature to solve the DNR problem. A two-stage robust model has been proposed in [2] for DNR considering load uncertainty, where the first stage selects a network configuration and the second stage performs an AC optimal power flow for the given demand realization. In [3], a spanning tree-based genetic algorithm has been proposed for DNR with an objective of power loss minimization. A genetic algorithm-based DNR has been proposed in [4] for power quality and reliability improvement. Similarly, a genetic algorithm has been used for DNR to improve the reliability and optimal placement of distributed generators in [5]. In [6], a mixed-integer quadratic programming has been used for reliability constrained power loss minimization and a path-based model has been adopted for the distribution system. Also, a path-to-branch incidence matrix has been proposed in [6] to incorporate the reliability indices in the DNR problem. In [7], a neighborhood search algorithm has been used for DNR to improve the reliability and reduce losses by taking into account the uncertainties of data. An algorithm based on binary particle swarm optimization has been used in [8] to solve DNR problem considering maximization of the reliability and minimization of active power losses.

Analytical and population-based intelligent search methods used for DNR to improve reliability of distribution systems have the following shortfalls. Accuracy and effectiveness of analytical-based methods for DNR depend upon the accuracy of models used, where accurate models impose scalability challenges. Also, mathematical models are usually derived based on several approximations and they require complete system information. Population based methods, on the other hand, are computationally expensive due to the large search space, especially when system sizes increase.

Learning-driven approaches have been used to tackle the limitations of analytical and population-based approaches since learning-driven approaches can handle uncertainties by extracting knowledge from historical data. Moreover, learning-driven models are not required to be solved whenever new scenarios are encountered because of their ability to use their knowledge gained from historical data to solve for the new scenarios. Out of various learning-driven approaches, reinforcement learning (RL)-based approaches have the capabilities to learn from experiences during online operations [9], [10]. Learning-driven approaches are, therefore, gaining significant attention for optimal DNR.

In [11], batch-constrained RL has been used for the dynamic DNR with the objective of minimizing network operational costs. The RL algorithm proposed in [11] can learn the network reconfiguration control policy from a historical dataset without the use of an actual distribution network. In [12], network power loss and number of switching actions in the distribution network are minimized using a long-short term memory (LSTM) network. RL has also been used to simultaneously reduce network power loss and improve voltage profiles [13], where loop-based encoding is leveraged with NoisyNet deep Q learning to improve training effectiveness and computational efficiency.

In [14], a deep Q-Learning-based DNR has been proposed for reliability improvement by minimizing the average curtailed power. In [15], deep Q learning has been implemented to minimize line congestion and voltage violation problems while performing the DNR. The work presented in [15] has been tested for the computational cost and the scalability as compared to brute-force search algorithm and the genetic algorithm. Although there are several similarities between DNR for different objectives, optimum DNR for both power loss reduction and reliability improvement is a challenging task since it requires determining network power loss and reliability indices for each possible configuration. Therefore, developing intelligent learning-based approaches for DNR to decrease network power losses and enhance reliability is pivotal.

This paper proposes a deep reinforcement learning (DRL)-based framework for DNR to minimize network power loss and enhance the reliability of distribution systems. In the proposed optimization framework, the objective is to minimize system power loss and various reliability indices including System Average Interruption Frequency Index (SAIFI), System Average Interruption Duration Index (SAIDI), and Average Curtailed Power (ACP). In addition to nodal power balance constraints, all-node-traversing and radiality constraints are considered. In the training phase of the proposed algorithm, Q values are predicted using forward propagation of a deep neural network (DNN). Actions are selected using the Epsilon-Greedy algorithm. When actions are passed through the training environment, the DRL agent gets rewarded (or penalized) based on its performance. Target Q values are calculated based on the reward. The mean squared error (MSE), which is the most commonly employed loss function for regression, is computed using the predicted and target Q values. Errors are then back-propagated to update the weights of DNN. The trained DRL agent is then used to find the best network configuration. The proposed framework is validated through a case study on a 33-node distribution test system, and the results show that the proposed framework can effectively find a network configuration with high reliability level and low power loss.

The rest of the paper is organized as follows. Section II explains the mathematical formulation of the DNR problem with the loss minimization and reliability indices. Section III describes the proposed framework and solution approach. Section IV validates the proposed work through a case study on the 33-node system with several scenarios. Section V provides concluding remarks.

II. Mathematical Modeling

This section presents the mathematical formulation of the DNR problem and describes states, actions, and reward function in the context of DNR.

A. Problem Formulation

This subsection presents the objective functions and the constraints of the DNR problem under consideration.

1) Objective Functions: Reliability is one of the major factors that indicates performance of the system. Reliability of distribution systems can be quantified using several reliability indices. Out of various reliability indices, SAIFI, SAIDI, and Average Curtailed Power (ACP) are taken as reliability-related objective functions for the problem under consideration since they can capture the severity of the outages and are directly affected by the topology or configuration of a distribution network. The aforementioned objective functions, along with the network power loss are explained as follows.

(a) SAIFI: It is the average number of interruptions a customer would experience in a year. Mathematically, SAIFI can be expressed as follows.

$$SAIFI = \frac{\sum\limits_{k \in \Omega_k} N_k \times \lambda_k}{\sum\limits_{k \in \Omega_k} N_k}, \tag{1}$$

where N_k is the number of customers served by node k; Ω_k is the set of nodes with power demand; and λ_k is average annual failure rate at node k, which can be defined as follows.

$$\lambda_k = \sum_{l \in \Omega_{lk}} \lambda_l, \tag{2}$$

where λ_l is the failure rate of branch (or edge) l; and Ω_{lk} is the set of branches (or edges) between substation node and node k.

(b) SAIDI: It is the average duration of interruptions a customer would experience in a year. Mathematically, SAIDI can be expressed as follows.

$$SAIDI = \frac{\sum\limits_{k \in \Omega_k} N_k \times U_k}{\sum\limits_{k \in \Omega_k} N_k}, \tag{3}$$

where U_k is the average annual power unavailability duration at node k, which can be defined as follows.

$$U_k = \sum_{l \in \Omega_{lk}} \lambda_l r_l, \tag{4}$$

where λ_l is the failure rate of branch (or edge) l; and r_l is the outage duration (or repair time) of branch (or edge) l.

(c) Average Curtailed Power (ACP): It is the average power curtailment of a distribution system in a year. Mathematically, ACP can be expressed as follows.

$$ACP = \sum_{k \in \Omega_k} P_{d,k} U_k, \tag{5}$$

where $P_{d,k}$ is the power demand at node k.

(d) Power Loss: The total power loss of the system is calculated by adding power losses of all branches of a particular configuration of the distribution network.

$$P_{loss} = \sum_{k \in \Omega_B} P_{loss,k}, \tag{6}$$

where Ω_B is the set of branches (or lines) of the particular configuration; and $P_{loss,k}$ is the power loss of the kth branch.

2) Constraints: The DNR problem under consideration is subjected to various constraints including nodal power balance constraint, radiality constraint, and all-node-traversing constraint.

(a) Node power balance constraint: The power balance constraint at each node of the system can be expressed as follows.

$$\sum_{j \in \Omega_g(j)} P_{g,j} + \sum_{l \in \Omega_L(j)} P_{l,j} = P_{D,j} \tag{7}$$

where $\Omega_g(j)$ is the set of sources connected to node j; $\Omega_L(j)$ is the set of lines connected to node j; $P_{g,j}$ is the power injected from source j; $P_{D,j}$ is the load at node j; and $P_{l,j}$ is the line power flow from node l to node j.

(b) Radiality constraint: Radiality constraint is always maintained in a distribution system in order to design the protection coordination schemes. Each candidate configuration should have a radial topology since most of the practical distribution systems do not have loop structure.

(c) All-node-traversing constraint: A distribution system operator should always configure the network in such a way that all loads are supplied with power in non-contingent scenarios. Therefore, for each candidate configuration, all-node-traversing constraint should always be satisfied.

Constraints (b) and (c) are satisfied if we search for a spanning tree. Consider a distribution network represented by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is a set of nodes (or vertices) and $\mathcal{E}$ is a set of edges (or branches). For the graph, a node-branch incidence matrix can be constructed after satisfying all-node-traversing constraint. If $n = |\mathcal{N}|$ denotes the number of nodes and $e = |\mathcal{E}|$ denotes the number of edges of a particular network configuration, then the node-branch incidence matrix $A \in \mathbb{R}^{n \times e}$ is the matrix with element a_{ij} calculated as follows.

$$a_{ij} = \begin{cases} +1 & \text{if branch } j \text{ starts at node } i \\ -1 & \text{if branch } j \text{ ends at node } i \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

If the node-branch incidence matrix A is full ranked, then the radiality constraint is satisfied.

B. States, Actions, and Reward Function

The choice of states, actions, and reward function can play a crucial role for the proper training of a reinforcement learning (RL) agent. States, actions, and reward function must be, therefore, chosen with careful consideration. For the DNR problem under consideration, a vector of power demand of all nodes is taken as the state. The action is a vector of opened edges. The cost function at time step t is the sum of SAIFI, SAIDI, ACP, and the system power loss, expressed as follows.

$$C_t = SAIFI_t + SAIDI_t + ACP_t + P_{loss,t} \tag{9}$$

The total reward at time step t is computed as follows.

$$R_t = \begin{cases} -C_t & \text{if all constraints are satisfied} \\ -\rho & \text{if any constraint is violated} \end{cases} \tag{10}$$

where ρ denotes the penalty factor.

III. Proposed Framework

This work leverages recently advanced reinforcement learning techniques for DNR to improve reliability of distribution systems. This section provides a brief overview of Deep Q learning, reward function, and training attributes of the Deep Q learning.

A. Deep Q Learning

A reinforcement learning (RL) is a branch of machine learning in which an *agent* learns to take suitable *actions* to maximize cumulative *reward* it gets from an uncertain *environment*. In general, an RL system consists of four main integrants: policy, reward, value functions, and environment model. An agent decides the action to be taken based on the policy. The policy maps states to actions. When the agent takes an action, it gets rewarded (or penalized). Value function calculates the expected value of cumulative reward that an agent gets when it follows a certain policy. There are different algorithms for RL. The choice of an algorithm depends on many factors such as the continuous/discrete nature of states, continuous/discrete action-space, etc. For the DNR problem under consideration, the action-space is discrete in nature, which makes Q-Learning a suitable candidate for the problem. However, a basic Q-Learning needs large sized look-up tables where state-action values are stored. To avoid the use of large sized look-up tables, a deep neural network (DNN) is used as an action-value function approximator. The addition of DNN in the basic Q-Learning makes the framework a Deep Q Network (DQN). The update rule for action-value function in Q-learning is defined as follows [9].

$$\begin{aligned} Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha \times [R_{t+1} \\ + \gamma \times \max_a Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)] \end{aligned} \tag{11}$$

where A_t and S_t are the action and state of an agent at t^{th} iteration; $Q(S_t, A_t)$ is the action-value function at t^{th} iteration; $Q(S_{t+1}, A_{t+1})$ is the action-value function at $(t{+}1)^{\text{th}}$ iteration; α is the learning rate; and γ is the reward discount factor.

Instead of iteratively updating the action-value function, the DNN is trained and the parameters of the action-value function are optimized to minimize the mean-squared error (MSE) loss function (i.e., regression loss function), which is expressed as follows [16].

$$L(\theta) = \mathbb{E}[(Q(S_t, A_t|\theta) - y_t)^2] \tag{12}$$

TABLE I
HYPER-PARAMETER SETTINGS OF THE PROPOSED DRL FRAMEWORK

Hyper-parameters	Values
Number of hidden layers	2
No. of neurons in hidden layers	10, 10
Learning rate	0.001
Activation function of output layer	Linear
Activation function of hidden layers	ReLU
Optimizer	Adam

where $\mathbb{E}$ denotes expectation operator; θ denotes the parameter of action-value function $Q(S_t, A_t)$; and y_t denotes the target action-value function, which is defined as follows.

$$y_t = R(S_t, A_t) - \gamma \times \max_a Q(S_{t+1}, A_{t+1}|\theta') \quad (13)$$

In (13), $R(S_t, A_t)$ denotes reward function at t^{th} iteration and θ' denotes the parameter of action-value function $Q(S_{t+1}, A_{t+1})$.

B. Training Attributes

The training of DQN is performed for a certain number of episodes (n_{ep}). The initial state of the system is the state with a certain value of load active power at each node. The weights of DNN are initialized with some random values. In each episode, the predicted Q values corresponding to each edge of the system is computed based on forward propagation of DNN. For the selection of actions, the Epsilon-Greedy (exploration-exploitation) algorithm [17] is used. The value of exploration rate i.e., epsilon (ε) is initialized at ε_{max}. In order to allow the DQN to explore during initial episodes, ε_{max} is set equal to 1 and as the episode progresses, the exploration rate is decreased progressively and the exploitation rate is increased. Until 80% of the total episodes, the epsilon is updated after each episode based on (14).

$$\varepsilon_{new} = \varepsilon_{old} - \frac{\varepsilon_{max} - \varepsilon_{min}}{0.8 \times n_{ep}}, \quad (14)$$

where ε_{min} is the minimum exploration rate. Since the DQN sufficiently explores till 80% of the total episodes, the exploration rate is set to a very low value (e.g., 0.005) for the last 20% of the episodes. The target Q value of the DQN is computed using (13). The MSE losses for each episode are computed based on (12) using the actual and target Q-values. These MSE losses are back-propagated to update the weights of the DNN.

Algorithm 1 provides the procedure of training the proposed DRL-based DNR.

Algorithm 1: Training of the proposed DRL-based DNR.

```
Input  : System data including line data, load data,
         etc.
Initialize parameters $\theta$ of the DQN with random values
for $episode \leftarrow 1$ to $n_{ep}$ do
    Initialize the system with initial state (here, load
      data at each node)
    for $t \leftarrow 1$ to $T$ do
        Generate action-value function Q based on
          current state
        Generate action using Epsilon-Greedy
          Algorithm
        Calculate the reward function $R_{t+1}(S_t, A_t)$
        Calculate DQN Loss Function based on the
          action-value function Q and the target
          action-value function
        Perform back-propagation to update parameters
          $\theta$ of the DQN
Output : Indices of opened edges
```

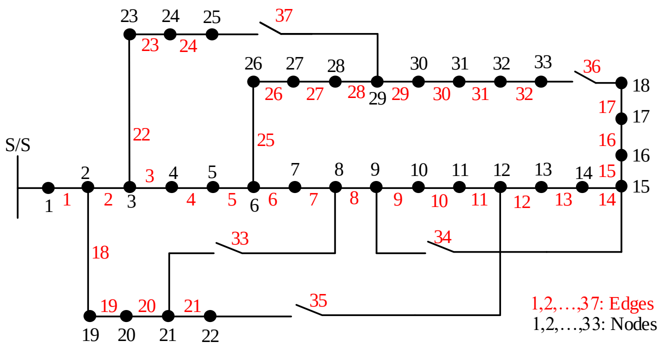

Fig. 1. 33-node distribution test system

IV. CASE STUDY AND DISCUSSION

A. Case Study Parameters

The proposed framework is implemented on the 33-node distribution test system. The case study parameters of the system are as follows. The failure rate of each branch is assumed proportional to its branch impedance. The highest failure rate (in this case 0.4 failures/year or f/yr) is assigned to the branch with the largest impedance; the lowest failure rate (in this case 0.1 f/yr) is assigned to the branch with the smallest impedance; and linear interpolation is used to determine the failure rates of remaining branches. Regarding the outage duration (or repair rate) of each branch, its value is assumed to be constant (6 hr is used). Normally-open switches (or tie switches) are assumed to be fully reliable. The failure rate (λ_k) and annual power unavailability duration (U_k) of each node are determined based on the values of failure and repair rates for each branch, respectively, using (2) and (4). The customer data for the system under study have been taken from [18].

The 33-node distribution test system is 100 kVA, 12.66 kV radial distribution system with 33 nodes, 32 branches and 5 tie-lines. Therefore, the total number of branches in this system is 37. All branches (including tie-lines) are numbered from 1 to 37 as shown in Fig. 1. The total load of the system is 3.71 MW. The detailed data of the system is provided in [19].

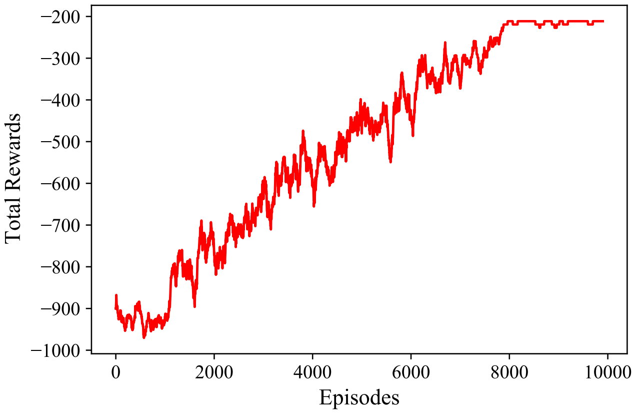

Fig. 2. Total Rewards of training episodes

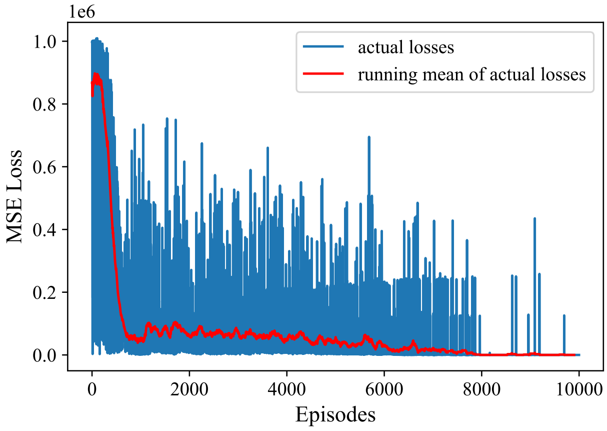

Fig. 3. MSE of losses of training episodes

B. Training

The training of the DQN for the 33-node system is performed for 10,000 episodes. The system state is initialized with a vector of load active power of all nodes. Initially, the exploration rate is set very high (i.e., 1) to allow sufficient exploration and the exploration rate is set to a very low value (0.005) after reaching 80% of the total episodes. Due to this reason, average reward goes on increasing (and MSE loss goes on decreasing) till 80% of the total episodes, but average reward and MSE loss saturate after reaching 80% of the total episodes. Fig. 2 shows the actual rewards and running mean (100-episode window) of actual rewards as the episode progresses. It can be seen from Fig. 2 that as the number of episodes increases, the running mean of the reward increases and saturates after 8,000 episodes. Similarly, Fig. 3 shows the actual values and running mean (100-episode window) of MSE losses as the episode progresses. Fig. 3 also shows that initially MSE loss is very high, goes on decreasing after a few episodes, and saturates after 8,000 episodes.

TABLE II
RESULTS FOR THE BASE CASE SCENARIO

Base configuration with open edges	33, 34, 35, 36, 37
Network power loss	203.10 kW
SAIFI	1.3788 interruption/customer/yr
SAIDI	8.2730 hr/customer/yr
Average Curtailed Power	29.43 MWh/yr

TABLE III
FINAL RESULTS OBTAINED AFTER TRAINING OF THE PROPOSED FRAMEWORK

Final configuration with open edges	3, 10, 15, 26, 35
Network power loss	185.53 kW
SAIFI	1.1098 interruption/customer/yr
SAIDI	6.6587 hr/customer/yr
Average Curtailed Power	23.59 MWh/yr

C. Comparison

The final results obtained after training of the proposed algorithm are compared with the base case, where all tie-switches are opened. When all tie-switches (switches 33, 34, 35, 36, and 37) are opened, the network power loss is 203.10 kW, SAIFI is 1.3788 interruption/customer/yr, SAIDI is 8.2730 hr/customer/yr, and average curtailed power is 29.44 MWh/yr. These results are shown in Table II.

The final configuration obtained after training is the configuration with open edges 3, 10, 15, 26, and 35. For this configuration, the network power loss is 185.53 kW, SAIFI is 1.1098 interruption/customer/yr, SAIDI is 6.6587 hr/customer/yr, and average curtailed power is 23.59 MWh/yr. These results are shown in Table III. These results show that the proposed approach can improve the reliability level and reduce power loss in distribution systems.

V. Conclusion

This paper has proposed a DRL-based framework to determine the configuration of a distribution network with optimal or near optimal values of network power loss, and various reliability indices including SAIFI, SAIDI, and average curtailed power. During the training of the proposed algorithm, the weights of DNN were initialized with random values and the system state was initialized with a vector of load active power at each node. The target Q values were computed based on the reward the DRL agent gets from the environment. The predicted and target Q values were used to update the weights of DNN. Case study was performed on the 33-node distribution test system. The results exhibit the effectiveness of the proposed framework to improve the reliability level and reduce power loss in distribution systems.

Acknowledgement

This work was supported by the U.S. National Science Foundation (NSF) under Grant NSF 1847578.

References

[1] A. A. Chowdhury and D. Koval, *Power Distribution System Reliability: Practical Methods and Applications*. J. Wiley Inc., 2009.

[2] C. Lee, C. Liu, S. Mehrotra, and Z. Bie, "Robust distribution network reconfiguration," *IEEE Trans Smart Grid*, vol. 6, no. 2, pp. 836–842, 2014.

[3] M. Gautam, N. Bhusal, M. Benidris, and S. J. Louis, "A spanning tree-based genetic algorithm for distribution network reconfiguration," in *2020 IEEE Industry Applications Society Annual Meeting*. IEEE, 2020, pp. 1–6.

[4] N. Gupta, A. Swarnkar, and K. Niazi, "Distribution network reconfiguration for power quality and reliability improvement using genetic algorithms," *International Journal of Electrical Power & Energy Systems*, vol. 54, pp. 664–671, 2014.

[5] Y. Tian, M. Benidris, S. Sulaeman, S. Elsaiah, and J. Mitra, "Optimal feeder reconfiguration and distributed generation placement for reliability improvement," in *2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, 2016, pp. 1–7.

[6] J. Jose and A. Kowli, "Reliability constrained distribution feeder reconfiguration for power loss minimization," in *2016 National Power Systems Conference (NPSC)*, 2016, pp. 1–6.

[7] P. Zhang, W. Li, and S. Wang, "Reliability-oriented distribution network reconfiguration considering uncertainties of data by interval analysis," *International Journal of Electrical Power & Energy Systems*, vol. 34, no. 1, pp. 138–144, 2012.

[8] B. Amanulla, S. Chakrabarti, and S. Singh, "Reconfiguration of power distribution systems considering reliability and power loss," *IEEE transactions on power delivery*, vol. 27, no. 2, pp. 918–926, 2012.

[9] A. Zai and B. Brown, *Deep reinforcement learning in action*. Manning Publications, 2020.

[10] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[11] Y. Gao, W. Wang, J. Shi, and N. Yu, "Batch-constrained reinforcement learning for dynamic distribution network reconfiguration," *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 5357–5369, 2020.

[12] X. Ji, Z. Yin, Y. Zhang, B. Xu *et al.*, "Real-time autonomous dynamic reconfiguration based on deep learning algorithm for distribution network," *Electric Power Systems Research*, vol. 195, p. 107132, 2021.

[13] B. Wang, H. Zhu, H. Xu, Y. Bao, and H. Di, "Distribution network reconfiguration based on noisynet deep Q-learning network," *IEEE Access*, vol. 9, pp. 90 358–90 365, 2021.

[14] M. Gautam, N. Bhusal, and M. Benidris, "Deep Q-learning-based distribution network reconfiguration for reliability improvement," in *2020 IEEE/PES Transmission and Distribution Conference and Exposition (T&D)*. IEEE, 2022, pp. 1–5.

[15] S. H. Oh, Y. T. Yoon, and S. W. Kim, "Online reconfiguration scheme of self-sufficient distribution network based on a reinforcement learning approach," *Applied Energy*, vol. 280, p. 115900, 2020.

[16] D. Cao, W. Hu, J. Zhao, G. Zhang, B. Zhang, Z. Liu, Z. Chen, and F. Blaabjerg, "Reinforcement learning and its applications in modern power and energy systems: A review," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1029–1042, 2020.

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

[18] A. Kavousi-Fard and T. Niknam, "Multi-objective stochastic distribution feeder reconfiguration from the reliability point of view," *Energy*, vol. 64, pp. 342–354, 2014.

[19] M. E. Baran and F. F. Wu, "Network reconfiguration in distribution systems for loss reduction and load balancing," *IEEE Power Engineering Review*, vol. 9, no. 4, pp. 101–102, 1989.