



# Restoring Distribution System Under Renewable Uncertainty Using Reinforcement Learning

## Preprint

Xiangyu Zhang, Abinet Tesfaye Eseye, Bernard Knueven, and Wesley Jones

*National Renewable Energy Laboratory*

*Presented at the IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (IEEE SmartGridComm) November 11-13, 2020*

**NREL is a national laboratory of the U.S. Department of Energy  
Office of Energy Efficiency & Renewable Energy  
Operated by the Alliance for Sustainable Energy, LLC**

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at [www.nrel.gov/publications](http://www.nrel.gov/publications).

Contract No. DE-AC36-08GO28308

**Conference Paper**  
NREL/CP-2C00-77116  
October 2020



# Restoring Distribution System Under Renewable Uncertainty Using Reinforcement Learning

## Preprint

Xiangyu Zhang, Abinet Tesfaye Eseye, Bernard Knueven, and Wesley Jones

*National Renewable Energy Laboratory*

### Suggested Citation

Zhang, Xiangyu, Abinet Tesfaye Eseye, Bernard Knueven, and Wesley Jones. 2020. *Restoring Distribution System Under Renewable Uncertainty Using Reinforcement Learning: Preprint*. Golden, CO: National Renewable Energy Laboratory. NREL/CP-2C00-77116. <https://www.nrel.gov/docs/fy21osti/77116.pdf>.

© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

**NREL is a national laboratory of the U.S. Department of Energy  
Office of Energy Efficiency & Renewable Energy  
Operated by the Alliance for Sustainable Energy, LLC**

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at [www.nrel.gov/publications](http://www.nrel.gov/publications).

Contract No. DE-AC36-08GO28308

**Conference Paper**  
NREL/CP-2C00-77116  
October 2020

National Renewable Energy Laboratory  
15013 Denver West Parkway  
Golden, CO 80401  
303-275-3000 • [www.nrel.gov](http://www.nrel.gov)

## NOTICE

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Office of Electricity Delivery and Energy Reliability. The views expressed herein do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at [www.nrel.gov/publications](http://www.nrel.gov/publications).

U.S. Department of Energy (DOE) reports produced after 1991 and a growing number of pre-1991 documents are available free via [www.OSTI.gov](http://www.OSTI.gov).

*Cover Photos by Dennis Schroeder: (clockwise, left to right) NREL 51934, NREL 45897, NREL 42160, NREL 45891, NREL 48097, NREL 46526.*

NREL prints on paper that contains recycled content.

# Restoring Distribution System Under Renewable Uncertainty Using Reinforcement Learning

Xiangyu Zhang, Abinet Tesfaye Eseye, Bernard Knueven, Wesley Jones  
Computational Science Center  
National Renewable Energy Laboratory (NREL)  
Golden, U.S.A

**Abstract**—Distributed energy resources (DERs) in distribution systems, including renewable generation, micro-turbine, and energy storage, can be used to restore critical loads following extreme events to increase grid resiliency. However, properly coordinating multiple DERs in the system for multi-step restoration process under renewable uncertainty and fuel availability is a complicated sequential optimal control problem. Due to its capability to handle system non-linearity and uncertainty, reinforcement learning (RL) stands out as a potentially powerful candidate in solving complex sequential control problems. Moreover, the offline training of RL provides excellent action readiness during online operation, making it suitable to problems such as load restoration, where in-time, correct and coordinated actions are needed. In this study, a distribution system prioritized load restoration based on a simplified single-bus system is studied: with imperfect renewable generation forecast, the performance of an RL controller is compared with that of a deterministic model predictive control (MPC). Our experiment results show that the RL controller is able to learn from experience, adapt to the imperfect forecast information and provide a more reliable restoration process when compared with the baseline controller.

## I. INTRODUCTION

Resilience of modern power systems means its capability to withstand extreme events (e.g., hurricane, earthquake or deliberate attack) and rapidly restore service for critical customers under impact. In the U.S., blackouts triggered by Hurricane Maria [1] (Puerto Rico, 2017), Superstorm Sandy [2] (Northeast, 2012) and many other extreme events are ringing the alarm for improving critical infrastructure’s resilience. Load restoration is one practice for improving grid resilience; it recovers load as much as possible during an outage event. Traditionally, an outage area is re-energized by connecting to an alternative substation [3]. However, under the impact of a massive extreme event, a neighboring substation either might

This work was authored by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the DOE Office of Electricity (OE) Advanced Grid Modeling program. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

This research was performed using computational resources sponsored by the Department of Energy’s Office of Energy Efficiency and Renewable Energy and located at the National Renewable Energy Laboratory.

978-1-7281-6127-3/20/\$31.00 ©2020 IEEE

not be able to provide enough power or, more likely, is out of service as well. As a result, utilizing distributed energy resources (DERs), either dispatchable or non-dispatchable renewable generations, within the distribution system to restore the electricity service becomes a potential solution.

One issue with load restoration using DERs is to identify a proper method to handle uncertainty from renewable generation for optimal control. According to [4], three mainstream approaches has been considered in literature: 1) relying on the forecast of renewable generation [5]; 2) conducting a scenario-based stochastic optimization [6]; and 3) using the robust optimization approach [7]. Apparently, using Approach 1, the controller is essentially still deterministic and its performance mainly depends on the forecast accuracy. A controller based on Approach 2, however, suffers from a heavy computation burden in order to consider a comprehensive set of random scenarios. The robust optimization based controller, though based on computation less expensive than Approach 2, is prone to be sub-optimal due to the conservative behavior of robust optimization techniques. To address these limitations, Wang *et al.* [4] propose a risk-limiting approach to restore loads in a distribution system by solving a chance-constrained optimization problem. To sum up, the state-of-the-art methods for load restoration are all optimization based, either deterministic or stochastic, in order to maximize load being restored.

In recent years, reinforcement learning (RL) and deep learning, together with high performance computing (HPC), have become a great combination for solving sequential optimal decision-making problems. Success stories from computer science [8], robotics [9] and energy systems [10] adequately demonstrate RL’s capability. Compared with optimization-based algorithms, RL can better handle/more easily model the nonlinearity and stochasticity in the controlled system. An additional important advantage of RL over an optimization-based controller is its action readiness: an RL optimal control policy can be trained offline (before extreme events) and loaded onto the optimal controller ahead of time. When control process is initiated, at each step, the control action can be generated instantly according to the learned policy instead of optimizing on-the-fly during the near real-time control. Due to these merits, a RL-based controller (RLC) can be a potentially powerful candidate for solving the load restoration problem under uncertainty. Therefore, in this paper, we will investigate the performance of an RLC and compare it with a deterministic

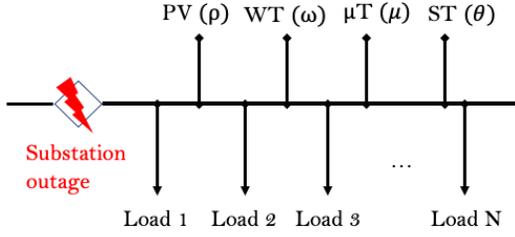


Fig. 1. Illustration of a single bus system.

model predictive controller (MPC). The results of this study showcase the effectiveness of RLC and provide some insights on future RLC design in power system domain.

In the rest of the paper, Section II presents the load restoration optimal control problem mathematically and the associated MPC method; Section III explains the prediction methods we used to predict generation from uncertain resources; Section IV shows the formulation of the proposed RLC; Section V evaluates the control performance of these two controllers and showcases the learning efficiency of the proposed RLC; and finally conclusion and future works are discussed in Section VI.

## II. A PRIORITIZED LOAD RESTORATION PROBLEM

### A. Problem Formulation and Assumptions

In this paper, we consider restoring a distribution network after a substation outage by using DER assets within the system; the objective is to maximize the prioritized load restoration to improve system resiliency. In the distribution system of interest, there are four DER assets (Two renewable DERs  $\mathcal{R} = \{\rho, \omega\}$  and two dispatchable DERs  $\mathcal{G} = \{\mu, \theta\}$ ) to be leveraged: a photovoltaic (PV) generator ( $\rho$ ), a wind generator ( $\omega$ ), a micro-turbine ( $\mu$ ), and an energy storage system ( $\theta$ ). Critical load  $i \in \mathcal{L}$  is prioritized by an importance factor  $\eta^i$  and  $\mathbf{H} = [\eta^1, \eta^2, \dots, \eta^N]^T \in \mathbb{R}^N$  is the vector form for all loads ( $N$  is the number of critical loads). The system configuration is illustrated in Fig. 1.

Several assumptions are made in this study as follows:

- 1) Fuel for the micro-turbine and the initial storage for battery system are limited, and these two DERs alone are not sufficient to restore the system.
- 2) Renewable generation from the PV and wind turbine can be predicted but the forecast is imperfect to reflect a realistic setting.
- 3) The demand from critical loads ( $\mathbf{P} = [p^1, p^2, \dots, p^N]^T \in \mathbb{R}^N$ ) is assumed to be time-invariant over the control horizon, and it can be partially restored at any step.
- 4) The network/power flow constraints are not considered. This work focuses on investigating the performance of RL controllers under uncertainty from the energy adequacy perspective.
- 5) The length of restoration control horizon/upstream repair time (i.e.,  $\mathcal{T}$ ) is deterministic and known in advance.

At each step  $t \in \mathcal{T}$ , the generation of micro-turbine ( $p_t^\mu$ ) and storage ( $p_t^\theta$ ), demand restored for each load ( $\mathbf{P}_t = [p_t^1, p_t^2, \dots, p_t^N]^T \in \mathbb{R}^N$ ) and renewable curtailment ( $p_t^\alpha$ ) are dynamically determined. Due to the uncertainty of renewable generation and the limit on available fuel and initial energy storage, strong temporal dependency exists over the control horizon, which makes a RL controller as a potential candidate for this problem.

### B. Mathematical Formulation

Before delving into an RL controller, in this section, the mathematical formulation of the above-mentioned sequential optimal control problem for prioritized load restoration is presented for better understanding.

As shown in (1), the objective function for the optimal control consists of three parts:

- 1) To improve system resiliency: maximize the load restoration over time (by priority ranking);
- 2) To provide a reliable and monotonic load restoration: penalize frequent/repeated load restoration and shedding due to the intermittent renewable generation;
- 3) Penalize unnecessary renewable curtailment.

$$\mathcal{C} = \sum_{t \in \mathcal{T}} \mathbf{H}^T \mathbf{P}_t - \epsilon \sum_{t \in \mathcal{T}} \mathbf{H}^T [\mathbf{P}_{t-1} - \mathbf{P}_t]^+ - \beta \mathbf{1}^T \mathbf{P}^\alpha \quad (1)$$

In (1),  $[[x^1, x^2, \dots, x^N]^+] = [(x^1)^+, (x^2)^+, \dots, (x^N)^+]$ , where  $(x^i)^+ = \max(0, x^i)$ .  $\mathbf{P}^\alpha$  is a vector representing renewable curtailment over  $\mathcal{T}$ . At  $t = 0$ , assume all loads are not served (i.e.,  $\mathbf{P}_0 = \mathbf{0}$ ). Parameters  $\epsilon$  and  $\beta$  are unitless penalties for shedding restored load and curtailing renewable, respectively. In general, the controller needs to be confident to sustain a load for at least  $\epsilon$  steps before it is restored; otherwise, overall restoring this load will be penalized.

While maximizing  $\mathcal{C}$ , the following constraints should be satisfied for all  $t \in \mathcal{T}$ , among which (2) represents generation-load power balance, (3) represents the feasible range of load restoration, (4) and (5) represent feasible output power from micro-turbine and the constraint on fuel availability (i.e.,  $E^\mu$ .  $\tau$  in (5) is the control interval), and (6) - (9) represent constraints on storage output/state of charge feasibility, charging/discharging state transition and initial storage. Ramping rate constraints for both dispatchable DERs are not considered.

$$\sum_{g \in \mathcal{G}} p_t^g + \sum_{r \in \mathcal{R}} \hat{p}_t^r - p_t^\alpha = \mathbf{1}^T \mathbf{P}_t \quad (2)$$

$$\mathbf{0} \leq \mathbf{P}_t \leq \mathbf{P} \quad (3)$$

$$\underline{p}^\mu \leq p_t^\mu \leq \overline{p}^\mu \quad (4)$$

$$\sum_{t \in \mathcal{T}} p_t^\mu \cdot \tau \leq E^\mu \quad (5)$$

$$-p_t^{\theta, ch} \leq p_t^\theta \leq p_t^{\theta, dis} \quad (6)$$

$$S_{t+1}^\theta = S_t^\theta - \zeta_t \cdot p_t^\theta \cdot \tau \quad (7)$$

$$\underline{S}^\theta \leq S_t^\theta \leq \overline{S}^\theta \quad (8)$$

$$S_0^\theta = s_0 \quad (9)$$

In (2), renewable generation forecast are given as exogenous inputs (i.e.,  $\hat{p}_t^\rho$  and  $\hat{p}_t^\omega$  are given for all  $t \in \mathcal{T}$ ). Though renewable generation forecast are used for decision-making, the simulator proceeds with actual generation  $p_t^\rho$  and  $p_t^\omega$ . In (7),  $\zeta_t$  is the energy storage efficiency and  $\zeta_t = 1/\zeta^{dis}$  when battery is discharging (i.e.,  $p_t^\theta > 0$ ) and  $\zeta_t = \zeta^{ch}$  when it is charging (i.e.,  $p_t^\theta < 0$ ). Because (7) is conditional, it makes the problem a mixed integer linear programming (MILP) problem.

Combine all above together, the sequential optimal control for restoring the distribution system is presented as below:

$$\begin{aligned} & \text{maximize} && (1) \\ & \mathbf{P}_t, p_t^\mu, p_t^\theta, p_t^\alpha (t \in \mathcal{T}) && (10) \\ & \text{subject to} && (2) - (9) \quad (\forall t \in \mathcal{T}) \end{aligned}$$

### C. Model Predictive Control

In this study, a model predictive controller (MPC) is used as a baseline controller. Due to the uncertainty from renewable generation, the MPC solves Problem (10) repeatedly with updated renewable forecast (i.e.,  $\hat{p}_t^\rho$  and  $\hat{p}_t^\omega$ ) and system state. We assume the MPC re-optimizes at every time interval in the control horizon, and over the control horizon, the optimization problem control horizon shrinks at each control interval by one till the presumed system recovery time. At each control step, with the optimal control problem solved on-the-fly, the immediate step decisions are applied, and the rest are discarded and the MPC control horizon shifts by one step forward in time.

## III. RENEWABLE FORECASTS

Two approaches for wind and PV generation forecast used in this paper are presented. It is worth mentioning that these two forecast approaches are used for the simplicity for demonstration and they do not represent the state-of-the-art high accuracy prediction methods, which is not the focus of this study.

### A. Wind Power Forecast

A short-term recursive multi-step time series forecasting technique [11] is leveraged for wind generation prediction. Specifically, a supervised learning model  $M$  is trained using the past eight days of data and the wind forecast  $\hat{\mathbf{P}}_t^\omega = [p_t^\omega, \hat{p}_{t+1}^\omega, \hat{p}_{t+2}^\omega, \dots, \hat{p}_{t+k}^\omega]$  can be made as shown below ( $k$  is the length of prediction period and  $l$  is the number of prior steps of the wind generation, used as predict features).

$$\begin{aligned} \hat{p}_{t+1}^\omega &= M(p_t^\omega, p_{t-1}^\omega, \dots, p_{t-l+1}^\omega) \\ \hat{p}_{t+2}^\omega &= M(\hat{p}_{t+1}^\omega, p_t^\omega, \dots, p_{t-l+2}^\omega) \\ &\dots \\ \hat{p}_{t+k}^\omega &= M(\hat{p}_{t+k-1}^\omega, \hat{p}_{t+k-2}^\omega, \dots, \hat{p}_{t+k-l}^\omega) \end{aligned} \quad (11)$$

### B. PV Power Forecast

The PV power forecast is based on a simple retrospective approach: namely output values from the same time the day before is used to model the generation of the next day with an adjustment, e.g., the predicted PV output for 11 a.m. today is the PV output was at 11 a.m. yesterday plus a calculated

adjustment, which corresponding to the daily weather changes. Specifically, at prediction time  $t$ , the prediction error is estimated based on the previous hours' actual realization. The retrospective forecast for the next three hours is adjusted (with a receding multiplier) to account for weather variability between yesterday's realization to today's realization so far. This method is straightforward to implement for the purposes of demonstration, and avoids some complications when creating look-ahead PV forecasts [12].

## IV. THE REINFORCEMENT LEARNING APPROACH

RL basics are not discussed here, interested readers should refer to [13] for more detailed preliminaries. Overall, training an RL agent is to learn from experience a mapping relationship (i.e., control policy)  $\pi(\mathbf{a}_t | \mathbf{S}_t)$  that determines an optimal action  $\mathbf{a}_t$  at state  $\mathbf{S}_t$  which will maximize expected cumulative future reward,  $\mathbb{E}(\sum_{t \in \mathcal{T}} r_t)$ . In deep reinforcement learning, the control policy is implemented using a neural network. Changing the parameters of the neural network (i.e.,  $\psi$ ), will result in a different policy  $\pi_\psi(\mathbf{a}_t | \mathbf{S}_t)$ . A class of RL algorithms, called policy gradient, is to use gradient ascent to update  $\psi$  so that  $\mathbb{E}(\sum_{t \in \mathcal{T}} r_t)$  is maximized and the optimal control policy  $\pi^*(\mathbf{a}_t | \mathbf{S}_t)$  is obtained until  $\psi$  is converged.

### A. Markov Decision Process Formulation

Typically, an optimization problem is formulated as a Markov Decision Process (MDP) to be solved using RL. Below, three most important elements of an MDP are defined corresponding to the problem formulation in (10).

**Action** is the set of decision variables the RL controller needs to determine at  $t \in \mathcal{T}$ . In this study, action is defined as  $\mathbf{a}_t = [p_t^\mu, p_t^\theta, p_t^\alpha]$ . With exogenous inputs (i.e.,  $p_t^\rho$  and  $p_t^\omega$ ) and  $\mathbf{a}_t$ , the total restored load  $\mathbf{1}^T \mathbf{P}_t$  can be determined by (2). In this single bus scenario, serving load with higher priority is dominantly optimal than serving lower priority load. So restored amount for each load  $p_t^i$  is determined once the total load restoration  $\mathbf{1}^T \mathbf{P}_t$  is determined, according to  $\mathbf{H}$ .

**State** represents the system status of the current step. In this study, state is defined as  $\mathbf{S}_t = [\hat{\mathbf{P}}_t^\rho, \hat{\mathbf{P}}_t^\omega, S_t^\theta, \hat{E}_t^\mu, \mathbf{1}^T \mathbf{P}_t / \mathbf{1}^T \mathbf{P}, t]$ .  $\hat{\mathbf{P}}_t^\rho$  and  $\hat{\mathbf{P}}_t^\omega$  are the PV and wind generation imperfect forecast for the next hour.  $S_t^\theta$  and  $\hat{E}_t^\mu$  are the state of charge for the storage and remaining fuel for the micro-turbine, representing power supporting capability of the controllable DERs at current step.  $\mathbf{1}^T \mathbf{P}_t / \mathbf{1}^T \mathbf{P}$  represents current load restoration level.  $t$  represents the current step index.

**Reward:** Given  $\mathbf{S}_t$  and the RL agent's decision  $\mathbf{a}_t$ , the environment returns a reward at each step representing how good the action is. The reward is defined as  $r_t = \mathbf{H}^T \mathbf{P}_t - \mathbf{H}^T [\mathbf{P}_{t-1} - \mathbf{P}_t]^+ - \beta p_t^\alpha$ , which is the same as one step value in the objective function in (1).

Based on this MDP formulation, an OpenAI Gym [14] environment is developed to enable the reinforcement learning.

### B. Evolution Strategies based RL (ES-RL)

Essentially, the process of on-policy reinforcement learning can be divided into two tasks: 1) experience collection and

TABLE I  
PARAMETERS USED FOR CASE STUDY

Var	Value	Var	Value
$\mathbf{H}$	[1.0, 1.0, 0.9, 0.85, 0.8, 0.65, 0.45, 0.4, 0.3, 0.3]	$[p^\mu, \bar{p}^\mu]$	[0, 300]
$\mathbf{P}$	[33, 34, 8.5, 85, 60, 60, 58, 115, 64, 85]	$E^\mu$	1000
$\mathcal{T}$	[1, 2, ..., 72]	$(p^{\theta,dis}, p^{\theta,ch})$	[200, 200]
$\mathcal{L}$	[1, 2, ..., 10]	$s_0$	720
PV	[0, 300]	$(S^\theta, \hat{S}^\theta)$	[160, 800]
Wind	[0, 150]	$\tau$	1/12
$\epsilon$	100	$\beta$	0.2
$\zeta^{ch}$	0.95	$\zeta^{dis}$	0.90

2) use the collected experience to update policy  $\pi_\psi(\mathbf{a}_t|\mathbf{S}_t)$ . By running these two tasks repeatedly, an optimal policy  $\pi^*(\mathbf{a}_t|\mathbf{S}_t)$  is expected to be determined once  $\psi$  is converged. Depending on the detail of policy update, there are many deep reinforcement learning algorithms (e.g., proximal policy optimization, deep deterministic policy gradient) suitable for the above-mentioned control problem. In this study, we choose to use a direct policy search algorithm based on evolution strategy (ES-RL) [15] due to its scalability. It is worth noting that ES-RL is gradient-based, instead of a purely heuristic algorithm. Specifically, it uses ES for zero-order gradient approximation: at each iteration, ES-RL first perturbs the policy parameters  $\psi$  by generating some random noises from an isotropic multivariate Gaussian  $e_i \sim \mathcal{N}(0, 1)$  for  $i \in \{1, \dots, n\}$ . Based on these noises,  $n$  new policies parameterized by  $\psi_t + \sigma e_i$  are obtained ( $\sigma$  is the standard deviation of the perturbing noise). These “mutated” policies  $\pi_{\psi_t + \sigma e_i}(\mathbf{a}_t|\mathbf{S}_t)$  will be used for experience generation and obtain the stochastic return from the environment ( $G_i = \sum_{t \in \mathcal{T}} r_t$ ). Using these results, the new policy will be updated using  $\psi_{t+1} = \psi_t + \iota \frac{1}{n\sigma} \sum_{i=1}^n G_i e_i$ , in which  $\iota$  is the learning rate. With adequate iterations,  $\psi_t$  is expected to converge to an optimal combination and an optimal policy is trained.

## V. CASE STUDY

### A. Experiment Settings

In this section, we study a specific load restoration problem with a control horizon of six hours (repair time needed to restore the service from upstream substation) and a control interval of five minutes. Table I details the parameters used in this case study. The baseline MPC was implemented using JuMP in Julia 1.4 and was solved on a MacBook Pro with Intel Core i7 Quad-Core Processor (2.80GHz) and 16GB RAM using the GLPK open source solver.

Real world PV and wind generation profiles in two months (July and August) are collected as the exogenous data used for training and testing. Use different renewable generation set in training and testing is to resemble the reality and to provide a more objective evaluation. Specifically, Fig. 2 shows the setup of controller-simulator interaction. First, a scenario sampler will randomly sample six hours of renewable generation profiles (i.e.,  $\mathbf{P}_{[t_0, t_0+1, \dots, t_0+T]}^{\mathcal{R}}$ ) from the exogenous data pool.

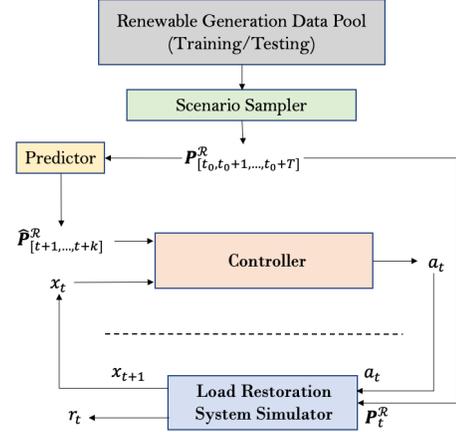


Fig. 2. Illustration of controller comparison setup, the performance of MPC and RLC can be evaluated by swap in either controller. This setup is also used for RLC training. Recall from Section IV-A, the RL state  $\mathbf{S}_t$  consists of renewable forecast and the system state  $x_t$ , two inputs for the controller.

Based on the selected period, the renewable forecaster mentioned in Section III is able to generate forecast profiles at each step (i.e.,  $\hat{\mathbf{P}}_{[t, t+1, \dots, t+k]}^{\mathcal{R}}$ , in which  $k$  is the forecast horizon). At step  $t$ , the controller, either MPC or RLC, makes decision based current system state (i.e.,  $x_t = [S_t^\theta, \hat{E}_t^\mu, \mathbf{1}^T \mathbf{P}_t / \mathbf{1}^T \mathbf{P}, t]$ ) and forecast made at this step. Based on the action, the system simulator will update system state and evaluate the reward at this step  $r_t$ . By the end of the control horizon, the total reward  $\sum_{t \in \mathcal{T}} r_t$  is used to evaluate the controller performance.

### B. RLC Training

The RL controller is trained on the High Performance Computing system at the U.S. Department of Energy’s National Renewable Energy Laboratory (NREL). Each Eagle computing node has dual Intel 18-core processors and in this study ten nodes are utilized for the training resulting in a cluster of 323 parallel workers for the optimal policy search using ES-RL algorithm mentioned in Section IV-B. Specifically, for each step of gradient update (i.e., training epoch), parallel workers together collect 5000 episodes of training data (equivalent to  $72 \times 5k = 360k$  control steps) by interacting with simulators. The learning step used in our study is 0.001 and the discount factor  $\gamma = 1$  since this control problem has a finite control horizon. Fig. 3 shows RL learning curves in three trials. The RL learning tasks are scheduled to be run for around two hours on Eagle, but from the resulting learning curves, it can be seen that learning has already converged to a policy by the end of the first hour. The trained RL agent is used in the following performance evaluation.

### C. Performance Evaluation

This section conducts performance evaluation of the proposed RL controller from three aspects: first, the controller’s reaction towards intermittent renewable generation is examined; second, a comparison between MPC and RLC is con-

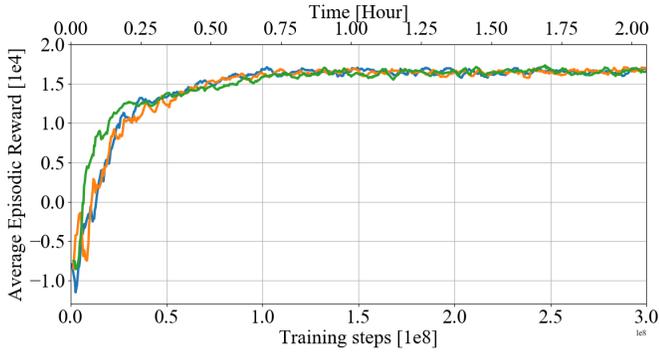


Fig. 3. Learning curves showing the average episode reward increases with the increase of training steps (i.e., amount of total experience collected). Three curves represent for three different trials. The bottom x-axis shows the training steps and the top x-axis shows the corresponding wall time.

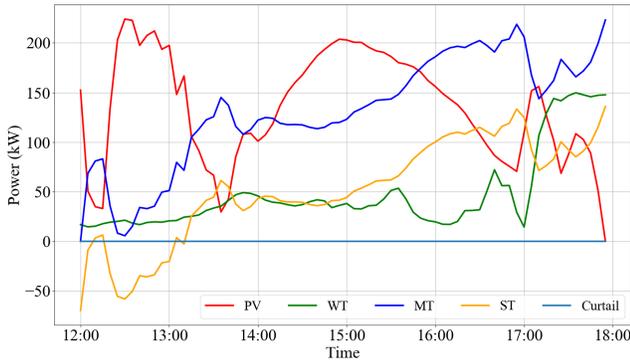


Fig. 4. Generation output from four DERs and renewable curtailment over one six-hour scenario. Substation outage occurs at 12:00 and the fault cannot be repaired until 18:00.

ducted using a specific scenario; and third, the MPC and RLC comparison based on multiple testing scenarios is studied.

1) *Controller response*: Fig. 4 shows the generation profiles of the four DERs ( $\mathcal{R} = \{\rho, \omega\}$  and  $\mathcal{G} = \{\mu, \theta\}$ ) over the 6-hour control horizon of a specific scenario, in which the substation outage occurred at 12:00. By observing this scenario, it can be seen that although the PV generation changes violently in this afternoon, the RLC can properly control the micro-turbine and storage system at each step to compensate for the variability in renewable generation, which helps providing continued support to loads that have been restored. Moreover, between 12:15 and 13:00, when solar generation is abundant, instead of greedily using it to restore more loads immediately, the RLC choose to charge the power to the storage at this time since it has learned from training that restoring load too soon might lead to penalty due to failure to sustain restored loads. From this scenario, we can see that RLC can make some seemingly reasonable decisions at each step. To further study the quality of these decisions, RLC is compared with the baseline MPC over several scenarios.

2) *RL vs. MPC: Single Scenario*: Fig. 5 shows the comparison between two restoration processes controlled by a MPC and an RLC. MPC started with picking up first eight loads

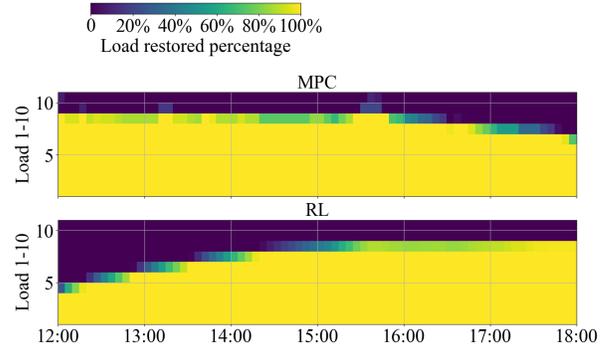


Fig. 5. Load restoration comparison between MPC and RL controller. According to Table I, load 1 to 10 have monotonically decreasing priority. Color bar shows the load restoration level with yellow means 100% restoration and black means 0%.

with higher priority but at the end of the control horizon only six loads are fully restored. This is mainly due to the forecast error of renewable generation, the difference between forecast and reality causes poor planning and thus in the later half of the control horizon, MPC realized that there isn't enough remaining fuel, stored energy and renewable generation and has to shed load 7, 8 again. Also, load 8 was repeatedly restored and partially shed until it was totally shed, providing an unreliable service to this load.

In contrast, RLC learned from experience (during training) that predicted information cannot be totally relied on and formed its own control policy under these uncertain scenarios. From Fig. 5, it can be seen that RLC started with restoring the most important four loads and gradually restore more loads as it become more confident to handle renewable uncertainty. For loads that have been restored, it is very likely that they will be served continuously.

3) *RL vs. MPC: Multiple Scenarios*: To get a more comprehensive comparison, each of these two controllers are examined in 50 scenarios, 25 each in training period and testing period. Comparing the RLC performance in both training and testing scenarios is to examine if the performance deteriorates in unseen testing scenarios.

Fig. 6 and Table II present the comparison results, from which the following observations/conclusions can be drawn:

- In general, RLC performs better than MPC considering the maximized objective's mean, median and distribution.
- The distribution of the objective function values for the MPC has higher variance than that of the RLC. This is due to MPC's strong dependence on the renewable forecast: in those scenarios where forecast is relatively better, MPC has good performance but when forecast error is large, MPC tends to have much worse performance.
- RLC performance does not become inferior under the unseen testing scenarios. The reason is that even though the RLC has not seen the exact scenarios in testing cases, the distribution of renewable generation and forecast error is similar in training and testing (two adjacent months

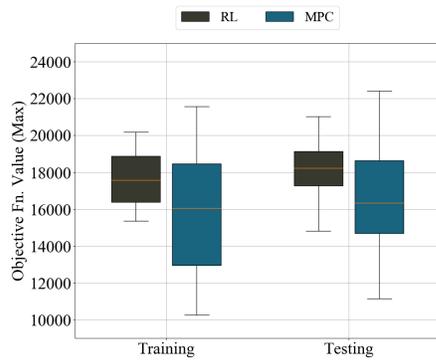


Fig. 6. Objective function values comparison between MPC and RLC. A total of 50 scenarios are selected among training and testing data (evenly split), and each boxplot shows the distribution of 25 scenarios.

TABLE II  
AVERAGE OBJECTIVE FN. VALUES OVER 25 SCENARIOS IN EACH CASE

Controller	Scenarios	
	Training	Testing
MPC	15811.77	16647.85
RLC	17633.46	18044.50

with the same forecasting technique).

To sum up, in this section, a comparison between RLC and MPC is conducted and we discover that due to the training experience, RLC has learned about how to handle imperfect forecast and provide a steadier and more reliable restoration process, which achieves a higher average objective function value than MPC. In addition to the experimental findings above, we also identify an issue/improvement area: in some cases, RLC does not deplete the fuel and stored energy in the controllable DERs by the end of the control horizon, which could be sub-optimal.

## VI. DISCUSSION AND FUTURE WORK

In this study, based on a load restoration problem on a single bus system, we conduct a preliminary study comparing the performance of a reinforcement learning controller (RLC) with that of a deterministic model predictive controller (MPC), considering the uncertainty from renewable generation. The results show that RLC can achieve better control performance than MPC: By learning from experience, RLC is able to learn an optimal control policy to handle forecast error.

Though preliminary results show that RLC is promising to provide better control, the following tasks will be addressed in our future work. First, more complex system will be tested. Additional complexity comes from three aspects:

- System complexity: power flow and network constraints will be considered.
- Operation complexity: a more versatile RLC will be trained to handle different operation conditions; e.g., different initial conditions such as  $s_0$  and  $E^\mu$ .
- Uncertainty complexity: uncertainty in the restoration time will be taken in consideration.

Second, although forecast values are used in the RL formulation, RLC does not necessarily need explicit renewable forecast values for decision-making. Instead, historical renewable generation can be used for the RL state and during the RLC training, it has the ability to learn a forecasting tool implicitly to facilitate decision-making.

Finally, whether an RL controller should be trained using historical data from the past 30 days, 120 days or half a year is worth investigating. This is because more historical data provides more scenarios to train a versatile and capable controller. Conversely, historical data from half a year ago might have different distribution and could change the overall distribution the RL training is optimizing against, thus causing sub-optimal behavior.

## REFERENCES

- Alexis Kwasinski, Fabio Andrade, Marcel J Castro-Sitiriche, and Efraín O'Neill-Carrillo. Hurricane maria effects on puerto rico electric power infrastructure. *IEEE Power and Energy Technology Systems Journal*, 6(1):85–94, 2019.
- Tina Comes and Bartel Van de Walle. Measuring disaster resilience: The impact of hurricane sandy on critical infrastructure systems. *ISCRAM*, 11:195–204, 2014.
- Salman Mohagheghi and Fang Yang. Applications of microgrids in distribution system service restoration. In *ISGT 2011*, pages 1–7. IEEE, 2011.
- Zhiwen Wang, Chen Shen, Yin Xu, Feng Liu, Xiangyu Wu, and Chen-Ching Liu. Risk-limiting load restoration for resilience enhancement with intermittent energy resources. *IEEE Transactions on Smart Grid*, 10(3):2507–2522, 2019.
- Zhaoyu Wang, Bokan Chen, Jianhui Wang, and Chen Chen. Networked microgrids for self-healing power systems. *IEEE Transactions on smart grid*, 7(1):310–319, 2015.
- Amin Gholami, Tohid Shekari, Farrokh Aminifar, and Mohammad Shahidehpour. Microgrid scheduling with uncertainty: The quest for resilience. *IEEE Transactions on Smart Grid*, 7(6):2849–2858, 2016.
- Wei Yuan, Jianhui Wang, Feng Qiu, Chen Chen, Chongqing Kang, and Bo Zeng. Robust optimization-based resilient distribution network planning against natural disasters. *IEEE Transactions on Smart Grid*, 7(6):2817–2826, 2016.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fiedjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.
- Gregory Kahn, Adam Villafior, Bosen Ding, Pieter Abbeel, and Sergey Levine. Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2018.
- Hanchen Xu, Xiao Li, Xiangyu Zhang, and Junbo Zhang. Arbitrage of energy storage in electricity markets with deep reinforcement learning. *arXiv preprint arXiv:1904.12232*, 2019.
- Souhaib Ben Taieb and Gianluca Bontempi. Recursive multi-step time series forecasting by perturbing data. In *2011 IEEE 11th International Conference on Data Mining*, pages 695–704. IEEE, 2011.
- David L Woodruff, Julio Deride, Andrea Staid, Jean-Paul Watson, Gerrit Slevogt, and César Silva-Monroy. Constructing probabilistic scenarios for wide-area solar power generation. *Solar Energy*, 160:153–167, 2018.
- Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.
- Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.