Modeling and Control of a Connected Photovoltaic Microgrid System: Investigation of Fuzzy Q-Learning Control Methods

Romita Biswas

Whiting School of Engineering, Johns Hopkins University

Abstract

The dependency on the electric transmission network is increasing at an exponential pace due to the advent of electric vehicles on the road. As the current power grid is centralized it has been subject to frequent power shortages, transmission losses, and domestic terrorist attacks. The integration of microgrids is a viable solution to current and future power grid challenges. Microgrids are smaller modular power generation units that operate near the end user reducing transmission losses and increasing power reliability. These microgrids are made up of a local energy storage unit, a power generation source, and the ability to connect and disconnect to the main grid. A microgrid's purpose is to provide its local community with quality electricity at all times. This paper investigates the control and operation of a proposed microgrid system for the local community of Ward 6 - Washington, DC. An economically feasible microgrid is designed and a control policy using Q-Learning methods is explored.

Keywords - microgrid, reinforcement learning, Q-learning, modeling, simulation, solar, energy storage

1. Introduction

The current electric power grid is currently under high stress due to aging equipment, reliance on fossil fuels, and increased demand from electromobility. The current power grid has seen a spike in the number of power outages and terrorist attacks. Changing weather patterns have placed high stress on centralized grid systems in certain regions like Texas where a massive power grid failure resulted in the loss of power for over 4.5 million homes [9]. The nation's vulnerable electricity infrastructure has also become a target for domestic terrorism [5]. Growth in electric vehicle ownership will place further stress on an already failing power grid. To better deal with current and future challenges, decentralization of the power grid is needed.

A growing solution for distributed power generation is microgrids. Microgrids are able to facilitate distributed generation and high penetration of renewable energy sources [2][7]. Microgrids have the advantage of being more reliable because of its self dependence on its own power generation unit and its ability to directly serve a local demand without passing through a transmission network. Microgrids are also able to directly connect and disconnect to the main grid at various transmission points. This allows them to serve local demand even if local power generation fails to meet the demand.

A microgrid is able to be modeled as a complex system - a collection of interacting elements performing a nonlinear task [1]. A complex system is distributed and self-organized. Any complex system is further able to be viewed as a System of System (SoS) [7]. A System of System aims to work together to achieve a common goal by communicating and transmitting tasks [1]. The main purpose of any microgrid is to work with various types of renewable energy sources and request power from the main grid to reliably meet the overall local load demand. Rather than using traditional control policies like PID or droop control, the application of reinforcement learning is investigated. Reinforcement learning allows the agent to learn the best actions to take given a certain state the microgrid is in. Fuzzy Q-Learning is explored in order to meet the learning goal of understanding what action to take given the reliability rating of the system at the current step.

2. Case Study Scope

The Mid-Atlantic region of the United States is predominantly fueled by natural gas and petroleum. Natural gas prices have lowered over the decade making it a preferred solution compared to the existing coal plants in the region. Even with lower prices and lower carbon emissions, cities including Washington D.C. have set goals to achieve fully clean energy by 2032. An increasing number of solar plants have been set up in the region to diversify the electricity production. However in order to transition specific Wards or neighborhoods to clean energy in Washington, a reliable control policy must be developed in order to serve local residential demand.

It has been found that the Mid-Atlantic region suffers from variable weather conditions throughout the year, presenting a challenge to policy makers in Washington. Despite being a clean and abundantly available source, wind energy in the region suffers from lack of energy density and intermittency [3][4][18]. In this region integrating renewable energy can present a risk in providing continuous power and meeting peak demand. Therefore wind power systems have been omitted from the study. In this paper only the integration of solar power is explored. It is assumed that the 100 MW battery used has 100% efficiency.

The cost of electricity is highest in August. Therefore the study has been scoped to assess the system in the month of August only. The residents of Washington consume an average of 10,5011.17 MWh during the month of August [14]. Ward 6 has been calculated to consume an average of 78.74 MWh, based on its resident proportion, in August. The city of Annapolis, Maryland has been chosen to represent the main grid in how it buys and sells energy to Washington-Ward 6. Local solar elevation data and cloud cover data has been analyzed to generate solar power generation data for Ward 6.

2.1. Solar Generation Equations

Solar panels use solar radiation to generate electricity, transforming photons that hit the photovoltaic panels into DC current. When the photons hit the panel they are absorbed by the panel's semiconducting silicon material. The movement of the electrons generates the DC current. Solar panels rely on solar elevation, cloud cover, topography, and solar irradiance.

For this paper the solar panel chosen is the Vertex 670W+ Module. The area of the panel is A = 2.9106 from 66 210 mm silicon wafers. The solar panel yield is rated at 21.6%. The losses are estimated to be 0.0651 based on DC to AC loss and random loss [17].

The power generated by the solar panel is proportional to the total area:

$$E = AyHr$$

where A is the area of the solar panel, y is the solar panel yield, H is the solar radiation, and r is the performance ratio. The solar radiation changes throughout the year as it is dependent on solar elevation and cloud cover. The solar radiation for a given hour is:

$$H = R_0(1 - 0.75\eta^{3.4})$$

$$R_0 = 990sin(\frac{\varphi_{tp} + \varphi_p}{2}) - 30$$

where $R_0$is clear sky insolation, $\eta$is cloud cover percentage, $\varphi_{tp}$is the solar elevation at the previous hour, and $\varphi_p$is the solar elevation at the current hour [17][18]. Cloud cover data was collected for the month of August over 5 years from 2018 to 2022 [9].

2.2. Main Grid Pricing Strategy

The main grid in this paper is represented as an unlimited source of energy to the local grid. The main grid acts as a bidder and provider. It provides a bidding price to buy excess energy from the local grid. The bidding price is represented by the residential cost of electricity per hour for the city of Annapolis. When acting as a provider the request price is set dynamically depending on the local demand.

The real price distribution for DC residents was used [14]. The following strategy to calculate dynamic request price was used:

$$X_{price} = \mu_{price} + Z_{local\ demand} * \sigma_{price}$$

Given the distribution, it was determined what the mean and standard deviation is. The current local demand is mapped to a Z score for local demand distribution. The Z score for local demand is mapped to the raw price using the local request price parameters.

2.3. Determining the Number of Panels

Using the local energy demand data and the calculated local solar power generation data, the number of panels for each hour was determined by the simple equation:

$$Num_{panels} = \frac{Energy\ Demand}{Solar\ Power} \forall\ hours$$

The number of panels varied dramatically depending on the solar output. In order to choose a reliable number of panels, the Z score of 1.96 mapping to a confidence level of 95% was chosen. The distribution parameters across the matrix Num_{panels} were generated. The final Num_{panels} was chosen to be mapped to the Z score 1.96.

2.4. Data Handling

All data used throughout this paper has been partitioned by the hour over 31 days of August for 5 years such that the appears like this where n = 155 days:

$$\begin{pmatrix} hour\ 0 \\ hour\ 1 \\ \ldots \\ hour\ 22 \\ hour\ 23 \end{pmatrix} = \begin{pmatrix} weather\ 0 & . & . & . & weather\ n-1 \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ weather\ 23 & . & . & . & weather\ n-1 \end{pmatrix}$$

3. Reinforcement Learning

The objective of reinforcement learning is to find a policy - a mapping from states to actions - that maximizes a reward. Learning is essentially a trial and error process where an agent learns through exploration and exploitation by receiving feedback from the system. The learning agent reinforces itself through successes and failures. Actions that are good when performed in a given state are rewarded, while actions that are bad are punished.

3.1. Q-Learning

Q learning is a type of model-free reinforcement learning, where the agent does not need to know all the details about the model. It is like a black-box type of reinforcement learning. Q-learning learns through the action-value function Q that maps state-action pairs to returns. An agent tries an action at a state and evaluates its reward immediately. Q-learning is less memory intensive as it maintains only one state-action pair given it is one-step Q-learning. The number of state-action pairs correspond to the number of N-steps in Q-learning.

Q-learning (off-policy TD control) for estimating $\pi \approx \pi_*$

Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(terminal, \cdot) = 0$

Loop for each episode:
  Initialize S
  Loop for each step of episode:
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma \max_a Q(S', a) - Q(S, A)\big]$
    $S \leftarrow S'$
  until S is terminal

Fig 1: Q-Learning Algorithm [13]

3.2. Fuzzy Reasoning Logic

In classical set theory, elements belong to a crisp set in a binary manner - yes or no. In fuzzy set theory, an element belongs to a set up to a degree. A degree of membership is associated with an element. Fuzzy logic allows for ambiguity by allowing the usage of fuzzy if then rules. This fuzzy rule building allows for relations among fuzzy variables to be represented using linguistic terms like very or somewhat.

3.2.1. Fuzzy Q-Learning

This paper used fuzzy Q-learning to generate a state space. The crisp set of inputs or observations are fuzzified. The fuzzified observations are then mapped to a fuzzy state using a fuzzy inference engine. The degree of membership to each linguistic value determines the strength of a certain rule in an inference engine. An inference engine is a set of if then rules. The degree of membership of each if variable determines the strength of the rule which in turn defines the degree of the output variable. As in, the strength of each rule determines the degree to which the agent is in a state [6].

4. *Modeling System Interactions*

The local microgrid system of Ward 6 is powered with solar energy. The goal of the system is to serve the local demand reliably through generated power, stored power, and requested power. For simplicity there is no limit to how much power can be requested from the main grid. However to represent real power grid interactions, there is a cost when requesting energy transfers from another grid (main-grid). When there is a shortage of power the Microgrid takes action to minimize that shortage. When there is an excess of power the Microgrid takes action to maximize the profit on the excess without hurting the system in the future.

In this environment, the interactions of a Microgrid with its local demand is infinite. However an infinite horizon is computationally intensive. This system has no reasonable termination state as local energy demand must always be met. An episode has been defined to be 24 hours. The start and end of the episode coincide with the lowest energy demand.

A step in this environment is a time step of an hour. The step changes controlled variables that the system has control over. The agent does not observe variables that are uncontrolled like the main grid parameters. The Microgrid is responsible for managing its own shortages and excess. Through managing its shortage and excess it can measure its own reliability through fuzzy logic systems. A highly reliable system means the Microgrid is self-sufficient. A low reliable system indicates the Microgrid is dependent on the main grid for support.

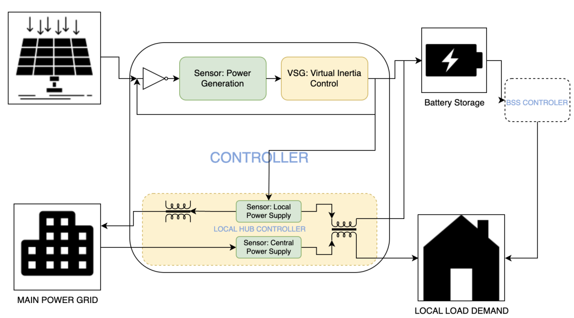

Fig 2: Model of System in Case Study

4.1. *Observation*

These observations are observed by the learning agent when interacting with the environment of the Microgrid. Resets to the environment start the agent at random days at hour 0.

Observation	**Notes**
Current Energy Demand (MWh)	-
Current Energy Supply (MWh)	-
Battery Supply (MWh)	-
Total Supply (MWh)	Current Energy Supply + Battery Supply
Current Delta (MWh)	Total Supply - Current Demand

Main Grid Request Price ($)	Dynamic based on necessity of local demand. Larger the delta, larger the cost to request.
Main Grid Bid Price ($)	Constant $0.14/kWh
Storage Price ($)	Constant $0.05/kWh
Current Econ Return	$Profit

4.1.1. Fuzzy State

In order to best preserve information about the state of the system given the observations, the state the learning agent learns is a 2D state.The state space represents the degree of autonomy of a system. The state is defined as:

S = [dependence degree, independence degree]

The dependence degree should measure how dependent the Microgrid is on the main grid. The independence degree should measure how independent the Microgrid is. A high degree of independence indicates high reliability if the microgrid system should need to disconnect from the main grid.

The fuzzy inference engine uses 8 rules. Technically fuzzy systems increase the complexity by increasing the number of fuzzy variables. The fuzzy system also has to go through the process of defuzzification to reduce the dimensionality size. There are various methods in reducing the complexity depending on how the overall fuzzy set for each output membership function is treated. In this paper we operate upon each fuzzy rule essentially approximating the crisp output. For each output linguistic variable, the probabilistic t-conorm operates on all of the individual strengths for all the corresponding rules. This iterative probabilistic t-conorm operation reduces the number of variables from 8 to 2.

Algorithm: Probabilistic T-conorm
```
x = [x1, x2, x3, x4]
    idx = 0
    for i in range(0, 4):
        a = x[i]
        for j in range(i + 1, 4):
            b = x[j]
            Y[0, idx] = a + b - (a * b)
            idx = idx + 1
```

4.1.1.1. Rules

The final fuzzy output state of s = [dependence degree, independence degree], is generated through a fuzzy inference engine. The rules require the crisp inputs of the observations be fuzzified before inputting into the inference engine. The rules are as follows:

- Rule 1 if *current_delta* is *not_reliable* and *battery_usage* is low and *current_economic_profit* is low, then Z is not independent
- Rule 2 if *current_delta* is *not_reliable* and *battery_usage* is low and *current_economic_profit* is high, then Z is not independent

- Rule 3 if *current_delta* is *not_reliable* and *battery_usage* is high and *current_economic_profit* is low, then Z is not independent
- Rule 4 if *current_delta* is *not_reliable* and ba*battery_usage*tt is high and *current_economic_profit* is high, then Z is independent
- Rule 5 if *current_delta* is *reliable* and *battery_usage* is high and *current_economic_profit* is high, then Z is independent
- Rule 6 if *current_delta* is *reliable* and *battery_usage* is low and *current_economic_profit* is high, then Z is independent
- Rule 7 if *current_delta* is *reliable* and *battery_usage* is high and *current_economic_profit* is low, then Z is independent
- Rule 8 if *current_delta* is *reliable* and *battery_usage* is low and *current_economic_profit* is low, then Z is not independent

4.1.1.2. Membership Functions

The membership functions are used to calculate the degree of belonging or membership a value has to a set. There is a degree of membership to how "reliable" or "unreliable" a value is.

4.1.1.2.1. Output Membership Functions

The output membership function is generated by taking the T-norm or in this case the algebraic product of the degree of all three fuzzy inputs. That value is then compared to the value of the whole output membership function. The T-conorm or maximum value is taken. This value indicates the strength of autonomy of the system. Below is an example of equations used to determine the rule strength:

weight1 = not_reliable * low_batt * low_profit
rule1 = np.max(np.multiply(weight1, z_not_indepedent))

The output membership functions are defined below:

$$y = \frac{1}{1 + e^{\frac{a}{Z-c}}}$$

For the degree of dependence, the parameter values are *a = 30, c = 30.* For the degree of independence, the parameter values are *a = 30, c = 70.*

4.1.1.2.2. Input Membership Functions

The input membership function is direct and is applied to only one crisp input. The crisp inputs of current_delta, battery_supply, and current_econ_return are mapped to a fuzzy input represented by a set. The same equation for a two sided gaussian is used for all three linguistic variables.

- if x <= c1
  - $y = e^{\frac{1}{2}*(\frac{x-c_1}{\sigma_1})^2}$
- if x < c2 and x > c1
  - $y = 1$
- else

- $y = e^{\frac{1}{2} * (\frac{x - c_2}{\sigma_2})^2}$

The variable current_delta is evaluated to analyze what the natural reliability of the system is. As in what the reliability of the system is given only solar power generation. There are 2 linguistic values for this variable: reliable or not reliable. For the degree of “not reliable”, the parameter values are *c1 = 0, c2 = 30, sig1 = 20, sig2 = 20.* For the degree of “reliable”, the parameter values are *c1 = 65, c2 = 100, sig1 = 20, sig2 = 20.*

The variable battery_supply is evaluated to analyze how much the battery is used. There are 2 linguistic values for this variable: used or not used. For the degree of “used”, the parameter values are *c1 = 75, c2 = 100, sig1 = 25, sig2 = 25.* For the degree of “not used”, the parameter values are *c1 = 0, c2 = 25, sig1 = 25, sig2 = 25.*

The variable current_econ_return is evaluated to analyze how profitable the system is. There are 2 linguistic values for this variable: profitable or not profitable. As profit can be extremely negative or positive depending on the action of the agent, the values have been limited such that any value below -10000 is limited to -10000. Any value above 15000 is limited to 15000. For the degree of “profitable”, the parameter values are *c1 = -10000, c2 = 35, sig1 = 150, sig2 = 150.* For the degree of “not profitable”, the parameter values are *c1 = 40, c2 = 15000, sig1 = 150, sig2 = 150.*

4.2. Actions

By default the agent always serves the local demand immediately with the current energy generation and battery supply, this allows for the system to maintain a level of virtual inertia control. This is critical to energy generation systems that are not motor driven. Once the current energy demand is met there is a delta. The actions that manage the delta are of interest to this paper. The actions in this environment are:

- Action 0: Sell (delta) to Main
- Action 1: Store (delta) to Battery
- Action 2: Buy (delta) from Main

4.3. Reward

The agent receives a reward for certain actions taken in a given state. The default reward at every step is +10. The agent always receives a reward for serving the local demand. The reward of interest is dependent on delta. The reward shaping is as such:

- Reward 0: Main Grid Request Price * Delta → i.e. selling during a shortage is very bad
- Reward 1: Storage Price * Delta → storing during a shortage isn’t rewarded
- Reward 2: (-1) * Main Grid Bid Price * Delta → negation taken so that the agent is rewarded for requesting help during a shortage. However it should not request if in a state of excess

5. Simulation

This section presents the simulation of the local microgrid for Ward 6. The figure below shows the energy demand trend for the Mid-Atlantic region, which represents the DC residential demand. The system is grid connected with battery storage. The goal is for the microgrid to take the best action given the state of autonomy the system is in. The state of autonomy has been represented by the vector s = [dependence, independence]. The system's default action is to always serve the local demand through its available solar power and battery storage. The simulation explores how the agent learns to manage the shortage or excess of the microgrid system after serving the local demand at time t.

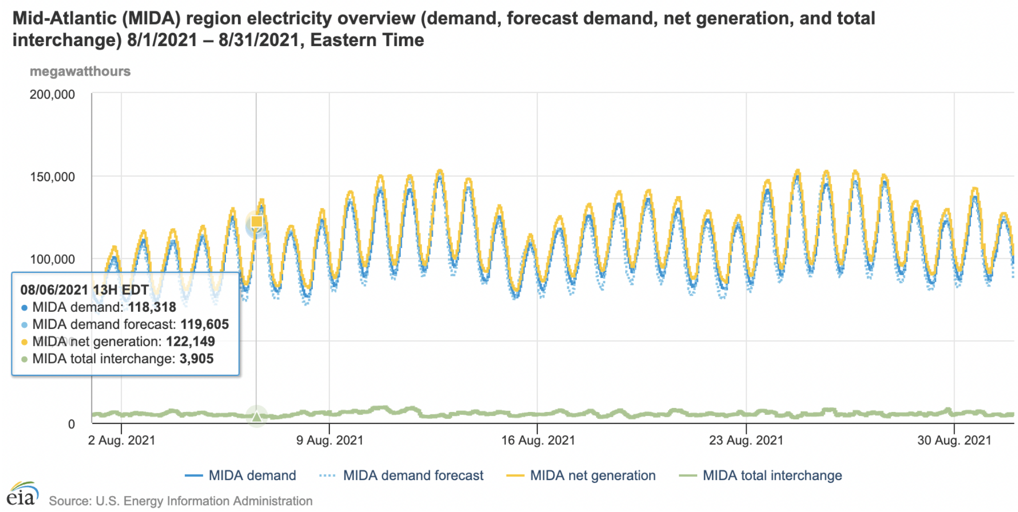

Fig 3: MIDA Electricity Demand Overview of August 2021 [15]

The simulation setup treats one episode as 24 hours. The environment resets at a random day at hour 0. As the experiment is limited to the month of August, each hour of the day is treated as a random process using the data from each hour of all days. The experiment is run for a various number of episodes in sets of 155 to ensure that all the data has been explored at least once.

5.1. Reward Shaping

Reward shaping was explored to investigate the effects of different scales on the agent's ability to learn effectively. The original intention behind the reward was that the agent would be heavily penalized for selling during a shortage and purchasing during an excess of power. For the following experiment run the agent completed 10 sets of 155 episodes. Below is the raw average reward without any reward shaping.

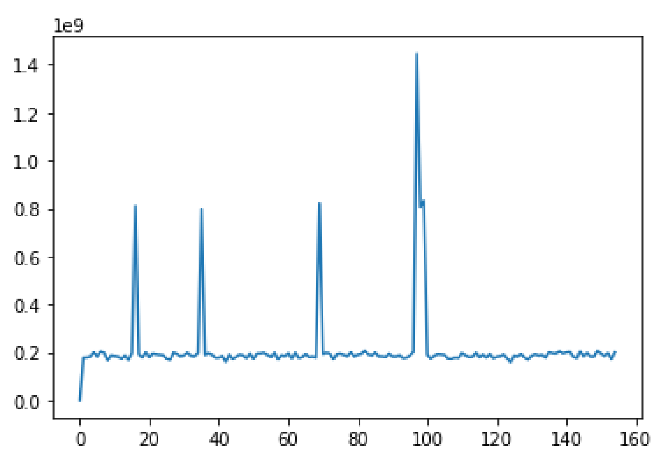

Fig 4: Average Reward for Raw Reward without Shaping - 10 sets

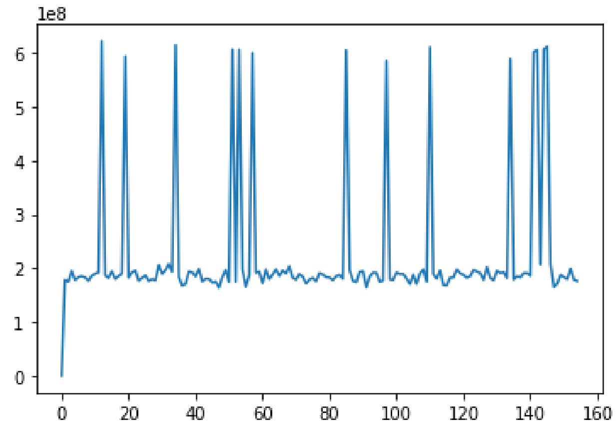

Fig 5: Average Reward for Raw Reward without Shaping - 15 sets

To minimize the spikes in the returned reward, reward shaping was explored. The reward was scaled so that the absolute maximum value of a reward would be 10000. If the absolute value is above 10000, the factor is calculated by taking the current reward value divided by 10000. The value is then scaled down by the factor value . The average reward is shown below. Notice that scaling the value had a negative impact on the learning, where the agent failed to maximize the reward and seems to have learned a bad policy.

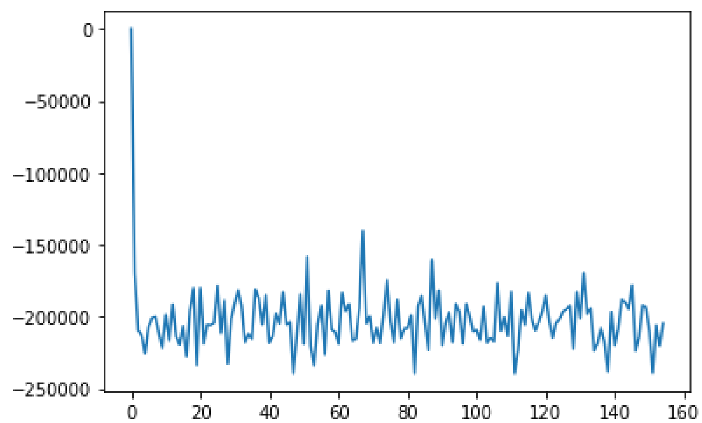

Fig 6: Average Reward for Scaled Reward Shaping limit abs(10000) - 10 sets

The reward was scaled so that the absolute maximum value of a reward would be 100,000. If the absolute value is above 100,000, the factor is calculated by taking the current reward value divided by 100,000. The value is then scaled down by the factor value . The average reward is shown below. Notice that scaling here seems to be successful. The spikes in the rewards have been reduced and the policy seems to have converged successfully on a stable reward.

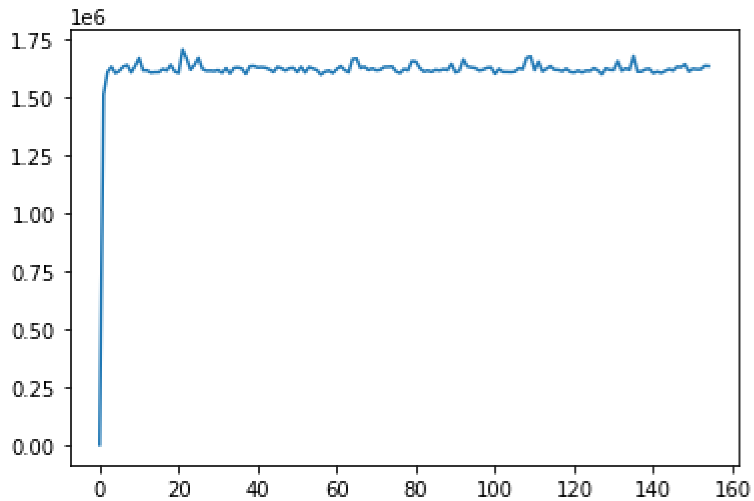

Fig 7: Average Reward for Scaled Reward Shaping limit abs(100,000) - 15 sets

6. Conclusion

This paper presented the modeling, simulation, and control of a viable microgrid for Ward 6 of Washington D.C. This microgrid was treated as a complex system or System of System where each subsystem acted independently of each other but still interacted. The system was made up of solar power generating units, a battery storage, load demand, and a connected stable main grid. A fuzzy Q-Learning algorithm was explored to examine if an optimal policy could be developed to control the local grid interactions when faced with an energy shortage or excess. The simulation was run directly on real data collected for Washington. The results displayed inconsistencies with the learning when using raw reward. Reward shaping through scaling failed on the first simulation attempts as the limit was too low. Once the limit was increased to 100,000 the policy was able to converge upon a reward. It is assumed that an optimal or good enough policy has been found through fuzzy Q-learning for the system presented in this paper. Future work includes additional system components simulating the interaction with another dynamic live microgrid.

7. References

1. Ahmad Alzahrania, Mehdi Ferdowsia, Pourya Shamsia, Cihan H.Daglib, C., b, & AbstractComplex computer systems and electric power grids share many properties of how they behave and how they are structured. A microgrid is a smaller electric grid that contains several homes. (2017, October 9). *Modeling and simulation of Microgrid*. Procedia Computer Science. Retrieved December 14, 2022, from https://www.sciencedirect.com/science/article/pii/S1877050917318574
2. A. Llria, O. Curea, J. Jimenez, H. Camblong, Survey on microgrids: unplanned islanding and related inverter control techniques, Int. J. Renew. Energy 36 (2011) 2052–2061.
3. Degeilh, Y., & Gross, G. (2014, August 15). Stochastic simulation of power systems with integrated intermittent renewable resources. International Journal of Electrical Power & Energy Systems. Retrieved December 9, 2022, from https://www.sciencedirect.com/science/article/pii/S0142061514004840.
4. Global wind atlas. Global Wind Atlas. (n.d.). Retrieved December 9, 2021, from https://globalwindatlas.info/.
5. Girgis, L., & Bernton, H. (2022, December 8). PSE substations among six attacked in Pacific Northwest in November. The Seattle Times. Retrieved December 14, 2022, from https://www.seattletimes.com/seattle-news/law-justice/pse-substations-among-five-attacked-in-pacific-northwest-in-november/

6. Kofinas, P., Dounis, A., & Vouros, G. (1970, January 1). *Fuzzy Q-learning for multi-agent decentralized energy management in microgrids: Semantic scholar*. Applied Energy. Retrieved December 14, 2022, from https://www.semanticscholar.org/paper/Fuzzy-Q-Learning-for-multi-agent-decentralized-in-Kofinas-Dounis/069f6760e7e45df6cccb006905b64742607ff55f
7. M.S. Mahmouda, Azher Hussaina M.A. Abidob (2014, February 6). *Modeling and control of microgrid: An overview*. Journal of the Franklin Institute. Retrieved December 14, 2022, from https://www.sciencedirect.com/science/article/pii/S0016003214000180
8. NOAA Solar Position calculator. (n.d.). Retrieved December 14, 2022, from https://gml.noaa.gov/grad/solcalc/azel.html
9. P. Barker, D. Herman, Technical and economic feasibility of microgrid- based power systems, in: Proceedings of the Seventh EPRI Distributed Resources Conference and Exhibition Dallas, TX, 20–22 March 2002.
10. *Real-time operating grid - U.S. Energy Information Administration (EIA).* Real-time Operating Grid - U.S. Energy Information Administration (EIA). (n.d.). Retrieved December 9, 2022, from https://www.eia.gov/electricity/gridmonitor/dashboard/electric_overview/regional/REG-MIDA.
11. *Research on autonomous operation control of microgrid based on deep ...* (n.d.). Retrieved December 14, 2022, from https://ieeexplore.ieee.org/document/9713298/references
12. Škiparev, Vjatšeslav & Machlev, Ram & Chowdhury, Nilanjan & Levron, Yoash & Petlenkov, Eduard & Belikov, Juri. (2021). Virtual Inertia Control Methods in Islanded Microgrids. Energies. 14. 1562. 10.3390/en14061562.
13. Sutton, R. S., & Barto, A. G. (2020). *Reinforcement learning: An introduction*. The MIT Press.
14. U.S. Bureau of Labor Statistics. (n.d.). *Average energy prices - washington DC*. U.S. Bureau of Labor Statistics. Retrieved December 14, 2022, from https://www.bls.gov/regions/mid-atlantic/data/averageenergyprices_washingtondc_table.htm
15. U.S. Energy Information Administration - EIA - independent statistics and analysis. District of Columbia - State Energy Profile Overview - U.S. Energy Information Administration (EIA). (n.d.). Retrieved December 9, 2021, from https://www.eia.gov/state/?sid=DC#tabs-2.
16. US/vertex 670W+ module. Trina Solar. (2021, September 30). Retrieved December 9, 2021, from https://www.trinasolar.com/us/product/VERTEX-DEG21C.20.
17. V. Skiparev, J. Belikov and E. Petlenkov, "Reinforcement Learning based Approach for Virtual Inertia Control in Microgrids with Renewable Energy Sources," 2020 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe), 2020, pp. 1020-1024, doi: 10.1109/ISGT-Europe47291.2020.9248777.
18. Vallée, F., Lobry, J., & Deblecker, O. (n.d.). (PDF) System Reliability Assessment Method for wind power ... Retrieved December 9, 2021, from https://www.researchgate.net/publication/3268561_System_Reliability_Assessment_Method_for_Wind_Power_Integration.
19. Zakaria, A., Ismail, F. B., Lipu, M. S. H., & Hannan, M. A. (2019, July 16). Uncertainty models for stochastic optimization in renewable energy applications. Renewable Energy. Retrieved December 9, 2021, from https://www.sciencedirect.com/science/article/pii/S0960148119311012.