Title: Enhancing Graph Reinforcement Learning Techniques in Smart Grid Services

Author: António Oliveira

Supervision: António Costa

Co-Supervision: Rosaldo Rossetti

Date: December 18, 2023

1 List of Acronyms

ANN Artificial Neural Network

MLP Multilayer Perceptron

CNN Convolutional Neural Network

RL Reinforcement Learning

MDP Markov Decision Process

DRL Deep Reinforcement Learning

GNN Graph Neural Network

GRL Graph Reinforcement Learning

SAC Soft Actor-Critic

DDPG Deep Deterministic Policy Gradient

GCN Graph Convolutional Network

GAT Graph Attention Network

2 Background Knowledge

2.1 Artificial Neural Networks

Artificial Neural Networks (ANNs) are a class of machine learning algorithms based on the neural process of biological learning. The simplest form of an ANN is a Multilayer Perceptron (MLP), also called a feedforward network, whose main objective is to approximate to function f that models relationships between input data x and output data y of considerate

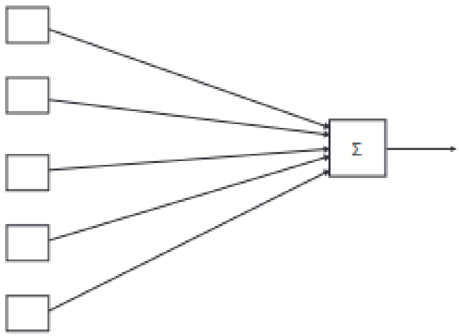

Figure 1: The Perceptron [1]

complexity. It defines a mapping $y = f(x;\theta)$ and learns the best composition of parameters θ to approximate it to the unknown model. The MLP serves as a fundamental part of developing the other discussed type of ANNs which is the Convolutional Neural Network (CNN). [1]

2.1.1 Feedforward Neural Networks

The main building block of a MLP is the *Perceptron*, a simple computational model initially designed as a binary classificator that mimics biological neurons' behaviour. A neuron might have many inputs x and has a single output y. It contains a vector of *weights* $w = (w_1 ... w_m)$, each associated with a single input, and a special weight b called the *bias*. In this context, a perceptron defines a computational operation formulated as equation 1 portrays. [1]

$$f(x) = \begin{cases} 1 \text{ if } b + \mathbf{w} \cdot \mathbf{x} > 0 \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

Functions that compute $b + \mathbf{w} \cdot \mathbf{x} > 0$ are called *linear units* and are identified with Σ (figure 1). An activation function g was introduced to enable the output of non-linear data, the default recommendation is the *Rectified Linear Unit* or *ReLU* and sigmoid is also another possibility.

Feedforward networks are composed of an input layer, formed of the vector of input values, an output layer which is the last layer of neurons, and an arbitrary number of hidden layers, the bigger the number the higher is the *depth* of the network.

On its own the model amounts only to a complex function, but with a real-world correspondence between input values and associated outputs, we can train a feedforward network to approximate the unknown function of the environment. In more concrete terms, this involves updating all of the different weight and bias values of each neuron to achieve an output as close as possible to the real or desired value, or minimize the total loss which indicates how distant is the network model to the real function to approximate to. *Loss functions* are used to calculate this value.

2.1.2 Convolutional Neural Networks

[1] [2] [3]

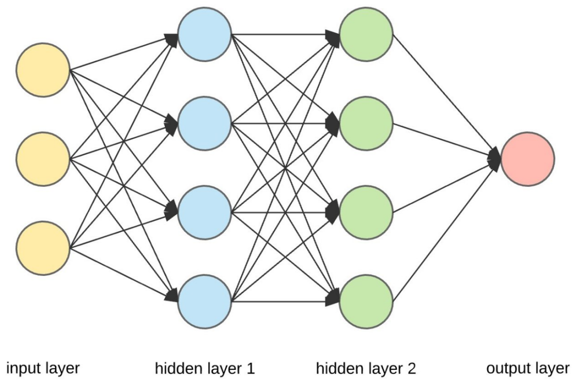

Figure 2: Feedforward Neural Network

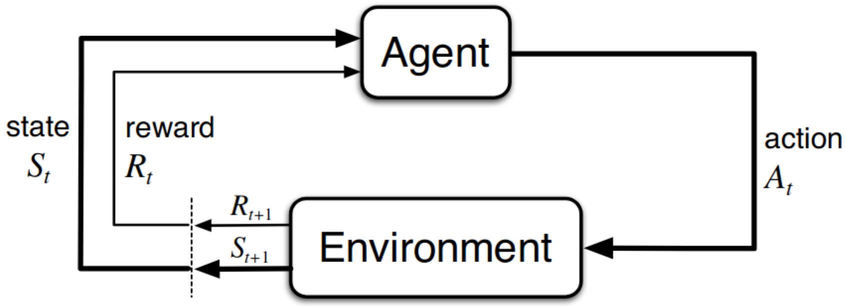

Figure 3: Interaction between agent and environment in an MDP

2.2 Reinforcement Learning

Reinforcement Learning (RL) consists of a category of machine learning methods to learn optimal action policies for achieving a specific goal considering the environment dynamics. In this context, an RL agent has the objective of autonomously discovering which courses of action yield the best cumulative reward, which is associated with the settled goal of the agent, by exploring the various decisions at their disposal in a trial-and-error fashion and considering potential delayed rewards. [4]

2.2.1 Markov Decision Process

Markov Decision Processs (MDPs) are a classical formalization of a sequential decision-making process, constituting the mathematical definition of the RL problem. In a MDP, an agent receives, in a particular time step t, a state, S_t, and a reward signal for the past action, R_t, from the environment it perceives. Furthermore, it chooses an action A_t considering the expected total cumulative reward signal in the long run, also called *expected return*, repeating the interaction sequentially until the system reaches a terminal state, that defines the end of an *episode*, or until it's stopped. Finite MDPs are processes which have finite sets of possible actions, states and reward signals, $\mathcal{A}$, $\mathcal{S}$ and $\mathcal{R}$, respectively. [4]

2.2.2 Deep Reinforcement Learning

[4] [5]

2.3 Graph Neural Networks

[6]

2.3.1 Graph Convolutional Network

[7]

2.3.2 Graph Attention Network

[8]

[9] [10] [11]

3 Related Works

Graph Reinforcement Learning (GRL) or Reinforcement Learning on Graphs is an area that is relatively new in the broader field of machine learning. GRL techniques have shown significant progress in solving problems with underlying graph-based representations such as power grid management [12] [13], smart transport [14] [15] or task offloading [16] [17].

Research on this topic has significantly increased in the last few years with the improvements of Deep Reinforcement Learning (DRL) techniques and the developments in Graph Neural Networks (GNNs) in the mid-2010s. GNNs became the state-of-the-art for solving numerous data mining tasks involving graph-structured data, excelling at classification, link prediction and representation learning [18] [19]. This advancement brought more sophisticated RL applications on graphs and the surge of a new field studying how to combine the improvements of graph mining and reinforcement learning techniques.

3.0.1 Graph Convolutional Networks

A common approach in Graph Reinforcement Learning model implementation is the use of Graph Convolutional Networks (GCNs) for leveraging graph-based structures to extract essential features of data in hand and improve the performance of RL agents in those environments.

[12] implements a GRL system to improve the decision quality of economic dispatch under high penetration of distributed energy generations. To accomplish this, a Soft Actor-Critic (SAC) system is employed with the main objective of finding the optimal action policy for minimizing generation cost with the appropriate reliability concerns. This problem is represented by an undirected graph with nodes describing the power grid elements with their respective attributes and edges describing the underlying energy connections between those units. To extract the structural features of the graph, this work implements a full connected layer to perform feature transformation with a two-layer GCN followed by three full connected layers for the non-linear mapping of state-to-action policy in both actor and critic modules. [20] develops a similar approach, with both concluding that it significantly reduces learning time for achieving better training results in comparison to plain SAC and showing significant improvement on economy and flexibility of the system on more complex and sparse state graphs. The use of GCNs enables the system to adapt to changes in the state space by leveraging the network's generalization ability. Another similar approach in a similar scenario with a Deep Q-Network algorithm is [21].

In [22] a three-layer GCN is used to extract node feature and graph topology information integrated into Rainbow-based [23] DRL algorithm for electric vehicle charging guidance. In this article, the testing results show promising performance in reducing operation costs for electric vehicle users, portraying a model with good generalization ability in untrained scenarios. [13] [24] [25]

3.0.2 Attention-based Mechanisms

Another effective approach in extracting relevant topology and graph features relies on using the Graph Attention Network (GAT) architecture instead of GCNs. [26] proposes a Deep Deterministic Policy Gradient (DDPG)-based algorithm improved with a GAT block with three graph Attention Layers for extracting and learning the topology information for dynamic economic energy dispatch. This paper compares the obtained test results against a GCN-DDPG model and shows increased performance over the GCN method in reducing cost and power loss. Beyond this, the work demonstrates that the GAT's attention mechanism enables the algorithm to focus on more important nodes and improve the signal-to-noise ratio compared to its GCN counterpart.

In [14], another model for the electric vehicle charging guidance is proposed, consisting of a bi-level approach of a Rainbow-based algorithm with a GAT block. The upper level focuses on the decision-making process regarding charging while the lower level handles routing. The proposed model proved to be more effective than a shortest distance path-based [27] and a DRL-based [28] approach. [29]

3.0.3 Other Approaches

[30] [31] [32]

References

[1] Eugene Charniak. *Introduction to Deep Learning.* The MIT Press, Cambridge, Massachusetts London, England, 2018.

[2] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning.* Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2016.

[3] Andrew W. Trask. *Grokking Deep Learning.* Manning, Shelter Island, 2019.

[4] Richard S. Sutton and Andrew Barto. *Reinforcement Learning: An Introduction.* Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, nachdruck edition, 2014.

[5] Miguel Morales. *Grokking Deep Reinforcement Learning.* Manning, Shelter Island [New York], 2020.

[6] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The Graph Neural Network Model. *IEEE Transactions on Neural Networks*, 20(1):61–80, January 2009.

[7] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks, February 2017.

[8] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph Attention Networks. In *International Conference on Learning Representations*, February 2018.

[9] Vinod Kumar and Dharmendra S. Rajput. *Concepts and Techniques of Graph Neural Networks*. 2023.

[10] Zhiyuan Liu and Jie Zhou. *Introduction to Graph Neural Networks*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Springer International Publishing, Cham, 2020.

[11] Lingfei Wu, Peng Cui, Jian Pei, and Liang Zhao, editors. *Graph Neural Networks: Foundations, Frontiers, and Applications*. Springer Nature Singapore, Singapore, 2022.

[12] Peng Li, Wenqi Huang, Zhen Dai, Jiaxuan Hou, Shang Cao, Jiayu Zhang, and Junbin Chen. A Novel Graph Reinforcement Learning Approach for Stochastic Dynamic Economic Dispatch under High Penetration of Renewable Energy. In *2022 4th Asia Energy and Electrical Engineering Symposium (AEEES)*, pages 498–503, March 2022.

[13] Xinpei Chen, Tao Yu, Zhenning Pan, Zihao Wang, and Shengchun Yang. Graph representation learning-based residential electricity behavior identification and energy management. *Protection and Control of Modern Power Systems*, 8(1):1–13, December 2023.

[14] Qiang Xing, Yan Xu, and Zhong Chen. A Bilevel Graph Reinforcement Learning Method for Electric Vehicle Fleet Charging Guidance. *IEEE Transactions on Smart Grid*, 14(4):3309–3312, July 2023.

[15] Paul Almasan, José Suárez-Varela, Krzysztof Rusek, Pere Barlet-Ros, and Albert Cabellos-Aparicio. Deep reinforcement learning meets graph neural networks: Exploring a routing optimization use case. *Computer Communications*, 196:184–194, December 2022.

[16] Zhen Gao, Lei Yang, and Yu Dai. Fast Adaptive Task Offloading and Resource Allocation in Large-Scale MEC Systems via Multi-Agent Graph Reinforcement Learning. *IEEE Internet of Things Journal*, pages 1–1, 2023.

[17] Nan Li, Alexandros Iosifidis, and Qi Zhang. Graph Reinforcement Learning-based CNN Inference Offloading in Dynamic Edge Computing. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, pages 982–987, December 2022.

[18] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful are Graph Neural Networks?, February 2019.

[19] Mingshuo Nie, Dongming Chen, and Dongqi Wang. Reinforcement Learning on Graphs: A Survey. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 7(4):1065–1082, August 2023.

[20] Junbin Chen, Tao Yu, Zhenning Pan, Mengyue Zhang, and Bairong Deng. A scalable graph reinforcement learning algorithm based stochastic dynamic dispatch of power system under high penetration of renewable energy. *International Journal of Electrical Power & Energy Systems*, 152:109212, October 2023.

[21] Tianqiao Zhao and Jianhui Wang. Learning Sequential Distribution System Restoration via Graph-Reinforcement Learning. *IEEE Transactions on Power Systems*, 37(2):1601–1611, March 2022.

[22] Qiang Xing, Yan Xu, Zhong Chen, Ziqi Zhang, and Zhao Shi. A Graph Reinforcement Learning-Based Decision-Making Platform for Real-Time Charging Navigation of Urban Electric Vehicles. *IEEE Transactions on Industrial Informatics*, 19(3):3284–3295, March 2023.

[23] Matteo Hessel, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining Improvements in Deep Reinforcement Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), April 2018.

[24] Dan Wang, Yalu Bai, Gang Huang, Bin Song, and F. Richard Yu. Cache-Aided MEC for IoT: Resource Allocation Using Deep Graph Reinforcement Learning. *IEEE Internet of Things Journal*, 10(13):11486–11496, July 2023.

[25] Xinwei Liu, Muchuan Qin, Yue He, Xiwei Mi, and Chengqing Yu. A new multi-data-driven spatiotemporal PM2.5 forecasting model based on an ensemble graph reinforcement learning convolutional network. *Atmospheric Pollution Research*, 12(10):101197, October 2021.

[26] Qiang Xing, Zhong Chen, Tian Zhang, Xu Li, and KeHui Sun. Real-time optimal scheduling for active distribution networks: A graph reinforcement learning method. *International Journal of Electrical Power & Energy Systems*, 145:108637, February 2023.

[27] Qiang Xing, Zhong Chen, Ziqi Zhang, Ruisheng Wang, and Tian Zhang. Modelling driving and charging behaviours of electric vehicles using a data-driven approach combined with behavioural economics theory. *Journal of Cleaner Production*, 324:129243, November 2021.

[28] Tao Qian, Chengcheng Shao, Xiuli Wang, and Mohammad Shahidehpour. Deep Reinforcement Learning for EV Charging Navigation by Coordinating Smart Grid and Intelligent Transportation System. *IEEE Transactions on Smart Grid*, 11(2):1714–1723, March 2020.

[29] Udesh Gunarathna, Renata Borovica-Gajic, Shanika Karunasekara, and Egemen Tanin. Solving Dynamic Graph Problems with Multi-Attention Deep Reinforcement Learning, January 2022.

[30] Yangzhou Pei, Jun Yang, Jundong Wang, Peidong Xu, Ting Zhou, and Fuzhang Wu. An emergency control strategy for undervoltage load shedding of power system: A graph deep reinforcement learning method. *IET Generation, Transmission & Distribution*, 17(9):2130–2141, May 2023.

[31] Jiechuan Jiang, Chen Dun, Tiejun Huang, and Zongqing Lu. Graph Convolutional Reinforcement Learning, February 2020.

[32] Fanfei Chen, John D. Martin, Yewei Huang, Jinkun Wang, and Brendan Englot. Autonomous Exploration Under Uncertainty via Deep Reinforcement Learning on Graphs. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6140–6147, October 2020.