A Latency-Aware Power-Efficient Reinforcement Learning Approach for Task Offloading in Multi-Access Edge Networks

Ali Aghasi and Rituraj Rituraj

November 16, 2022

A Latency-Aware Power-efficient Reinforcement Learning Approach for Task Offloading in Multi-Access Edge Networks

Ali Aghasi
Department of Computer Engineering
University of Isfahan
Azadi Sq, Isfahan
Iranaghasi@eng.ui.ac.ir

Rituraj Rituraj
Doctoral School of Applied Informatics and Applied Mathematics Obuda University
Budapest, Hungary
rituraj88@stud.uni-obuda.hu

***Abstract* – Since some cloud resources are located as edge servers near mobile devices, these devices can offload some of their tasks to those servers. This will accelerate the task execution to meet the increasing computing demands of mobile applications. Various approaches have been proposed to make offloading decisions about offloading. In this paper we present a Reinforcement Learning(RL) approach that considers delayed feedback from the environment, which is more realistic than conventional RL methods. The simulation results show that the proposed method succeeded to handle the random delayed feedback of the environment properly and enhanced the conventional reinforcement methods significantly.**

Keywords – Mobile-edge computing, Reinforcement Learning, Task offloading

I. INTRODUCTION

With the outstanding growth of the Internet of Things (IoT) and mobile applications in multi-access networks, enhancing the quality of user experience has attracted a lot of attention. Due to the limited resources of mobile devices, such as batteries, the execution latency must not exceed a specified deadline [1]. Assigning critical and compute-intensive tasks to the cloud may result in even longer response times owing to the propagation delay to the cloud data centers. Bringing these resources close to the network edge, reduces the transfer latency [2]. In this computing paradigm that is called Multi-Access Edge Computing (MEC), mobile devices can offload their compute-intensive tasks to the MEC servers to alleviate the network congestion problem and improve the application's response time [3]. Minimizing the computation latency is the major objective of task offloading through which, the energy consumption of mobile devices can be saved. Optimum decision-making is necessary for maximum use of the offloading mechanism. In this regard there are various strategies whose all aim summarizes in answering two following questions as stated by [4]. a) which tasks should be offloaded? b) where should they be offloaded? These approaches mainly rely in model-based or model-free methods. Formulating the objectives of the environment based on a simplified model, and tries to optimize them by existing optimization techniques, is the essence of model-based approaches whereas in model-free ones, a decision making agent tries to realize the best decisions by online interacting with the environment. The main model-free decision making approaches are based on reinforcement learning concepts. On the contrary, the model-based methods use a wide range of algorithms [5]. For instance, in [6] the nonlinear programming is applied to minimize the task completion time and power consumption. To simplify the model, authors did not consider uplink energy consumption. Authors of [7] was used Mixed Integer Programming (MIP) technique to perform the task scheduling and resource allocation in task offloading scenarios. Their model fails to be scalable because as the number of mobile devices increase, the task admission rate decreases. Optimal task offloading is inherently a NP hard problem [8] so the optimization techniques usually lead to a suboptimal solution of the model space.

In addition, the model simplification, which is necessary in resource limited devices, degrades the accuracy of analytical model-based methods. These drawbacks motivate the researchers to accept the challenges of machine learning techniques. As a branch of artificial intelligence, machine learning tries to learn from input data and use the knowledge to control and predict the system's behavior [9]. It is usually performed in three ways: Supervised Learning, Unsupervised Learning, and Reinforcement Learning (RL) [10]. Despite the other two paradigms, RL interacts online with the environment and uses the feedback to build its own knowledge base with no prior data. This feature makes RL interesting for decision making purposes. There are various studies in offloading decision systems that implements the RL paradigm in different shapes. In [11] authors, uses Q-learning method for computation delay and energy consumption minimization. Substituting deep neural networks as the state estimator with plain Q tables, introduces Deep Reinforcement Learning (DRL) which paves the way for applying RL in more sophisticated decision making applications. Huang et al. [12] used a deep reinforcement learning approach to minimize the average computation rate of all connected mobile devices. In [13] authors use DRL to allocate computation resources of edge server to offloaded task. One of the main challenges of applying RL approaches in real world scenarios which is almost neglected in studies, is the delayed feedback. Upon taking an action by the agent over the environment. The reaction of the environment is often accompanied by a delay that neglecting it can effectively reduce sample efficiency and prolong the convergence to optimal policy. In this paper we propose a SARSA mechanism to optimize computation latency with task offloading in multi access networks which, efficiently handle the delayed feedback problem.

I. Task offloading Architecture

In this paper we consider the case of single base station-multiple devices architecture that has been shown in Fig 1.

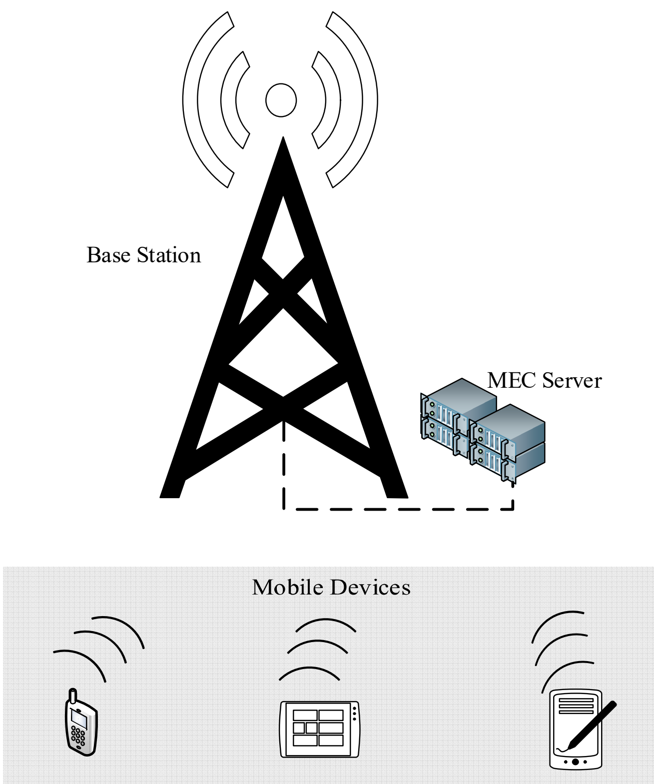

Fig. 1 Single base station Architecture

Extending the method for complex architectures is quite possible. In this Architecture, the mobile devices face three options to choose among for task computation; local execution, edge offloading and cloud migration.

II. Proposed Reinforcement Learning adaptation

We consider the agent as a task off loader operated on each device. At each decision epoch, the agent receives the state of the system ***S*** and issues an action ***A***. To reduce the overhead of task offloading computation, the state space should be kept as minimal as possible. To discretize the atate-space we assign a rank to any continuous interval. This assigning has been achieved empirically. The remaining power as an integer rank between one and ten (P_r), the computation complexity of the dispatched task as integer rank between one and 4 (C_t) and the remaining bandwidth of the channel as an integer rank between one and ten (B_r). table 1 shows the state variables of the environment. As it is mentioned before the action set includes 3 Members as follow.

A = {Local , Edge , Cloud}

The reward function has been designed to direct the agent toward optimal policy.

Table 1: State Space

State variable	**Explanation**	**Range**
P_r	Remaining Power of the Device (2mAh-14mAh)	1-10
C_t	Computational Complexity of the Task (50m clock cycle- 800m clock cycle)	1-4
B_r	Remaining Bandwidth (2mbps – 200mbps)	1-10
q_l	The Available Computation Capacity of the Device (200m clock cycle-1000m clock cycle)	1-4
q_e	The Available Computation Capacity of the MEC Server (2 cores - 64 cores)	1-4

The proposed function has considered computation latency, power consumption and bandwidth utilization. Depending on the taken action, the reward is calculated differently.

$$R=\begin{cases} -\text{(computation power)}*\text{(computation latency)}\ \textbf{if}\ A=\text{Local} \\ -\text{(transmission power)}*\text{(response time)}\ \textbf{if}\ A=\text{Edge, Cloud} \end{cases}$$

Response time is equal to the summation of computation latency on the remote device and transmission latency on uplink and downlink of the wireless channel.

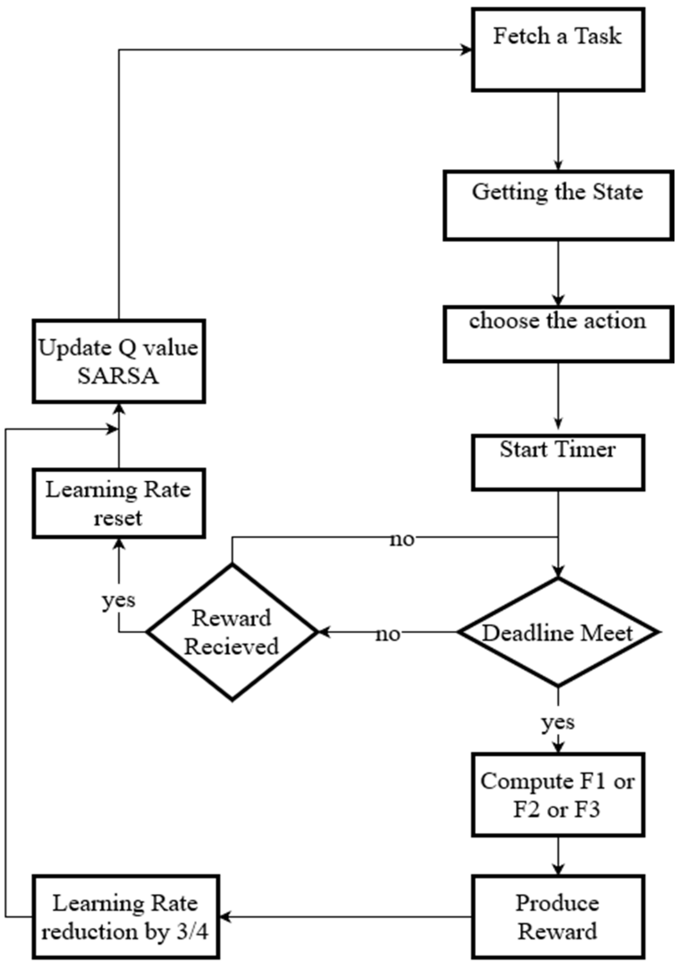

Fig. 2 The flowchart of the proposed mechanism

After receiving the reward, agent updates the Q-table by means of famous SARSA update equation. Using SARSA make the learning procedure more stable.

$$Q_{new}(s_t,a) = Q(s_t,a)+\propto [R + \gamma Q(s_{t+1},a^*) - Q(s_t,a)] \quad (1)$$

In real world scenarios like one discussed in this paper the feedback of reward (R). usually experiences a random delay which is needed to be considered and handled properly. We use a timer and computational models [14] as a side channel information for reward signal. The timer measure feedback delay of the reward and computational models calculate the transmission and computation power. If The timer reaches the deadline, the reward will be calculated by substituting the time elements with timer values and power elements with models F_1 and F_2. It worth mentioning that we use 3 different deadlines for offloading (T_1 , T_2 , T_3). Therefore, in the face of long delay the reward will be produced as:

$$R= \begin{cases} -(F_1) * (T_1) & \textbf{if } A= \text{Local} \\ -(F_2) * (T_2) & \textbf{if } A= \text{Edge} \\ -(F_2) * (T_3) & \textbf{if } A= \text{Cloud} \end{cases}$$

To mitigate the effect of inaccuracy of this method on the update procedure, the learning rate is reduced after meeting the deadline. The deadlines can be set based on the statistical observations. The flowchart of the mechanism is depicted in Fig 2.

III. Experimental Results

A simulation-based set-up has been utilized to evaluate the performance of the proposed method. The simulation parameters are listed in table 2

Table 2 simulation parameters

Element	Amount	Randomness
Num of mobile devices	[16,24]	Uniform distribution
Num of wireless channels	4	
Channel bandwidth	1.2GHz	
Device computation capacity	$[1.2,2.5] *10^9$ clock cycles/sec	Uniform distribution
MEC computation capacity	$32*10^9$ clock cycles/sec (64 cores)	

The proposed method has been compared against ordinary SARSA method and greedy heuristic method .The amount of average power consumption and computation latency reduction that each method achieved in comparison with threshold offloading scheme has been illustrated in Fig 3 and 4. The horizontal axis shows the methods, and the vertical axis shows the percentage of reduction. Each figure consists two parts. In the first part. The tree task offloading methods compared in respect to only local execution and only edge execution scheme.

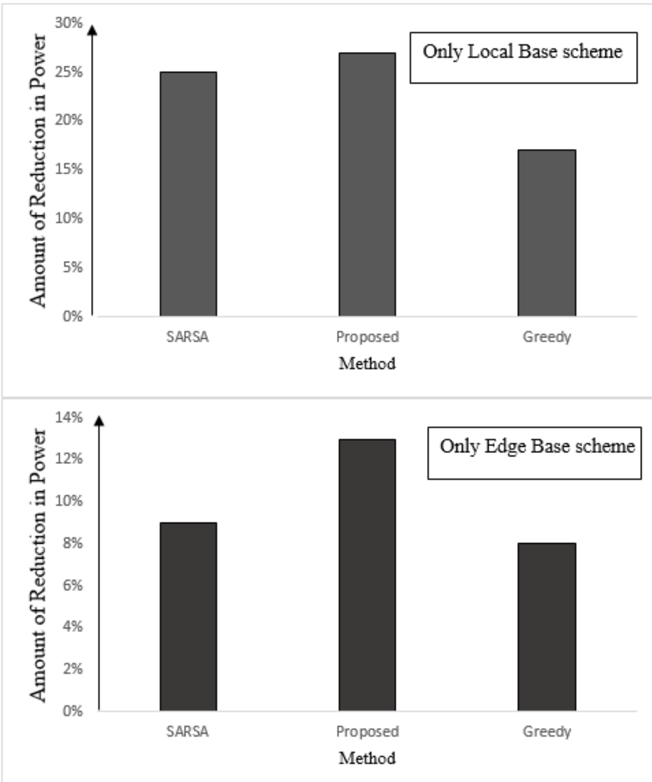

Fig. 3 the amount of power reduction compared with the base schemes

In only local execution scheme, the power consumption usually is high thus the effect of offloading policies is bold and noticeable. On the contrary, in only edge execution policy, the main power contributor which can be compromised by any approach is the transmission power. On the other hand, when it comes to computation latency metric, the only local execution scheme is less effected than edge only execution one.

In all modes the proposed method could outperform other methods. Convergence rate is one of the most important criteria in performance evaluation of RL algorithms. Fig 5 shows the convergence rate of the proposed method versus ordinary SARSA algorithm. Besides of the faster convergence to the optimal policy, the proposed method shows a more stable and smooth curve. The delayed feedback makes the credit assignment problem even worse. This problem is the source of misinterpretation in the relation of reward to optimal policy.

The performance of the reinforcement learning can indeed improve using further learning techniques proposed for various other applications, e.g., [15-25]. The optimal decisions can simultaneously benefit from soft computing and artificial intelligence techniques which are proven effective in a diverse range of applications, e.g., [26-36] which will be considered in our upcoming research.

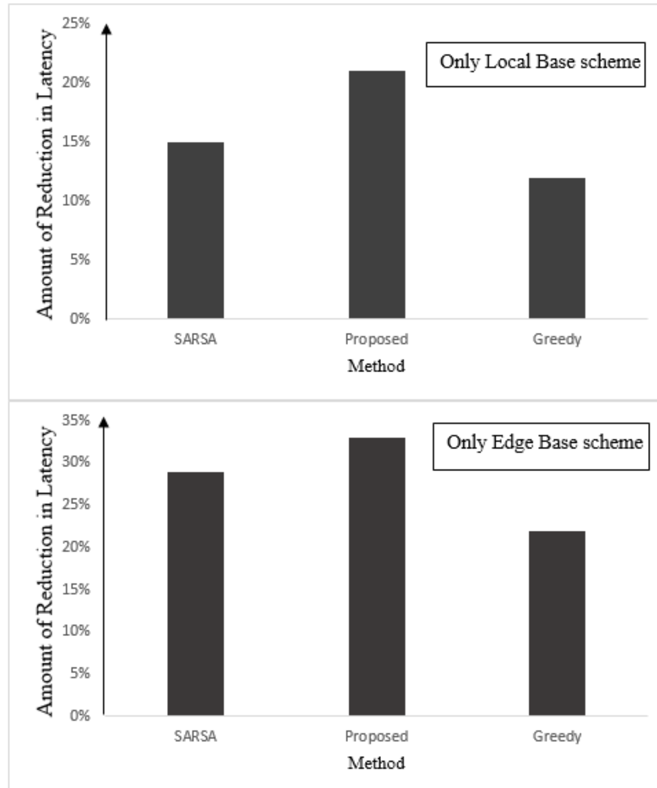

Fig. 4 the amount of computation latency reduction compared with the base schemes

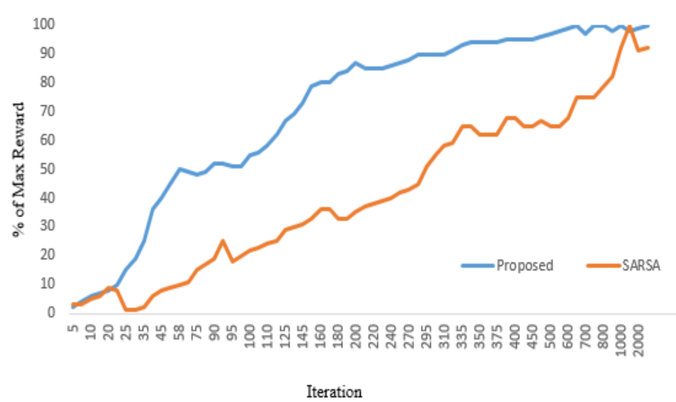

Fig. 5 convergence rate

For the future research, the application and the performance of ensemble and hybrid machine learning, such as those proposed in, e.g., [37-45], should be explored. It is essential to initiate comparative analysis and consider standard and advanced machine learning methods to come up with optimal model as proposed in several recent works, e.g., [46-52]. Literature suggests that often ensemble and hybrid machine learning outperform other artificial intelligence methods. Therefore, an in depth and focused research on these techniques is essential for future research.

IV. Conclusion

Task offloading is a nontrivial decision making problem that have a great impact in reliability and performance of new generations of wireless and IoT devices in multi-access networks. Reinforcement Learning is a promising approach to such a decision making problems which can act in absence of any model depending on the feedback of environment. The delay that comes with this feedback in most real environments creates problems in the process of reaching the optimal policy. In this paper we showed that proper handling of this issue can boost the performance reinforcement learning approaches. The results demonstrate well that the proposed method has not only made more optimal decisions, but also has acted faster in reaching the optimal policy than conventional SARSA algorithm.

References

[1] Rajasekar, Vani, J. Premalatha, K. Sathya, and Muzafer Saračević. "Secure remote user authentication scheme on health care, IoT and cloud applications: a multilayer systematic survey." *Acta Polytechnica Hungarica* 18, no. 3 (2021): 87-106.

[2] Y. Mao, C. You, J. Zhang, K. Huang, K.B. Letaief, A survey on mobile edge computing: the communication perspective, IEEE Commun. Surv. Tutor. 19 (2017) 2322–2358.

[3] C. Wu, Q. Peng, Y. Xia, Y. Ma, W. Zheng, H. Xie, S. Pang, F. Li, X. Fu, X. Li, et al., Online user allocation in mobile edge computing environments: A decentralized reactive approach, J. Syst. Archit. 113 (2021) 101904.

[4] M. Aazam, S. Zeadally, K.A. Harras, Offloading in fog computing for IoT:Review, enabling technologies, and research opportunities, Future Gener. Comput. Syst. (2018).

[5] Shen, Jing, Yongjie Li, Yong Zhang, Fanqin Zhou, Lei Feng, and Yang Yang. "A Survey on Task Offloading in Edge Computing for Smart Grid." In Proceedings of the 11th International Conference on Computer Engineering and Networks, pp. 13-20. Springer, Singapore, 2022..

[6] M. Chen, Y. Hao, Task offloading for mobile edge computing in software defined ultra-dense network, IEEE J. Sel. Areas Commun. 36 (3) (2018) 587–597.

[7] H.A. Alameddine, S. Sharafeddine, S. Sebbah, S. Ayoubi, C. Assi, Dynamic task offloading and scheduling for low-latency IoT services in multi-access edge

[8] Q.-V. Pham, L.B. Le, S.-H. Chung, W.-J. Hwang, Mobile edge computing with wireless backhaul: Joint task offloading and resource allocation, IEEE Access 7(2019) 16444–16459.

[9] Pejić, Aleksandar, and Piroska Stanić Molcer. "Predictive machine learning approach for complex problem solving process data mining." *Acta Polytechnica Hungarica* 18, no. 1 (2021): 45-63.

[10] A.C. Faul, A Concise Introduction to Machine Learning, CRC Press, 2019.

[11] J. Wang, K. Liu, M. Ni, J. Pan, Learning Based Mobility Management Under Uncertainties for Mobile Edge Computing, in: 2018 IEEE Global Communications Conference (GLOBECOM), IEEE, 2018, December, pp. 1–6.

[12] L. Huang, S. Bi, Y.J. Zhang, Deep Reinforcement Learning for Online
Computation Offloading in Wireless Powered Mobile-Edge Computing Networks, IEEE Trans. Mob. Comput. (2019).

[13] T. Yang, Y. Hu, M.C. Gursoy, A. Schmeink, R. Mathar, Deep reinforcement learning based resource allocation in low latency edge computing networks, in: 2018 15th International Symposium on Wireless Communication Systems (ISWCS), IEEE, 2018, August, pp. 1–5.

[14] You, Changsheng, Kaibin Huang, Hyukjin Chae, and Byoung-Hoon Kim. "Energy-efficient resource allocation for mobile-edge computation offloading." IEEE Transactions on Wireless Communications 16, no. 3 (2016): 1397-1411.

[15] Wang, H., et al., 2022. Comprehensive review of load forecasting with emphasis on intelligent computing approaches. Energy Reports, 8, pp.13189-13198.

[16] Pap, J., et al., 2022. Modeling Organizational Performance with Machine Learning. Journal of Open Innovation: Technology, Market, and Complexity, 8(4), p.177.

[17] Nejad, H.D., et al., 2022. Fuzzy State-Dependent Riccati Equation (FSDRE) Control of the Reverse Osmosis Desalination System With Photovoltaic Power Supply. IEEE Access, 10, pp.95585-95603.

[18] Pap, J., et al., 2022. Correlation Analysis of Factors Affecting Firm Performance and Employees Wellbeing: Application of Advanced Machine Learning Analysis. Algorithms, 15(9), p.300.

[19] Alanazi, A., et al., 2022. Determining Optimal Power Flow Solutions Using New Adaptive Gaussian TLBO Method. Applied Sciences, 12(16), p.7959.

[20] Aazami, R., et al., 2022. Optimal Control of an Energy-Storage System in a Microgrid for Reducing Wind-Power Fluctuations. Sustainability, 14(10), p.6183.

[21] Tavoosi, J., et al., 2022. A machine learning approach for active/reactive power control of grid-connected doubly-fed induction generators. Ain Shams Engineering Journal, 13(2), p.101564.

[22] Ardabili S., et al., Systematic Review of Deep Learning and Machine Learning for Building Energy, Frontiers in Energy Research, 10, 2022

[23] Akbari, E., et al., 2022. A Fault-Tolerant Cascaded Switched-Capacitor Multilevel Inverter for Domestic Applications in Smart Grids. IEEE Access.

[24] Rezaei, M.A., et al., 2022. A New Hybrid Cascaded Switched-Capacitor Reduced Switch Multilevel Inverter for Renewable Sources and Domestic Loads. IEEE Access, 10, pp.14157-14183.

[25] Iranmehr H., et al., Modeling the Price of Emergency Power Transmission Lines in the Reserve Market Due to the Influence of Renewable Energies, Frontiers in Energy Research, 9, 2022

[26] Band, S.S., et al., 2022. Feasibility of soft computing techniques for estimating the long-term mean monthly wind speed. Energy Reports, 8, pp.638-648.

[27] Shakibjoo, et al., A.H. and Vandevelde, L., 2021. Optimized Type-2 Fuzzy Frequency Control for Multi-Area Power Systems. IEEE access, 10, pp.6989-7002.

[28] Zhang, G., et al., 2021. Solar radiation estimation in different climates with meteorological variables using Bayesian model averaging and new soft computing models. Energy Reports, 7, pp.8973-8996.

[29] Cao, Y., et al., 2021. Deep learned recurrent type-3 fuzzy system: Application for renewable energy modeling/prediction. Energy Reports, 7, pp.8115-8127.

[30] Bavili, R.E., et al., 2021. A New Active Fault Tolerant Control System: Predictive Online Fault Estimation. IEEE Access, 9, pp.118461-118471.
[31] Tavoosi, J., et al., 2021. Modeling renewable energy systems by a self-evolving nonlinear consequent part recurrent type-2 fuzzy system for power prediction. Sustainability, 13(6), p.3301.
[32] Liu, Z., et al., 2021. A new online learned interval type-3 fuzzy control system for solar energy management systems. IEEE Access, 9, pp.10498-10508.
[33] Band, S.S., et al., 2022. When Smart Cities Get Smarter via Machine Learning: An In-depth Literature Review. IEEE Access.
[34] Bourouis, S., et al., 2022. Meta-Heuristic Algorithm-Tuned Neural Network for Breast Cancer Diagnosis Using Ultrasound Images. Frontiers in Oncology, 12, p.834028.
[35] Mosavi, A.H., et al., 2022. Deep learning fuzzy immersion and invariance control for type-I diabetes. Computers in Biology and Medicine, 149, p.105975.
[36] Almutairi, K., et al., 2022. A TLBO-Tuned Neural Processor for Predicting Heating Load in Residential Buildings. Sustainability, 14(10), p.5924.
[37] Ahmad, Z., et al., 2020. Machine learning modeling of aerobic biodegradation for azo dyes and hexavalent chromium. Mathematics, 8(6), p.913.
[38] Mosavi, A., et al., 2020. Machine learning for modeling the singular multi-pantograph equations. Entropy, 22(9), p.1041.
[39] Tavoosi, J., et al., 2022. A machine learning approach for active/reactive power control of grid-connected doubly-fed induction generators. Ain Shams Engineering Journal, 13(2), p.101564.
[40] Claywell, R., et al., 2020. Adaptive neuro-fuzzy inference system and a multilayer perceptron model trained with grey wolf optimizer for predicting solar diffuse fraction. Entropy, 22(11), p.1192.
[41] Ardabili, S., et al., 2019, September. Deep learning and machine learning in hydrological processes climate change and earth systems a systematic review. In International conference on global research and education (pp. 52-62). Springer, Cham.
[42] Torabi, M., et al., 2019. A Hybrid clustering and classification technique for forecasting short-term energy consumption. Environmental progress & sustainable energy, 38(1), pp.66-76.
[43] Rezakazemi, M., et al., 2019. ANFIS pattern for molecular membranes separation optimization. Journal of Molecular Liquids, 274, pp.470-476.
[44] Mosavi, A., et al., 2020. Comprehensive review of deep reinforcement learning methods and applications in economics. Mathematics, 8(10), p.1640.
[45] Ijadi Maghsoodi, A., et al., 2018. Renewable energy technology selection problem using integrated h-swara-multimoora approach. Sustainability, 10(12), p.4481.
[46] Samadianfard, S., et al., 2019. Support vector regression integrated with fruit fly optimization algorithm for river flow forecasting in Lake Urmia Basin. Water, 11(9), p.1934.
[47] Choubin, B., et al., 2019. Earth fissure hazard prediction using machine learning models. Environmental research, 179, p.108770.
[48] Mohammadzadeh S, D., et al., 2019. Prediction of compression index of fine-grained soils using a gene expression programming model. Infrastructures, 4(2), p.26.
[49] Shamshirband, S., et al., 2020. Prediction of significant wave height; comparison between nested grid numerical model, and machine learning models of artificial neural networks, extreme learning and support vector machines. Engineering Applications of Computational Fluid Mechanics, 14(1), pp.805-817.
[50] Ghalandari, M., et al., 2019. Flutter speed estimation using presented differential quadrature method formulation. Engineering Applications of Computational Fluid Mechanics, 13(1), pp.804-810.
[51] Kalbasi, R., et al., 2021. Finding the best station in Belgium to use residential-scale solar heating, one-year dynamic simulation with considering all system losses: economic analysis of using ETSW. Sustainable Energy Technologies and Assessments, 45, p.101097.
[52] Karballaeezadeh, N., et al., 2019. Prediction of remaining service life of pavement using an optimized support vector machine (case study of Semnan–Firuzkuh road). Engineering Applications of Computational Fluid Mechanics, 13(1), pp.188-198.