Optimal Scheduling Algorithm Using Machine Learning

[1]Varshini Adi, [2]Bhargavi Durseti,[3]Savio Soni, [4] Dr.T.Subha Mastan Rao
[1,2]*B.Tech Student, Department of Computer Science and Engineering, CMR Technical Campus, Medchal, Hyderabad, Telangana, India*
[3]*Associate Professor, Department of Computer Science and Engineering, CMR Technical Campus, Medchal, Hyderabad Telangana, India,*

Abstract-The Wireless rechargeable sensor networks have benefited immensely from mobile chargers (WRSNs). While most current research has been on on-demand recharging of WRSNs, little attention has been made to the combined consideration of residual energies and priorities when defining the charging schedule of energy-hungry nodes. Furthermore, most schemes ignore the issue of ill-timed charging responses to nodes with unequal energy consumption rates when making scheduling decisions, and they also ignore the issue of considering numerous network properties when making scheduling decisions. We address the aforementioned challenges in this study and present a unique scheduling strategy for WRSN on-demand charging. For establishing the charging schedule of the mobile chargers, we first use fuzzy logic, which integrates numerous network data. Next, we present a memorization-based method for determining the best charging schedule for mobile chargers. To illustrate the efficacy and efficiency of our approach, we run extensive simulations. The comparative findings show that the suggested scheme outperforms the current state-of-the-art methods in terms of variety of performance.

Keywords: Wireless Rechargeable Sensor Networks, Mobile Charger Vehicle, Reinforcement Learning.

1.INTRODUCTION

A wireless sensor network is a collection of sensor nodes that are used to continuously monitor and record physical or environmental changes as data for a variety of real-time applications. The internet of things refers to the interconnection of wireless sensor nodes assigned to do a specific task. The functioning period of the sensors is limited in most real-time scenarios due to the limits of their energy sources i.e., sensors nodes have very less battery capacity. As nodes have very less battery capacity recharging them is very difficult when the sensor nodes are turned on for longer hours. As a result, understanding the issues and developing solutions to improve the lifetime of the network is very important.

In real-world scenarios, the energy consumption of sensors varies based on their role and responsibilities. Every node in the sensor network has a different level of priority that means every node will have a different task to perform on different levels. So each node will consume a different amount of energy i.e., they consume a different amount of battery units for every unit of time.

As a result, problems such as data loss and sensor node premature death, as well as an energy- hole problem, arise in wireless sensor networks. These situations complicate the data transfer process to the base station and exacerbate the network partition problem. The network's lifespan is decreased because of the aforementioned concerns, and service quality frequently deteriorates.

Increasing the lifetime of the network in a wireless sensor network is a challenge. Because it involves a variety of things, such as sensor deployment, sensor clustering, network aggregation, optimal routing, and battery unit effectiveness. These strategies are implemented in a combined way to prolong the life of the network. However, because the number of working nodes changes over time in an asynchronous wireless sensor network, performing the tasks such as scheduling, and routing are challenging. Therefore, based on the asynchronous behavior of the IoT network, routing should be Automated to increase the performance of the network.

The authors devised a temporal-spatial charging scheme to determine the best charging path to minimize node deaths. However, the above-mentioned systems overlook multi- node charging and heterogeneous energy consumption of nodes, which should be taken into account because they diminish energy efficiency.

The authors created an efficient partitioning algorithm to partition the entire network into regions such that every region will get a rechargeable vehicle. In doing this the overall time taken to serve the sensor nodes will decrease and efficiency will increase not only that the average time taken will drastically come down when we split the region. It is in a way represents the divide and conquer methodology.

In the Nearest-Job-Next with Pre-emption (NJNP) method was presented, in which the node that is nearest to the MC is selected to be charged first. The MC, on the other side, gets pre-empted in the event of a new charging request from a closer node. In dense networks, too much MC pre-emption leads the nodes far away from the MC to die early, resulting in NJNP insufficiency.

A charging scheme was proposed that took into account both the requesting time of nodes and their distance from the MC, but it works the same as NJNP in the case of a large number of charging requests occurring at the same time, which is not ideal. The inventors attempted to optimize the covering utility by scheduling numerous MCs for energy delivery to the nodes. However, they neglected to consider other goals, such as increasing the survival rate while reducing charging latency provided an approximation approach for reducing the number of MCs required for recharging the nodes. Using the notion of the smallest enclosing disc, the authors came up with an approximation solution and found optimal sojourn locations for the MC. [10].

2.LITERATURE SURVEY

A. Tomar have researched that the ubiquitous wireless rechargeable sensor networks have been sustained by the development of mobile charging using wireless energy transfer technology. The research shows that employing multiple charging vehicles (MCVs) and a multi-node charging architecture, on-demand recharging of the sensor nodes (SNs) may greatly increase the charging performance. Most current schemes overlook the heterogeneous rate of SNs' energy consumption, partial charging of the SNs, and combined optimization of several network parameters, depriving these schemes of the advantages of extending network lifespan to the fullest degree.

Yong Jin, Jia Xu, Sixu Wu said, According to research on wireless sensor networks (WSNs), sensor nodes' (SNs) low battery capacity prevents continuous operation. In order to address the shortage of energy in WSNs, recent results in the field of wireless energy transfer (WET) have garnered a lot of interest from academics and industry. A new paradigm of wireless rechargeable sensor networks is created by WET, whose basic idea is to recharge SNs using one or more wireless mobile chargers (MCs) (WRSNs). A well-known NP-hard problem is choosing the best sequence in which to charge the SNs (also known as the charging schedule) in an on-demand WRSN. Additionally, because to the spatial and time limits introduced by requesting SNs, caution must be used while establishing an MC's charging schedule. The problem of scheduling an MC is initially formulated using Linear Programming (LP) in this study, and we then provide an effective solution using the gravitational search technique (GSA). An innovative agent representation strategy and an effective fitness function are proposed for our method. We do thorough simulations on the suggested scheme to show how well it performs in comparison to two cutting-edge algorithms, first come first serve (FCFS) and closest task next with preemption (NJNP). The simulation results show that in terms of charging delay, the suggested approach performs better than both of the currently used techniques.

P. Kar and S. Misra have researched Wireless charging vehicles (WCVs) may send electrical energy to sensors with the use of wireless power transfer technology, opening up a new paradigm for extending network lifetime. Existing collaborative charging systems often use a periodic and deterministic approach, ignoring the impacts of non-deterministic phenomena like topological changes and node failures, rendering them unsuitable for large-scale WRSNs. The on-demand charging architecture is developed for in this research using the temporal-spatial charging scheduling algorithm (TSCA). In order to extend network lifetime, we strive to reduce the proportion of dead nodes while optimising energy efficiency. First, a WCV will choose a workable movement option after compiling charging requests. The charging sequence is then modified to improve efficiency using a simple route planning algorithm. Additionally, global optimizations are made. After that, a node deletion method is created to get rid of charging nodes with low efficiency.

3.PROPOSED METHODOLOGY

We are proposing A wireless sensor network consists of heterogeneous sensor nodes (i.e., nodes that has different battery capacities as well have a different level of priorities), an immobile base station, and a mobile charger to recharge all the sensor nodes.

As sensory nodes have very limited battery capacities mobile chargers need to recharge these nodes very frequently. Each node has a rechargeable battery with an Emax total energy limit (i.e., battery capacity). We assume that the base station is aware of each node's location and that an event occurs with the same probability at each sensor node. Furthermore, throughout each data collecting round, the nodes continually sense the surroundings and report the perceived data to the base station. There are also no isolated nodes in the network. Each mobile charger is equipped with a large battery capacity such that it recharges every node that is present in the recharge schedule of the state I generated by the base station.

The following is the technique for how the system works. Because each sensor node has a finite battery capacity, when the battery level falls below a certain threshold, the sensor node sends a message packet to the base station requesting recharging. The message contains the node identification number, the node's residual battery (i.e., the battery that has been left out), and the priority.

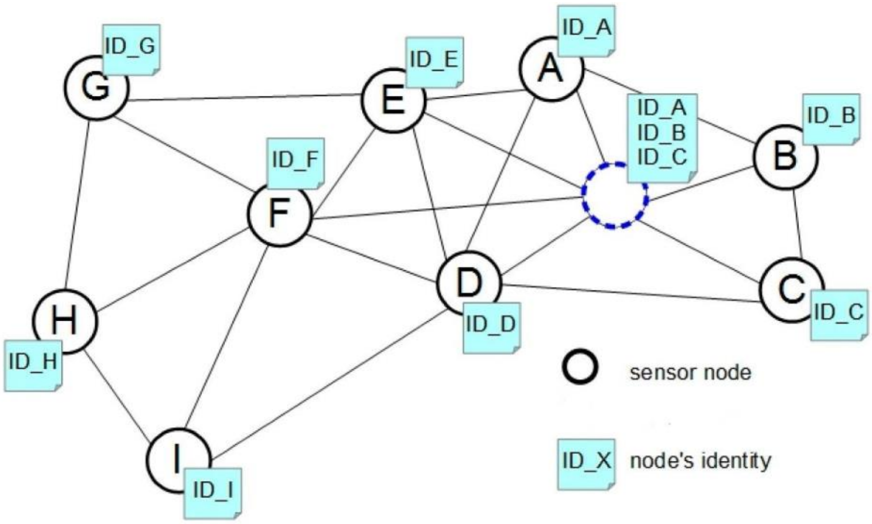
When the base station gets this message, it uses the Pythagoras formula to compute the distance between the node and the base station. The mobile charger will wait till the state time is through (i.e., each state has a certain time limit). After the time restriction has passed, a mobile charger with a charging schedule is sent to all of the request generated nodes to be recharged.

Schedule = {Base station → ti → ti+1 → · · · → tr} i.e., first rechargeable vehicle will perform will service the node ti then ti+1 and so on up to tr. And at last the total distance is travelled by the mobile charger is calculated and average waiting time of all the sensor nodes is calculated to check the optimality.

This problem is a minimization problem aiming at decreasing the average waiting time and total distance travelled by the charger vehicle. In this system, RL is used to solve the problem of scheduling. We could divide the system into states, actions.

So, reinforcement learning is the best fit to solve the scheduling problem. The entire system is divided into equal time intervals, in every time interval CMS collects the recharge requests. In our system these time intervals with requests are considered as states. Out of many schedules an optimal schedule will be given to MCV for recharging, this set of schedules in every state is action space. Action in state is a subset of all the schedules that are generated. Reinforcement learning agent cannot detect sensor nodes, but it is trained to act on the sensor node weights. RL agent acts for achieving long term reward. We are taking the reward function as an inverse of average waiting time and distance. We are presenting a fuzzy logic system for calculation of sensor node weights based on the properties like node distance, node recharge request time, remaining battery capacity.

As an on-policy version, in an optimal schedule the nodes are arranged in a sorted order based on the fuzzy node weights. To maximize the long-term reward, we must choose the best possible schedule out of all the schedules.



4.ALGORITHM

Code snippet to find node weights

```
for(int i=0;i<input.size();i++)
{
    for(int j=0;j<fuzzy_logic.size();j++)
    {
        if( fuzzy_logic[j][0] == input[i][1]  &&
            fuzzy_logic[j][1] == input[i][2]  &&
            fuzzy_logic[j][2] == input[i][3]   )
        {
            int vehicle_weight=j;
            input[i].push_back(vehicle_weight);
            break;
        }
    }
}
```

Fig1: Code snippet to find node weight

Algorithm to find permutations of a state

Input state consisting of requests of sensor nodes Data:

Generated Permutations: generated permutations,

Elements ToPermute: elements to permute (also initial permutation), Length OfArray: length of the array if LengthOfArray == 1 then

add Elements ToPermute to Generated Permutations; else

Length OfArray = LengthOfArray - 1;

Recursive HeapsAlgorithm (Elements ToPermute, LengthOfArray);

for i = 0; i < LengthOfArray; i++ do if

LengthOfArray is even then

SwapElements (i,Length OfArray);

else

SwapElements (0,Length OfArray); end

RecursiveHeapsAlgorithm (Generated Permutations, Elements ToPermute, Length OfArray); end

```
end
```

This algorithm works as decrease and conquer.

Algorithm to find the optimal schedule

```
While Sk is not the terminal state do
        Initialize N
      For every Ni do
                  Initialize Q (p, r, x_dist, y_dist, t, Nid)
        end for
         For every Qi do
     Find node weight and serial number using fuzzy logic table
        end for
           if efficient schedule is already present then
                output the schedule
       else if merging is possible then   output the
merged schedule
       else
  derive all the combinations of serial numbers.
           find optimal combination and store it in master table
       end if
      end while
```

Where,
S denotes state (consisting of set of recharge requests from sensor nodes).
P is priority of the sensor nodes
R is residual energy sensor node
N denotes no. of number of nodes in wireless rechargeable sensor network.
Q is request variable consisting of priority, Residual Energy, Distance, Request time, Node id of sensor nodes.
T is request time of sensor node.
Nid is node id of respective sensor node.

Algorithm to search whether optimal schedule is already present

```
vector<int>temp=schedule;
sort(temp.begin(),temp.end());
for(int i=0;i<optimal_schedules.size();i++)
{
    vector<int>var=optimal_schedules[i];
    sort(var.begin(),var.end());
    if(var==temp)
    {
        cout<<"schedule found already"<<endl;
        for(int j=0;j<schedule.size();j++)
        {
            cout<<"S"<<optimal_schedules[i][j]<<" ";
        }
        goto index;
    }

}
```

Fig: Code snippet to search existing optimal schedule

Here the input requests are sorted and for every iteration a vector form memorization table is considered, and it is sorted and compared with input requests, such that the overall computation time will be reduced.

Algorithm to find optimal schedule by merging two optimal schedules.

```
for(int i=0;i<optimal_schedules.size();i++)
{
    if( optimal_schedules[i].size() < temp.size())
    {
        for(int j=0;j<optimal_schedules.size();j++)
        {
            vector<int>join=optimal_schedules[i];
            for(int k=0;k<optimal_schedules[j].size();k++)
            join.push_back(optimal_schedules[j][k]);

            vector<int>orig=join;

            sort(join.begin(),join.end());
            if(temp==join)
            {
               cout<<"schedule found already"<<endl;
               for(int i=0;i<orig.size();i++)
               {
               cout<<"S"<<orig[i]<<" ";
               }
               goto index;

            }

        }
    }
}
```

Fig: Code snippet to find optimal schedule by merging two schedules

5.RESULTS

The images of the output of the developed model are shown below:

Case 1:

```
Enter No of nodes Requesting for a Recharge
2
Enter Priority , Residual Energy , X-cord , Y-cord , Request_time, Node_Number
3 3 1 2 10 20

3 3 9 9 1 17

1 3 3 3 10 1 2 0
2 3 3 2 1 9 9 1


S0 S1


S1 S0



The schedule with optimal waiting time and less distance is
S0 S1
```

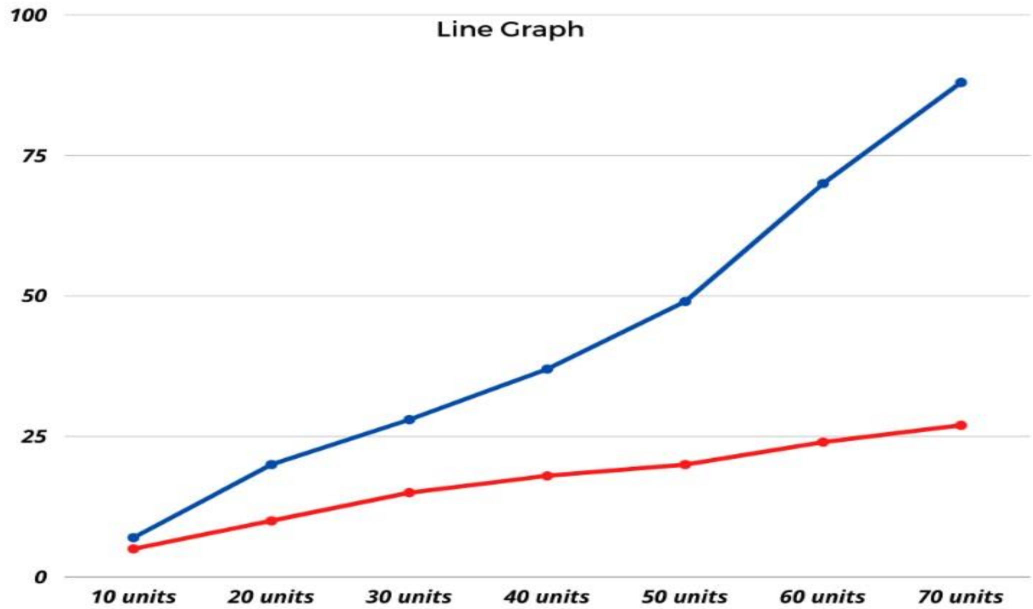
Case 2:

```
Wish to continue
1
Enter No of nodes Requesting for a Recharge
4
Enter Priority , Residual Energy , X-cord , Y-cord , Request_time, Node_Number
3 3 1 2 10 20

3 3 9 9 1 17

3 3 21 21 11 45

3 2 1 2 30 14

1 3 3 3 10 1 2 0
2 3 3 2 1 9 9 1
3 3 3 1 11 21 21 2
4 3 2 3 30 1 2 3


S0 S1 S2 S3 schedule found already
S0 S1 S3 S2 Wish to continue
```

Comparison with shortest distance first algorithm

X-Axis: Max node distances.

Y-Axis: Total distance covered by a rechargeable vehicle to recharge all the nodes in a schedule.



Distance covered comparison graph

- Proposed Memorization based algorithm
- Shortest Distance first algorithm

Distance Comparison of proposed FRL with FCFS, SDF, Fuzzy algorithm

Comparison of proposed algorithm with SJF, Fuzzy, FCFS with respect to distance

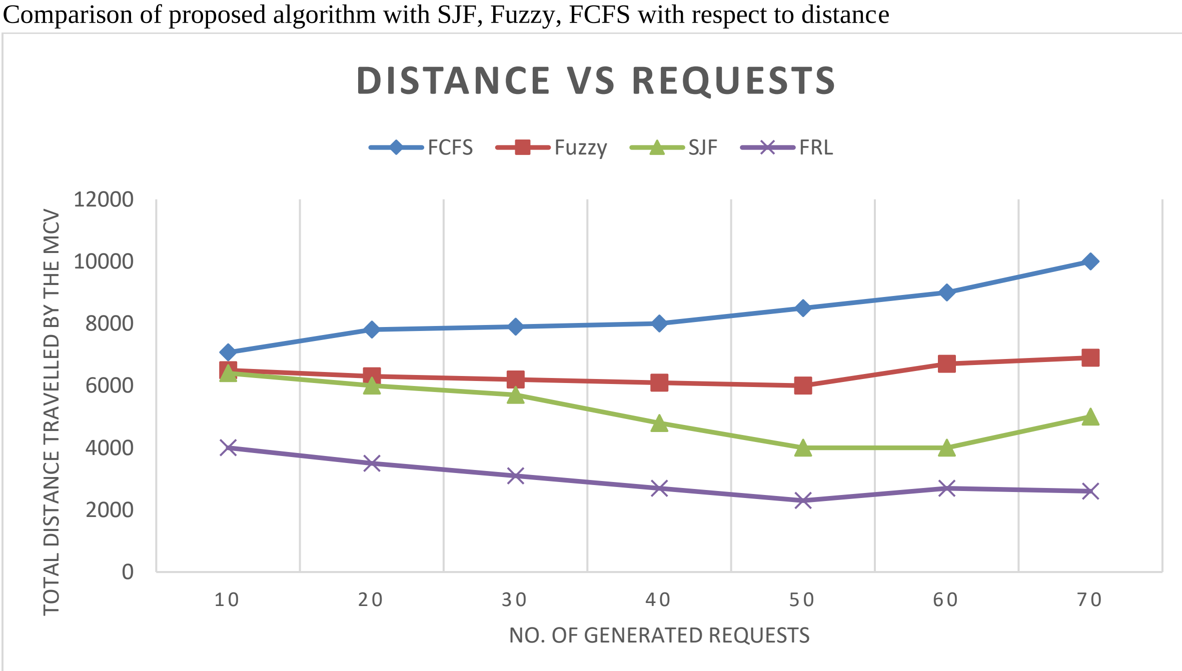


Fig : Distance Comparison of Proposed FRL algorithm with FCFS, SDF, FUZZY

6.CONCLUSION

Research and experiment have been conducted to detect the optimal schedule with help of fuzzy logic and reinforcement learning. The experiment to detect optimal schedule has been done according to the research methodology that has been proposed according to the objective to the research. The research questions that have been taken here will be discussed in this section along with the discussion of the present limitations and future scopes of the project.

7.FUTURE ENHANCEMENTS

We want to combine our memorization-based approach with reinforcement learning in the future. Reinforcement Learning is a deep learning approach for optimizing overall reward of the system. This long-term learning approach uses neural networks to optimize a particular functionality of the system. We could obtain a much more optimal schedule this way. The scheduling can be done using multiple inputs in order to support real time sensor network systems. In future, multiple mobile chargers can be used to recharge the sensor nodes in the networks.

8.ACKNOWLEDGEMENT

We thank CMR Technical Campus for supporting this paper entitled "A MOBILE CHARGER SCHEDULING ALGORITHM FOR RECHARGEABLE WIRELESS SENSOR NETWORK USING MACHINE LEARNING", which provided good facilities and support to accomplish our work. We sincerely thank our Chairman, Director, Deans, Head of the Department, Department of Computer Science and Engineering, Guide and Teaching and Non-Teaching faculty members for giving valuable suggestions and guidance in every aspect of our work

REFERENCE

[1].A. Tomar, L. Muduli and P. K. Jana, "A Fuzzy Logic-Based On-Demand Charging Algorithm for Wireless Rechargeable Sensor Networks With Multiple Chargers," in IEEE Transactions on Mobile Computing, vol. 20, no. 9, pp. 2715-2727, 1 Sept. 2021, doi: 10.1109/TMC.2020.2990419.

[2]. Yong Jin, Jia Xu, Sixu Wu, Lijie Xu, Dejun Yang, Kaijian Xia, Bus network assisted drone scheduling for sustainable charging of wireless rechargeable sensor network, Journal of Systems Architecture, Volume 116,2021,102059,ISSN 1383-7621,https://doi.org/10.1016/j.sysarc.2021.102059.

[3] P. Kar and S. Misra, "Reliable and efficient data acquisition in wireless sensor networks in the presence of trans faulty nodes," IEEE Transactions on Network and Service Management, vol. 13, no. 1,pp. 99–112, 2016.

[4] L. Fu, P. Cheng, Y. Gu, J. Chen and T. He, "Optimal Charging in Wireless Rechargeable Sensor Networks," in IEEE Transactions on Vehicular Technology, vol. 65, no. 1, pp. 278-291, Jan. 2016, doi: 10.1109/TVT.2015.2391119.

[5] Chi Lin, Zhiyuan Wang, Ding Han, Youkun Wu, Chang Wu Yu, Guowei Wu, TADP: Enabling temporal and distantial priority scheduling for on-demand charging architecture in wireless rechargeable sensor Networks, Journal of Systems Architecture.

[6] C. Lin, J. Zhou, C. Guo, H. Song, G. Wu and M. S. Obaidat, "TSCA: A Temporal-Spatial Real-Time Charging Scheduling Algorithm for On-Demand Architecture in Wireless Rechargeable Sensor Networks," in IEEE Transactions on Mobile Computing, vol. 17, no. 1, pp. 211-224, 1 Jan. 2018, doi: 10.1109/TMC.2017.2703094.

[7] L. He, L. Kong, Y. Gu, J. Pan and T. Zhu, "Evaluating the On-Demand Mobile Charging in Wireless Sensor Networks," in IEEE Transactions on Mobile Computing, vol. 14, no. 9, pp. 1861-1875, 1 Sept. 2015, doi: 10.1109/TMC.2014.2368557.

[8] B. Fateh and M. Govindarasu, "Joint Scheduling of Tasks and Messages for Energy Minimization in Interference-Aware Real-Time Sensor Networks," in IEEE Transactions on Mobile Computing, vol. 14, no. 1, pp. 86-98, Jan. 2015, doi: 10.1109/TMC.2013.81.

[9] @article{10.1145/2998179,author = {Bandari, Maryam and Simon, Robert and Aydin, Hakan},title = {DMS-Based Energy Optimizations for Clustered WSNs},year = {2017}, issue_date = {August 2017}, publisher = {Association for Computing Machinery}, address = {New York, NY, USA}, url = {https://doi.org/10.1145/2998179},

[10] A. Kaswan, A. Tomar, and P. K. Jana, "An efficient scheduling scheme for mobile charger in on-demand wireless rechargeable sensor networks," Journal of Network and Computer Applications, vol.114, pp. 123–134, 2018.

[11] B. Jang, M. Kim, G. Harerimana and J. W. Kim,

"Q-Learning Algorithms: A Comprehensive Classification and Applications," in IEEE Access, vol. 7, pp. 133653-133667, 2019, doi: 10.1109/ACCESS.2019.2941229.
[12] W. Qiang and Z. Zhongli, "Reinforcement learning model, algorithms and its application," 2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC), 2011, pp. 1143-1146, doi: 10.1109/MEC.2011.6025669.
[13] I. Aleo, P. Arena and L. Patané, "SARSA-based reinforcement learning for motion planning in serial manipulators," The 2010 International Joint Conference on Neural Networks (IJCNN), 2010, pp. 1-6, doi: 10.1109/IJCNN.2010.5596282.
[14] M. A. Samsuden, N. M. Diah and N. A. Rahman, "A Review Paper on Implementing Reinforcement Learning Technique in Optimising Games Performance," 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET), 2019, pp. 258-263, doi: 10.1109/ICSEngT.2019.8906400.
[15] K. Kim, "Enhancing Reinforcement Learning Performance in Delayed Reward System Using DQN and Heuristics," in IEEE Access, vol. 10, pp. 50641-50650, 2022, doi: 10.1109/ACCESS.2022.3174361.
[16] J. Perdiz, L. Garrote, G. Pires and U. J. Nunes, "A Reinforcement Learning Assisted Eye-Driven Computer Game Employing a Decision Tree-Based Approach and CNN Classification," in IEEE Access, vol. 9, pp. 46011-46021, 2021, doi: 10.1109/ACCESS.2021.3068055.
[17] J. Perdiz, L. Garrote, G. Pires and U. J. Nunes, "Measuring the impact of reinforcement learning on an electrooculography-only computer game," 2018 IEEE 6th International Conference on Serious Games and Applications for Health (SeGAH), 2018, pp. 1-8, doi: 10.1109/SeGAH.2018.8401359.