ISSN 2278-3091
Volume 12, No.1, January - February 2023
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse061212023.pdf
https://doi.org/10.30534/ijatcse/2023/061212023

Multi-agent Reinforcement Learning Approach to Enhance Proactive and Resilience System based IoT

Nourelhouda FEREHAN[1], Abdelkrim HAQIQ[2], Abdelaziz Fezziki[3]
[1]Hassan First University, Faculty of Sciences and Techniques, Computer, Networks, Mobility and Modeling laboratory: IR2M, Settat, Morocco, n.ferehan@uhp.ac.ma
[2]Hassan First University, Faculty of Sciences and Techniques, Computer, Networks, Mobility and Modeling laboratory: IR2M, Settat, Morocco, a.haqiq@uhp.ac.ma
[3]Cadi Ayyad University, Faculty of Sciences Semlalia, Computing Systems Engineering Laboratory: ISI, Marrakech, Morocco, a.elfezziki@uca.ac.ma

Received Date December 18, 2022 Accepted Date: January 26, 2023 Published Date: February 06, 2023

ABSTRACT

Internet of Things (IoT) systems that use reinforcement learning and multi-agent system are described in this research. Intelligent agents reside on computers that comprise this system. They enable a system to take independent actions, communicate with other systems, and adapt to changing situations. We presented a system that uses intelligent agents embedded in machines to determine which procedures are most critical and how they should be distributed. Robots with intelligent agents can improve their decision-making abilities. The proposed system and transmitting rule function compare the scheduling problem with early completion, efficiency, and delay in checking the system and the dispatching. Multi-agent using resilience systems are competitive even in a continually developing environment. It is possible that reinforcement learning with intelligence agents will be used in the future because it provides users with a unique approach to problem-solving. Additionally, these methodologies have new ways to optimize complex systems in scheduling, project planning, and other business-related domains when used in conjunction with the Internet of Things (IoT) standard technology. The rest of the paper is organized in give section, after providing the introduction, in 2nd section previous work has been discussed, in the third section, the methodology is discussed, in the 4th dataset and result and discussion is explain in the last work is concluded.

Key words: Multi-Agent, Reinforcement learning, Enhance proactive, Resilience system, and IoT.

1. INTRODUCTION

For this work, the major goal is a comprehensive analysis of Reinforcement learning methods from a multi-agent system based on the Internet of Things (IoT). These parameters can only be determined with a thorough understanding of Reinforcement methods. New techniques to optimize complex systems in various industries, including logistics and scheduling, have been made possible by combining these methods and techniques typically used for the Internet of Things. Removing this hidden potential may result in a paradigm [1]. We may expect vertical and horizontal integration and flexibility development, and the key focus should be on human supervision during this digital transformation process [2] [3]. Reinforcement-based learning is an excellent method for developing self-organizing and self-improvement multi-agent systems. For the simple reason that it offers a new approach to the development of machine learning, reinforcement learning has the potential to revolutionize the way intelligence agents is used in the near future.

It is possible to acquire cooperative behavior without full communication between the controllers using data-driven control schemes based on multi-agent reinforcement learning (RL) with the correct training process and reward function. Reinforcement learning-based multi-agent generation control is recommended in [4][11] for better control coordination in this case because the ideal actions for all agents are divided into distinct phases. Control performance is constrained by the discretization of actions in traditional Reinforcement learning-based

To ensure that limitations are maintained when more than one line is down, an offline training of the multi-agent framework is carried out to determine where and how large shunts are needed. The training method makes use of historical data and transmission line insubstantiality curves. It's regularly updated to reflect any adjustments to the system's parameters. It is used at the end of each training period to measure the effectiveness of a reward function for completing the tasks. An agent's actions during a training session are recorded in a replay buffer, with their rewards and the system current state. Regularly, the multi-agent framework settings are changed using a random minibatch. Because of the multi-agent system built into the function of getting the agents can learn from their mistakes and avoid participating in bad behavior in the future. When properly instructed, they will eventually learn how to

conduct themselves properly [5] System administrators might first validate judgments made by the agents to reduce the risk.

2. LITERATURE REVIEW

An event-driven dynamic that is conscious of the context in which it operates [6] presents an approach for managing Internet of Things services that is compatible with many different types of physical things and integrates them into a more general setting. An Internet of Things infrastructure that is aware of the current situation and dynamic, event-driven service coordination that is able to react to shifts in the surrounding environment. The best elements of service-oriented architecture (SOA) and enterprise application architecture (EDA) are combined in a single solution (event-driven architecture). The integration of a situational event pattern and a situational event-driven service coordination behaviour model is accomplished through the utilisation of an event condition-action trigger mechanism.

A goal-driven service orchestration technique is illustrated in [7], and it can be applied to IoT situations. It is able to personalise the service procedures in order to accommodate the requirements of the user as well as the circumstances in which they are being utilised. As an orchestrator, a RESTful and semantic metadata specification for Internet of Things services is being created. [7] provides an overview of the semantic metadata definitions for Internet of Things services. The combination of orchestration and choreography is used to develop a MQTT client that can be updated from a location that is geographically dispersed. The findings of this article lead the authors to the conclusion that both designs ought to be utilised, given that each one enhances the capabilities of the other. It has been found that a combination of orchestration and choreography is superior to either one used alone because it enables the creation of new interactions that are significant. This is the main reason for this observation.

(Li et al.,2021) proposes an entirely model-free multi-agent real-time deep reinforcement learning method with parameterized double deep Q-networks that can capture a hybrid policy of discrete and continuous actions. Furthermore, a thorough comparison of several multi-agent reinforcement learning and model-based optimization approaches is made to demonstrate that the proposed method works better. Much attention has been paid to underwater optical wireless sensor networks with high transmission rates and low delays for underwater applications that require a lot of bandwidth. Underwater optical communication nodes have numerous challenges due to the constantly shifting topography created by ocean currents. To solve this issue and improve the stability of dynamic networks, we provide a new routing protocol for underwater optical sensor networks based on multi-agent reinforcement learning (MARL)[1]. Before any route selection can be made dynamically based on the information flowing between nodes, the network is first represented as a multi-agent system using a reinforcement learning technique. Simulated results reveal that MARL consumes less energy in dynamic networks and has a higher delivery rate (about 95 %) than conventional routing methods.

Innovative manufacturing systems are becoming more and more intelligent with the use of advanced cognitive computing. As a result, a Self-X network with a higher level of automation is being developed. According to our knowledge, the "Self-X" levels are still unavailable. The proposed framework [8] network is made possible by introducing an IKG-based multi-agent reinforcement learning (MARL). Starting with empirical knowledge and patterns discovered throughout production, an IKG should be created. Data from both humans and machines can be used for this purpose. A graph neural network embedding approach [8] has been proposed to enable semantic-based self-configurable solution searches and problem decompositions. Thorough knowledge of the established IKG is required for this. It is shown that the proposed method can be applied to a multi-robot reaching job to demonstrate its viability. An exploratory study also considers what could go wrong and what could happen in the future [8]. This is done to promote more honest dialogue and in-depth investigation to improve industrial efficiency even further.

As part of the European Project ATENA, [8] demonstrates a Cyber-Physical System (CPS) defense control method (CI). The controller's role is to choose the best countermeasures to utilize in order to address the system's flaws. General sum games represent the attack/defense problem, where the defender's goal is to minimize damage by determining the optimum trade-off between actions to reduce damage and costs of those actions [9]. The problem is solved via reinforcement. Finding the zero-sum Nash equilibrium and putting the attacker in a position where the damage can inflict on the infrastructure is less than what it costs to carry out their attack strategy are examples of learning and simulation results that demonstrate how the proposed concept can be applied to minimize the damage done by an adversary to the protected critical infrastructure (CI).

Recently, (Anuar et al.,2013) AI and ML have grown increasingly popular in various fields, including the oil and gas industry (OGI), where they help people perform challenging tasks. In addition, multi-agent systems (MAS) are a subset of distributed AI that addresses the needs of distributed systems and has found application in various industries. ML and MAS have been looked at in several studies as strategies to improve operational efficiency, manage the supply chain, and handle difficulties in production and maintenance in the (Anuar et al.,2013). OGI. But ML has only been employed for a few jobs, and MAS has performed well in simulated scenarios. Research in the disciplines of ML and MAS needs to be done for the OGI to accept them to their full potential. For example,

incorporating machine learning (ML) into MAS can aid the industry in various ways.

[4] discussed the reinforcement-based learning multi-agent to distribute the energy in the household. In order to control the flexibility provided by electric devices and space heaters loads in an environment that is only partially observable, agents work together. The traditional independent Q-learning strategy reduces the ability of agents to work together in unpredictable situations. Learning offline convex optimizations on previous data and deriving marginal reward contributions to total rewards enhances stability and large-scale performance in this unique combination. There are several ways prosumers could use fixed-size Q-tables without exchanging any personal information with each other or with a central coordinator. [4] the contribution shows that existing IDPS schemes are reviewed and categorized using classical artificial computational intelligence with multi-agent assistance. In this study, the significance of the methodologies and procedures, as well as their performance and limitations, are examined. The restrictions are viewed as obstacles to building a wireless IDPS architecture based on collaboration (Co-WIDPS) and are treated as such. Fog learning and knowledge management are used to produce a superior technical platform that can detect threats with more precision. In the end, we discuss a few major future research issues that could help speed up the development and implementation of Co-WIDPSs based on computational intelligence.

.

3. METHODOLOGY

A computer model known as an artificial neural network (ANN) has processing units that receive input and output data based on its activation functions. To begin, the network should be taught to look for patterns in the input data that are not immediately apparent, such as the weights of the connections between nodes in the network. A similar dataset might be used to make predictions about its behavior [11]. The proposed strategy aids agents in anticipating the next move of the opposite party before making an offer. In this way, they are able to conduct what-if studies. It is conceivable to utilize an artificial neural network (ANN) in automated negotiation to determine whether or not an agreement can be reached. A neural network (ANN) was used by [9] to discover a correlation between a supplier's previous bids and its current bid price.. When it comes to negotiating, this helps both sides save time by allowing them to make more informed decisions regarding their next proposal. An agent based framework architecture for IoT is shown in Figure 1.

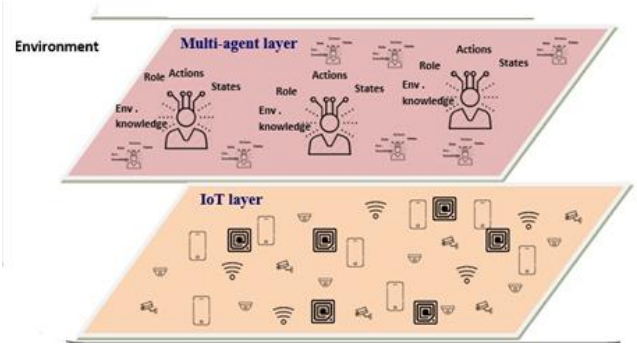

Figure 1: An agent based framework architecture for IoT

There are various drawbacks to using an artificial neural network (ANN) to recognize patterns and forecast future events. Real-world negotiations do not always provide enough information to use this technique of training effectively. Another issue is the overuse of fitting. Excessive training results in an overfitted model that can no longer learn from new data. Instead of identifying data patterns, the model detects noise and errors [1]. A lack of generalizability means that the model cannot accurately anticipate what will happen in other datasets. As a result, it cannot be employed in dynamic negotiation circumstances because it is developed during the training phase and cannot be altered subsequently. Real-world negotiations often include a time constraint, which ANN cannot handle.

3.1 ANN Algorithm

1. Using a random weight for each link, the process is started.
2. In order to discover the hidden node activation rate, it uses the inputs and the links between the inputs
3. The rate of activation of output nodes can be calculated from the activation rates of hidden nodes and linkages to output.
4. As a last step, re-calibrate all hidden-to-output node linkages based on the new error rate.To communicate the error to the hidden nodes, use the weights and error at the output node.
5. The weight of the hidden node can be adjusted relative to the nodes being fed into the network.
6. Do the process again and again until the convergence criteria are met.
7. The activation rate of the output nodes is given a score based on the last linkage weights.

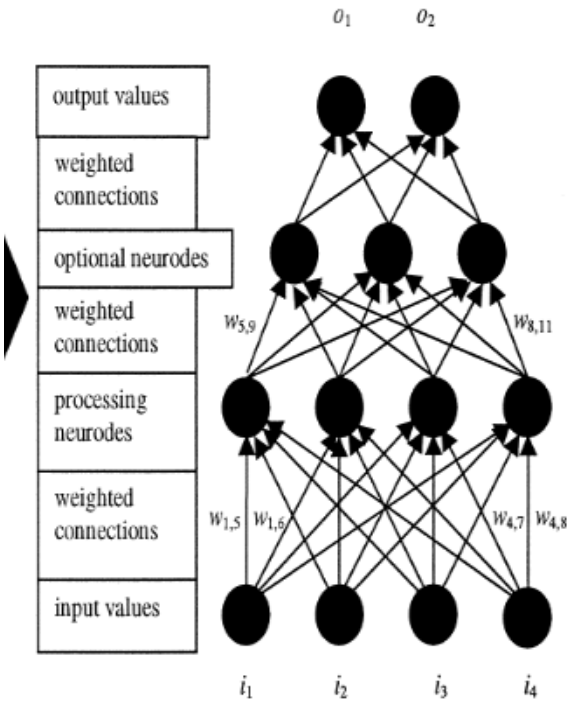

Figure 2: Architecture of Artificial neural Network

Figure 2 shows the Architecture of Artificial neural Network. Reinforcement learning is a technique in which computer programmers devise a means to encourage good behavior while punishing it. If you want the agent to accomplish something, assign it a positive value; if you do not give it a negative value. By doing so, the agent is tasked with locating the most effective long-term and comprehensive solution.
In order to avoid being bogged down in short-term aims, an agent must have long-term objectives. Over time, the agent develops the ability to prevent evil and seek out good. With the use of incentives and penalties, AI systems may learn independently without human supervision.

3.2 Simulation Framework& Dataset

Using an agent-based simulation model (ABSM) to solve IoT issues is widely accepted. As a robust simulation model, it is capable of solving these issues and helping the IoT environment thrive. ABSM and cloud architecture could be employed jointly in an IoT scenario, according to Fortino and Guerrieri [2] Distributed multi-agent systems benefitted from ABSM and the cloud architecture, which provided significant computational power and a large amount of storage space for IoT devices. New ABSM can mimic IoT using the most critical aspects of an IoT environment. This team devised a strategy to demonstrate the random nature of IoT occurrences. ABSM can be used to validate IoT systems, as the simulated findings were quite comparable to those in the actual world. ABSM agents' architecture was utilized to describe how a single agent might make decisions during an evacuation. In a real-world setting, their work was applied to a building with IoT sensors. IoT settings are shown in a new way by [8].. The suggested method simulates complicated IoT systems using the cognitive ABSM framework. Many complex network topologies, such as random and scale-free networks, have been shown to exist. A case study was utilized to demonstrate how well ABSM can simulate connected IoT devices. The below figure 3 shows the Simulation Framework.

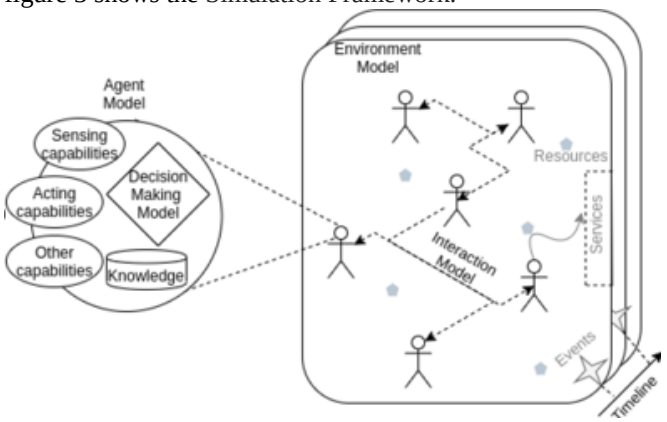

Figure 3: Simulation Framework

In-depth analysis of Internet of Things (IoT) big data may benefit from the use of ABSM distributed simulation. When dealing with multi-agent systems, it's critical to have one that can access both current and historical data, and traditional distributed simulation methods don't work well. To address scalability and performance demands, big data analytics requires a new approach to simulation. Simulation elements can be dispersed throughout a larger area, which is possible thanks to the ABSM model [10]. Agent collaboration and communication are simplified using this approach. Aside from the fact that discrete-event simulation (DES) does not have the same issues, agent-based modeling has its issues. It is an approach based on logical process modeling that is devoted to the study of processes. Even if it's simple to predict the behavior of an agent, DES has a hard time putting it into action.
ABSM has been used in a variety of industries, including manufacturing, retail, telecommunications, and traffic management, to simulate complex systems. Examples of applications of ABSM include acting cellular chemical interactions and the parking of automobiles. BSM has been applied in various ways that account for social and economic issues, such as the behavior of customers. Using an agent-based simulation model, [4] were able to simulate how customers behave in the media market. There's more on frequency-sharing mechanisms in [4], which examines the ABSM framework's heterogeneous nodes as agents. They found ABSM to be more effective in learning about market models than traditional analytical methods.
The financial crisis that resulted from cities being entirely shut down in several countries was enormous. People were at risk

because some chose to prevent financial loss at all costs. When it comes to the relationship between health and money, ABSM models were applied. In addition, ABSM models were utilized to determine how likely the virus would spread to a wide area. Such information is critical in deciding whether to partially or fully implement a shutdown.

3.3 Dataset

We trained and tested our algorithm on the UNSW-15 and NSL-KDD datasets. In this case, it is just a list of 49 sample features labeled "class." More than two and a half million records are available. It is large enough to use for both training and testing the model. In this case, the data is looking for a "label class." Use 10 neurons per node in a network, with an error rate of 1 percent for sending and selecting the desired data. It takes roughly 10 minutes to train a dataset 1,000 times. MATLAB 2019 and 2010a's neural network tool is used in our experiments. The neural network tool provides three methods for testing, training, and validating data.

4. Results and Discussion

The most remarkable about our concept is that it uses paired CSPR Spreading-Sequences that are made in a distributed and independent manner. Additionally, in the proposed work, most focus has been on crypto-agility. ChaCha20, for example, is a software-based cryptographic primitive that is well-suited to the Internet of Things and can be used in our approach.

We could determine our proposition's strength in AJ by looking at the CC features of the CSPR set of sequences. In the first place, a discovery has been made that there had never been a thorough investigation of the CC. As a result, we conducted both empirical statistics and a probabilistic examination of a large CSPR set of sequences.

On the other hand, the PRNG set of sequences is made up of a non-independent set of rows. As a result, mastering one line provides a foundation for learning others. Second, our insider attacker scenario relies heavily on this. It is possible to resolve the first issue without resorting to LFSR coding. It's only possible to solve the second problem with sets in which each sequence is unique. CSPR codes created independently fix each of these issues. This is the first time CSPR Sequence Sets have been proposed for WCS, and we believe we are the first to do so.

In our suggested CSPR-based sequence creation, unpredictability and independence are the most significant aspects. The CC of the sets' sequences is affected by this. What does this mean in terms of other sequence families?. It has not examined the full range of consequences these changes can have on existing systems in the real world. The CC statistics attributes are also crucial for a single node because the sequences constantly change.

As a result, this issue is no longer relevant. IoT devices like ChaCha20, built on a software-based approach, function swiftly with AES hardware modules on most SoCs.

Our MTD IoT solutions do not have a problem because we cannot be confident in concrete values. Any MTD-inspired system with many nodes and sequences will statistically converge to the CC and AJ features we explored. Even in our IoT environment.

Instead, the focus should be on safeguarding the Internet of Things. What we care about is the overall service provided by the IoT system, rather than the individual IoT nodes that make up that system, Even if just 10% of IoT nodes (or any particular node only 10% of the time) can withstand a +25 dB powerful jammer, we can still provide a specific service.

A "strength lies in (large) numbers" in IoT networks with a small number of nodes.

The most significant obstacles to a successful implementation. The spreading sequences must be synchronized in order for the signal to be demodulated at the RX-node for our concept to work in the real world. A solution to this problem exists since CSPR sequences are excellent at autocorrelation [4].

4 .CONCLUSION

The proposed method would be ideal for signature-based detection. Data from excellent and bad networks and various combinations of the two are used in the testing process. Using the new technology, an Internet of Things (IoT) controller may decide what data is safe and risky. It is then discarded since it is considered a threat. It works 84% of the time and only provides false positives 8% of the time. In this case, it is clear that the proposed strategy is accurate and works well with vast and diverse datasets

REFERENCES

1. Chen, P., Liu, S., Chen, B., & Yu, L. (2022). Multi-Agent Reinforcement Learning for Decentralized Resilient Secondary Control of Energy Storage Systems against DoS Attacks. IEEE Transactions on Smart Grid.

2. Li, X., Hu, X., Li, W., & Hu, H. (2019, May). A multi-agent reinforcement learning routing protocol for underwater optical sensor networks. In ICC 2019-2019 IEEE International Conference on Communications (ICC) (pp. 1-7). IEEE.

3. Charbonnier, F., Morstyn, T., & McCulloch, M. D. (2022). Scalable multi-agent reinforcement learning for distributed control of residential energy flexibility. Applied Energy, 314, 118825.

4. Kamruzzaman, M. D., Abdelmalak, M., Elsaiah, S., & Benidris, M. (2021, July). A Data-driven Shunt Dispatch Approach to Enhance Power System Resilience against Windstorms. In 2021 IEEE Power & Energy Society General Meeting (PESGM) (pp. 1-5). IEEE.

5. Ding, G. (2021). Multi-Agent Reinforcement Learning as Applied to Autonomous Systems (Doctoral dissertation, University of Colorado at Boulder).

6. Navas, R. E., Cuppens, F., Cuppens, N. B., Toutain, L., & Papadopoulos, G. Z. (2021). Physical resilience to insider attacks in IoT networks: Independent cryptographically secure

sequences for DSSS anti-jamming. Computer Networks, 187, 107751.
7. Wang, Y., Qiu, D., & Strbac, G. (2022). Multi-agent deep reinforcement learning for resilience-driven routing and scheduling of mobile energy storage systems. Applied Energy, 310, 118575.
8. Panfili, M., Giuseppi, A., Fiaschetti, A., Al-Jibreen, H. B., Pietrabissa, A., & Priscoli, F. D. (2018, June). A game-theoretical approach to cyber-security of critical infrastructures based on multi-agent reinforcement learning. In 2018 26th Mediterranean Conference on Control and Automation (MED) (pp. 460-465). IEEE.
9. Hanga, K. M., & Kovalchuk, Y. (2019). Machine learning and multi-agent systems in oil and gas industry applications: A survey. Computer Science Review, 34, 100191.
10. Yoon, S., Cho, J. H., Kim, D. S., Moore, T. J., Free-Nelson, F., & Lim, H. (2021). DESOLATER: Deep Reinforcement Learning-Based Resource Allocation and Moving Target Defense Deployment Framework. IEEE Access, 9, 70700-70714.
11. Alanya-Beltran, J., Raut, R., Patil, S., Christobel, Y. A., Vats, P., Nagaprasad, S., & Kumar, Y. P. (2022). Machine Learning-Based Intelligent Wireless Communication System for Solving Real-World Security Issues. Security and Communication Networks, 2022.