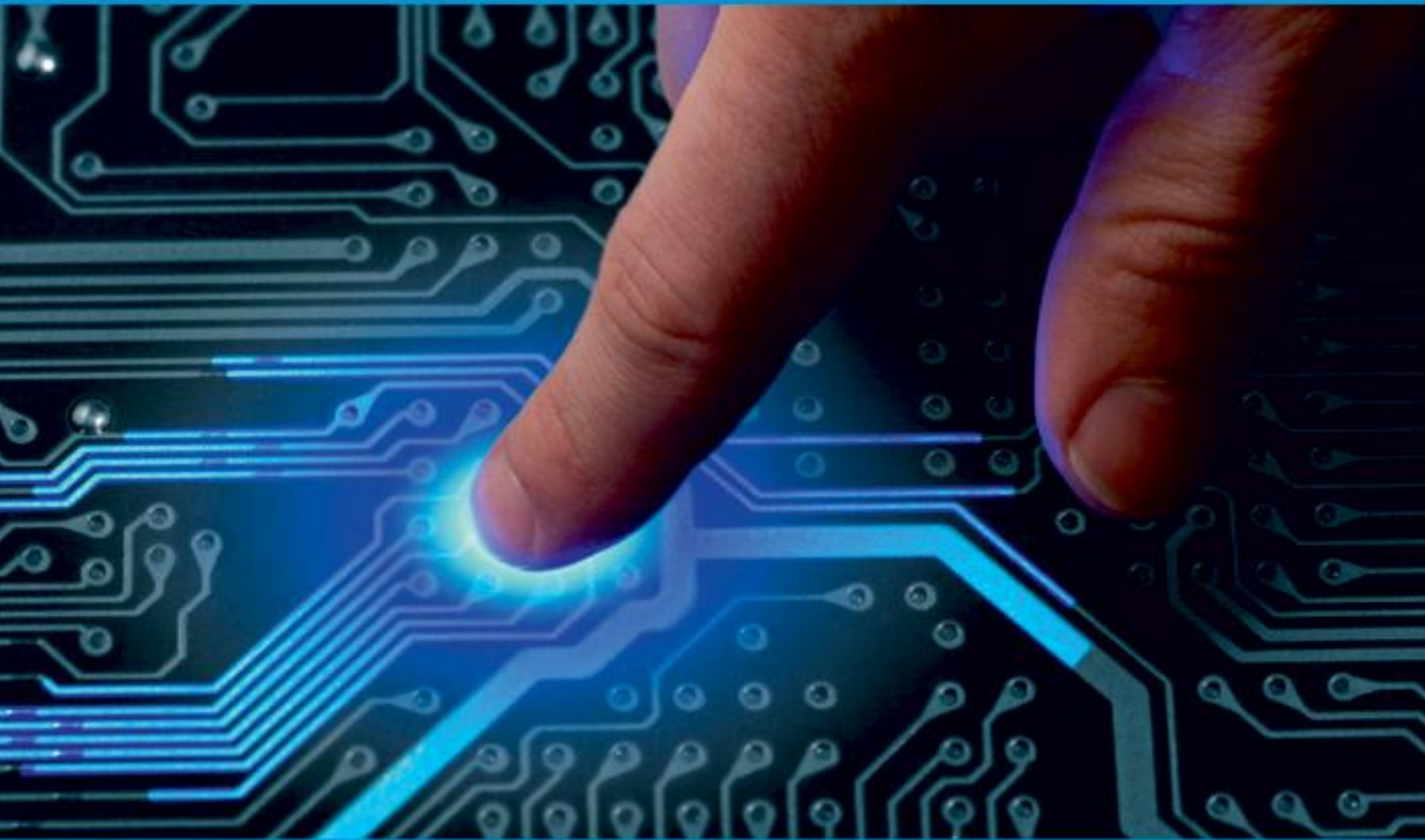
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 2, April 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

Adaptive RL-Based Optimization for Dynamic Task Scheduling in CC Environment

R Shiniktha , E V Vikram Bharathi, N Vimali, Ms. K Mohanapriya M.E.,

UG Students, Dept. of C.S.E., Velalar College of Engineering and Technology, Erode, Tamilnadu, India

Assistant Professor, Velalar College of Engineering and Technology, Erode, Tamilnadu, India

ABSTRACT: The allocation mechanism involves allocating tasks to virtual machines based on resource requirements. However, enhancing this allocation mechanism with reinforcement learning (RL) introduces a dynamic decision-making approach. RL models consist of agents, environments, and rewards, wherein agents learn to make decisions by interacting with an environment and receiving feedback in the form of rewards. In the context of task allocation, the RL agent could learn to optimize allocations based on factors such as waiting time, resource utilization, and cost. To integrate RL into the existing system, the first step is defining the RL environment, encompassing the current state of the task and virtual machine configurations. This involves specifying the state space, action space, and reward structure. Next, a suitable RL algorithm must be chosen based on the complexity of the problem. Algorithms like Q-learning or Deep Q Networks (DQN) could be employed to train the RL agent on historical data or simulated scenarios. Once trained, the RL agent's decision-making process replaces the initial first-fit algorithm, allowing the system to dynamically adapt to changing conditions and optimize task allocations over time.

KEYWORDS: Cloud Computing, Task Allocation, Virtual machine, Reinforcement learning, Optimization

I. INTRODUCTION

Cloud computing is a technology that enables users to access computing resources, such as servers, storage, databases, networking, software, and more, over the internet on a pay-as-you-go basis. It offers flexibility, scalability, and cost-effectiveness compared to traditional on-premises infrastructure. Major providers include Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP). Cloud computing is the on-demand access of computing resources—physical servers or virtual servers, data storage, networking capabilities, application development tools, software, AI-powered analytic tools and more—over the internet with pay-per-use pricing. In contrast to biometric systems that can only authenticate one person at a time, our system harnesses the power of face recognition models built upon the "labelled faces in the wild" dataset. This dataset comprises diverse facial images with corresponding labels and serves as the foundation for our system's accuracy. Utilizing state-of-the-art deep learning techniques, our system transforms each facial image into a 128-bit encoded vector. This vector representation is then used in conjunction with TensorFlow, a powerful machine learning framework, for classification tasks.

In cloud environment, maximum utilization of resource is possible with good resource management strategies. Workload prediction plays a vital role in estimating the actual resource required for successful execution of an application on cloud. The effective distribution and administration of computer resources, such as CPU, memory, storage, and network bandwidth, inside cloud architecture is referred to as resource utilization in cloud computing. In order to guarantee the affordability, efficiency, scalability, and dependability of cloud services, it entails optimizing the use of these resources. To optimize resource usage in the cloud, strategies including virtualization, load balancing, auto-scaling, and resource pooling are frequently employed. Utilize reinforcement learning to dynamically allocate tasks to virtual machines, ensuring optimal utilization of available resources and improving overall system efficiency.

II. RELATED WORK

Libing Wang [1] and his team proposed a Job-shop scheduling problem (JSP) is used to determine the processing order of the jobs and is a typical scheduling problem in smart manufacturing. Considering the dynamics and the uncertainties such as machine breakdown and job rework of the job-shop environment, it is essential to flexibly adjust the scheduling strategy according to the current state. In recent years, deep

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Refereed Journal |
|| Volume 12, Issue 2, April 2024 ||
International Conference on Recent Development in Engineering and Technology – ICRDET 24
Organized by
Dhaanish Ahmed Institute of Technology, KG Chavadi, Coimbatore, Tamilnadu, India

reinforcement learning (DRL) has achieved remarkable results on various scheduling problems, including dynamic robot control , dynamic resource management , Internet of Things communication and smart manufacturing . Due to the advantage of real-time decision-making, deep reinforcement learning can give immediate action for the problem if the model is well-trained. A dynamic adaptive scheduling system using deep reinforcement learning is introduced to cope with the unexpected conditions in a real-world environment. Muhammed Tawfiqul Islam [2] and his team proposed a Cloud offers affordable compute resources which are easier to manage. Hence, many organizations are shifting towards a cloud deployment of their big data computing clusters. However, job scheduling is a complex problem in the presence of various Service Level Agreement (SLA) objectives such as monetary cost reduction, and job performance improvement. In addition, the DRL-based agents can also learn the inherent characteristics of different types of jobs to find a proper placement to reduce both the total cluster VM usage cost and the average job duration such as the monetary cost reduction. In this work, we focus on the SLA-based job scheduling problem for a cloud-deployed Apache Spark cluster. When a Spark cluster is deployed, it can be used to run one or more jobs. Generally, when a job is submitted for execution, the framework scheduler is responsible for allocating chunks of resources (e.g., CPU, memory), which are called executors. A job can run one or more tasks in parallel with these executors. The default Spark scheduler can create the executors of a job in a distributed fashion in the worker nodes. This approach allows balanced use of the cluster and results in performance improvements to the compute-intensive workloads as interference between co-located executors are avoided. Huayu Zhu [3] and others.Resource scheduling problems (RSPs) in cloud manufacturing (CMfg) often manifest as dynamic scheduling problems in which scheduling strategies depend on real-time environments and demands. Generally, multiple resources in the CMfg scheduling process cause difficulties in system modeling. To solve this problem, we propose Sharer, a deep reinforcement learning (DRL)-based method that converts scheduling problems with multiple resources into one learning target and learns effective strategies automatically. Yi Wei [4] and others. Workflow technology is an efficient means for constructing complex applications which involve multiple applications with different functions. In recent years, with the rapid development of cloud computing, deploying such workflow applications in cloud environment is becoming increasingly popular in many fields, such as scientific computing, big data analysis, collaborative design and manufacturing. With the cloud rapid development of computing technologies, a large number of cloud based applications are delivered across the internet. Cloud workflow applications are typical ones which can be used to handle complex and combinatorial tasks in academia and industry. Several works have been proposed to address workflow scheduling problems in the literature . Most studies are based on the assumption that the execution time of each task on different types of VM instances has been known in advance and it will not change. However, this assumption is not reasonable because SaaS providers hardly know the real performance of VMs in a dynamic cloud environment. In this study Kaixuan Kang [5] and others Intelligent task scheduling solutions are highly demanded in the operation of complex cloud data centers so that resources can be utilized in an energy-efficient way while still ensuring various requirements of users. In this study Kaixuan Kang [5] and others Intelligent task scheduling solutions are highly demanded in the operation of complex cloud data centers so that resources can be utilized in an energy-efficient way while still ensuring various requirements of users.

III.PROPOSED SYSTEM

The proposed system introduces a dynamic task allocation mechanism empowered by reinforcement learning techniques to address the limitations of the existing system. Leveraging machine learning algorithms, the system learns from historical data and adapts its allocation strategies in real-time, ensuring optimal resource utilization. By incorporating intelligence into the decision-making process, the system can make informed choices based on factors such as workload patterns, resource availability, and cost considerations. This adaptive approach aims to enhance overall system efficiency, reduce waiting times, and optimize resource allocation, ultimately leading to a more responsive, energy consumption, to improve performance, and cost-effective cloud computing environment.

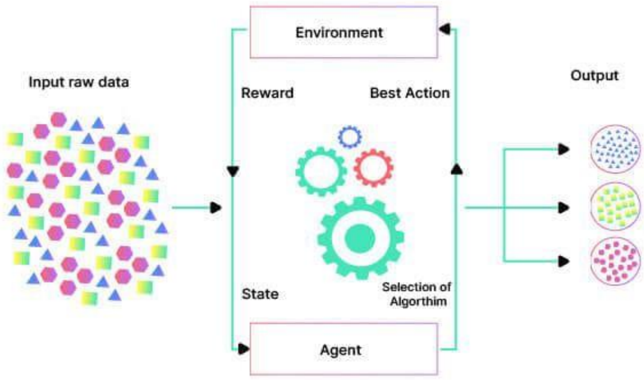
| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 2, April 2024 ||

International Conference on Recent Development in Engineering and Technology – ICRDET 24

Organized by

Dhaanish Ahmed Institute of Technology, KG Chavadi, Coimbatore, Tamilnadu, India



WORKFLOW OF RL

IV. RL FLOWCHART

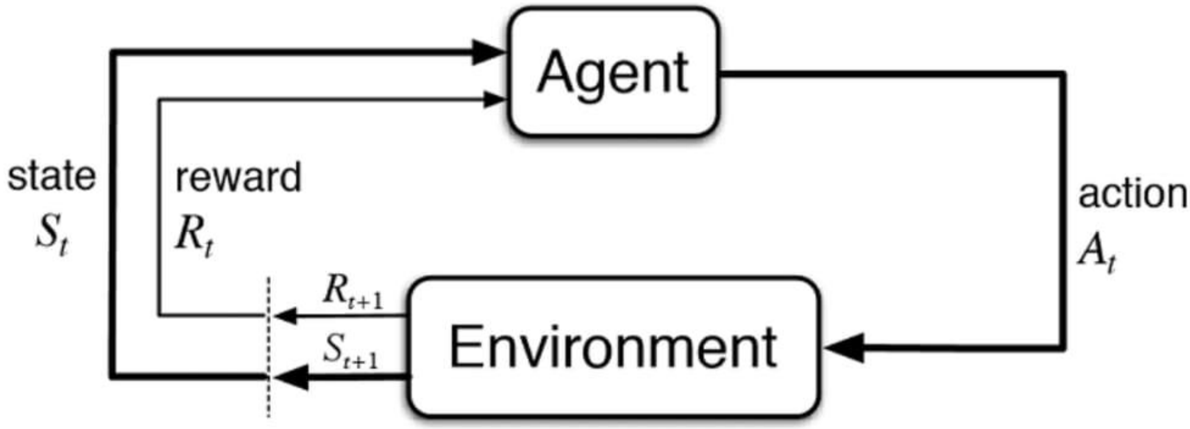


Fig.1. Flowchart of RL

Step 1: TASK AND VM DETAILS

It focuses on creating a user interface using Tkinter, allowing users to input task and VM details conveniently. It includes functions to dynamically generate entry fields, enabling users to specify essential parameters such as Task ID, RAM, Bandwidth, and OS for tasks, as well as VM ID, CPU, Memory, and Storage for virtual machines.Step 2: Dashboard Navigation

Step 2: TASK ALLOCATION

The task allocation module implements a First Fit algorithm to allocate tasks to virtual machines based on their resource requirements. It identifies suitable VMs for each task, optimizing resource utilization. The allocation details, including the task ID and the corresponding allocated VM, are displayed in the console for user feedback.

Step 3: REINFORCEMENT LEARNING INTEGRATION

This module enhances the task allocation strategy by integrating a reinforcement learning model. It incorporates historical allocation data and real-time conditions to dynamically adjust the allocation policy. The reinforcement learning algorithm learns from past experiences, adapting to changing workloads and improving the overall efficiency of task allocation in cloud computing environments.

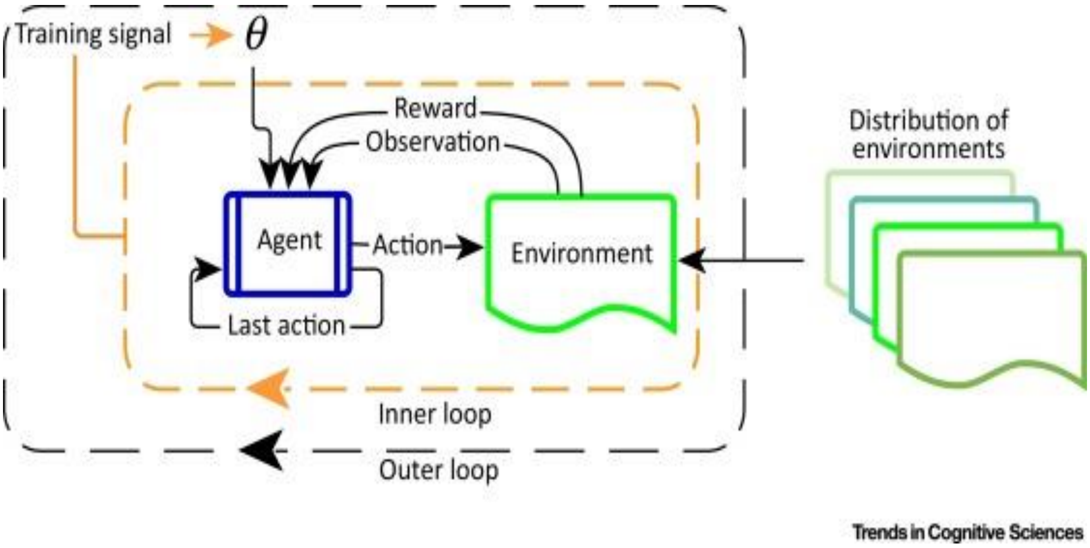
| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 2, April 2024 ||

International Conference on Recent Development in Engineering and Technology – ICRDET 24

Organized by

Dhaanish Ahmed Institute of Technology, KG Chavadi, Coimbatore, Tamilnadu, India



Step 4: RESULT VISUALIZATION

The result visualization module employs Matplotlib to create interactive charts displaying key metrics such as waiting times, speeds, and costs for the allocated tasks. Users can visually assess the system's performance and make informed decisions regarding resource scaling and optimization based on the presented data.

Step 5: USER INTERFACE INTERACTION

Users can input details, initiate allocation, and visualize the results through charts, providing a user-friendly experience for understanding and evaluating the efficiency of the task allocation system.

V. RESULTS AND DISCUSSIONS

By applying reinforcement learning (RL) to the task allocation system in cloud computing offers a dynamic and adaptive approach to optimizing resource utilization while considering cost factors. By leveraging RL algorithms such as Q-learning or Deep Q Networks (DQN), the system can learn optimal task allocation policies through interactions with the environment over time. RL enables the system to continuously refine its decision-making process based on feedback received from previous allocations, thus improving its ability to select the most cost-effective virtual machines for each task. Through trial and error, the RL agent learns to balance resource requirements with cost considerations, ultimately leading to more efficient and economical task allocations within the cloud infrastructure. Moreover, the utilization of RL introduces a level of adaptability and responsiveness to changing conditions in the cloud environment. RL agents can adapt their allocation strategies in real-time based on fluctuations in resource availability, workload demands, and cost variations. This adaptability ensures that the system remains effective in diverse and dynamic computing environments, where traditional static allocation methods may fall short. By harnessing the power of RL, the cost-aware task allocation system becomes more resilient to uncertainties and fluctuations, thereby enhancing the overall efficiency and cost-effectiveness of cloud computing operations.

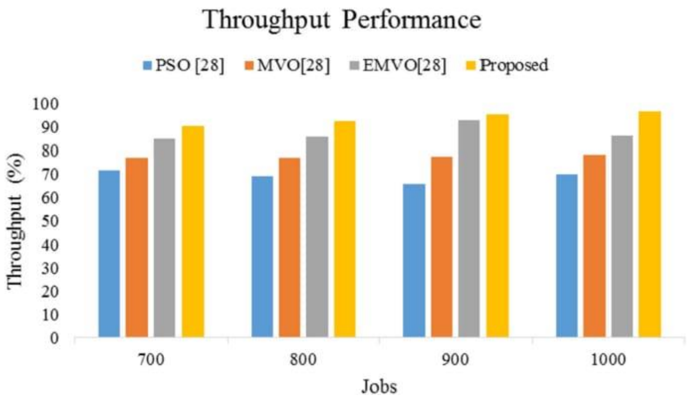


Fig.2.Comparision of Performance

VI. Conclusion and Future Work

In conclusion, this project addresses the dynamic challenge of task allocation in a virtualized environment using a First Fit algorithm. The reinforcement learning technique enables the system to intelligently allocate tasks to virtual machines, considering factors such as waiting time, speed, and cost. The graphical output using Matplotlib provides an effective means of visualizing the allocation results, allowing users to gain insights into the system's performance. The successful implementation of this system contributes to the field of task scheduling and virtualization, offering a solution that balances efficiency and resource optimization in a dynamic computing environment. Further enhancements and adaptability to diverse scenarios can be explored to extend the system's applicability and robustness. As a potential avenue for future enhancements, the project could be extended to incorporate more advanced machine learning algorithms, such as deep reinforcement learning, to further optimize task allocation in dynamic virtualized environments. Additionally, the system could benefit from a user-friendly graphical interface for real-time monitoring and control, enabling users to interact with and influence the allocation process. Integration with cloud computing platforms and the ability to adapt to varying workloads and resource availability would enhance the system's scalability. Furthermore, exploring the incorporation of energy-aware scheduling mechanisms and security considerations could contribute to a more comprehensive and resilient solution for modern computing infrastructures.

References

1. Libing Wang, Xin Hu, Yin Wang, Sujie Xu, Shujun Ma, Kexin Yang, Zhijun Liu, Weidong Wang, "Dynamic job-shop Scheduling in smart manufacturing using deep reinforcement learning," February 2021 https://doi.org/10.1016/j.comnet.2021.107969.
2. Muhammed Tawfiqul Islam, Shanika Karunasekara, Rajkumar Buyya, "Performance and Cost-Efficient Spark Job Scheduling Based on Deep Reinforcement Learning in Cloud Computing Environments," July 2022 10.1109/TPDS.2021.3124670.
3. Huayu Zhu, Mengrong Li, Yong Tang, Yanfei Sun, "A Deep-Reinforcement-Learning-Based Optimization Approach for Real-Time Scheduling in Cloud Manufacturing," January 2020 10.1109/ACCESS.2020.2964955.
4. Yi Wei, Daniel Kudenko, Shijun Liu, Li Pan, Lei Wu, Xiangxu Meng, "A Reinforcement Learning Based Workflow Application Scheduling Approach in Dynamic Cloud Environment," Oct 2018 https://doi.org/10.1007/978-3-030-00916-8_12.
5. Kaixuan Kang, Ding Ding, Huamao Xie, Qian Yin, Jing Zeng, "Adaptive DRL-Based Task Scheduling for Energy-Efficient Cloud Computing," December 2022 10.1109/TNSM.2021.3137926.
6. Liu, Renke, "Deep Reinforcement learning-based dynamic scheduling," March 2022, 10356/158353.
7. Zhiping Peng, Delong Cui, Jinglong Zuo1, Qirui Li, Bo Xu, Weiwei Lin, "Random task scheduling scheme based on reinforcement learning in cloud computing," August 2015, I10.1007/s10586-015-0484-.
8. Ding Ding a, Xiaocong Fan b, Yihuan Zhao a, Kaixuan Kang a, Qian Yin a, Jing Zeng, "Q-learning based dynamic Task Scheduling for Energy-Efficient Cloud Computing," Febuary 2022, doi.org/10.1016/j.future.2020.02.018.
9. L. Jiang, H. Huang, and Z. Ding, "Path planning for intelligent robots based on deep Q-learning with

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Refereed Journal |

|| Volume 12, Issue 2, April 2024 ||

International Conference on Recent Development in Engineering and Technology – ICRDET 24

Organized by

Dhaanish Ahmed Institute of Technology, KG Chavadi, Coimbatore, Tamilnadu, India

experience replay and heuristic knowledge,”, July 2020, pp. 1179–1189.

10. O. Kundu, S. Dutta, S. Kumar, Deep-pack, “A vision-based 2D online bin packing algorithm with deep reinforcement learning,” in January 2019, https://doi.org/10.1109/RO.
11. X. Liao, Z.Liu X.Hu, S. Ma, X.Li L.Xu, W. Wang, F.M. Ghannouchi, Distributed intelligence: a verification for multi-agent DRL based multibeam satellite resource allocation, IEEE Commun. Lett. (2020), https://doi.org/10.1109/
12. Y. Liu, L. Wang, X. V. Wang, X. Xu, and L. Zhang, ‘‘Scheduling in cloud manufacturing: State-of-the-art and research challenges,’’ Int. J. Prod. Res., vol. 57, nos. 15–16, pp. 4854–4879, Aug. 2019.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

Scan to save the contact details

www.ijircce.com