International Journal of Science, Research and Technology (IJSRAT)

|Bimonthly | Peer-Reviewed | Scholarly Journal|

|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|

DOI: 10.15662/IJSRAT.2026.0901008

Reinforcement Learning for Dynamic Cloud Resource BI Optimization

Aditi Namdeo

AI Researcher, Amazon, Seattle, USA

Publication History: 11-1-2025 (Received); 30-1-2025 (Revised); 9-2-2026 (Accepted); 18-2-2026 (Published).

ABSTRACT: Dynamic cloud environments: These environments have a varying volume of data, query workload, number of users and/or business continuity objectives (SLOs) for Business Intelligence (BI) workloads. Due to these uncertainties, it will be impossible to balance the three important factors of performance, energy efficiency and cost with fixed rule and threshold based autoscaling approaches. In this research article we want to propose a framework for dynamically optimizing cloud resource BI based on Reinforcement Learning (RL) model who will continually interact with the cloud environment and learn a set of optimal policies for cloud resource allocation. This framework consists of five layers such as BI workload monitoring, state space construction, RL decision engine, resource orchestration and evaluation of performance feedback. Every system metric (such as CPU usage, memory usage, query response time, queue length, storage I/O, workload priority and cost constraints) is important. Those can be the scaling of virtual machines, moving containers around and changing memory sizes, prioritizing BI queries, moving workloads from node to node, etc., and are decided by the RL agent. The rewards can be specified as a multi-objective function, which include reduction of response time, reduction of cloud cost, resource utilization gain and SLA and energy consumption satisfaction. The proposed model is adaptive – it adapts itself based on the trends of the historical data or real-time data of BI workload. Algorithms like Q-learning, Deep Q-Networks and Proximal Policy Optimization could be potentially applied for an intelligent decision making in cloud based BI platforms, the study mentions. This framework can enhance the analytical responsiveness, minimise resources wastage and enable cost-aware business decision systems. It also helps businesses in planning for capacity with a pro-active assessment and hand over their real-time dashboards and predictive analytics services. Lastly, reinforcement learning can be leveraged for scaling and adapting for optimal BI workloads in a complex, dynamic and data rich cloud environment.

KEYWORDS: Reinforcement Learning, Cloud Computing, Business Intelligence, Resource Optimization, Dynamic Autoscaling, Deep Q-Network, SLA Management

I. INTRODUCTION

Today's world Business Intelligence (BI) system is one of the most important Technology Enablers with the advent of Cloud Computing. Decision making and visualizing vast amount of data collected and stored in different Business Intelligence platforms is becoming more prominent in the businesses. Some of these platforms can be used to disseminate information like real time dashboards, sales forecasting, customer behaviour analysis, supply chain monitoring, financial reporting, risk assessment and predictive analytics [1]. The business world has been relying on data more and more; hence, they are in need of Cloud BI infrastructure. However, traditional static computing systems make the (fairly small and static) assumption that the number of users, their queries, the volume of data to which these users have access and their load intensities and response time requirements are fairly similar. This dynamic nature brings a big challenge to efficiently allocate resources in cloud, guaranteeing high performance, low operational cost and service reliability [2].

There is some variability in BI workloads. In peak hours, the routine reports can be processed in BI system and business rules can be used for processing ad hoc analysis queries. However, in the same system there might be sudden increase in user requests and its data processing requirements due to different reasons such as seasonal demands, emergency case for decision making, heavy data processing requirement etc [3]. Without investing in resources, possibly Dashboard response time will be slow, SLA may not be met and everyone won't be happy with the BI platform. But, when resources are oversized, it can have a negative effect on performance and lead to cloud costs and wasting of energy and infrastructure. But, it is not about the quantity of resources but about the right resources at the right time, suitable to the type of workloads [4].

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

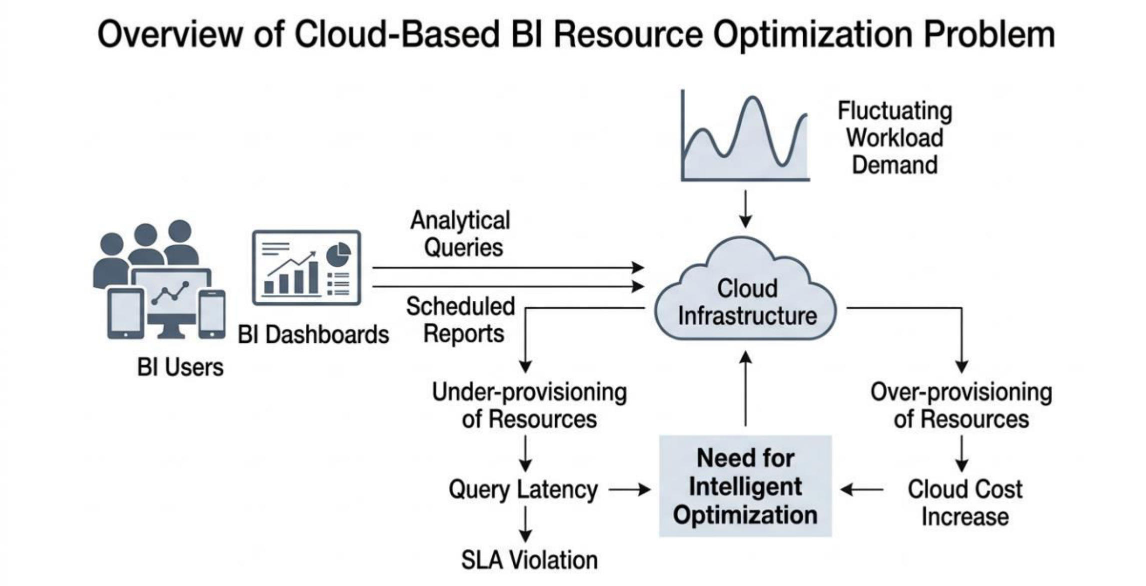

Figure 1: Overview of Cloud-Based BI Resource Optimization Problem

Cloud resource management is usually performed via a set of guidelines, thresholds or even manually. For instance, you can set it up, such that when memory is low, it will add memory to your computer, and when CPU is high, it will remove memory from your computer. These auto scaling mechanisms are simple to implement, threshold based and only work in simple and static BI environments [5]. Commonly they respond when the workload degradation has occurred and they don't have a complete understanding of the workload patterns. However, they do not take multiple objectives of optimization into account, e.g. cost, latency, throughput, energy use and service level agreements compliance. Thus traditional approach to resources allocation might not work for real time BI workloads for adaptive, intelligent and continuous decision making [6].

Reinforcement Learning (RL) is one of the various solutions. RL is one of the types of ML, where an intelligent agent makes decisions based on the interaction with the environment. Without any particular rules, the only thing the agent has to do is to find out what the state of the system is, use this information to select an action and receive the reward for that action; it will learn to select a better action. Depending on the type of cloud resources, the optimisation of the cloud resources may be dynamic, uncertain and always changing. An RL agent could then be trained on the past and real time workload behaviors to find out the resource allocation policies that improve the long-term system behavior [7] [8].

In the case of dynamically optimising use of cloud resources for BI, the scene of the RL agent is the scene of the decision for those resources or the environment of the cloud decision making (in our case). This can be the status of anything in the environment such as CPU utilization, memory utilization, query response time, number of users in the workload queue, storage input/output rate, network bandwidth, cost per resource unit and status of SLA etc. These states can be utilized for communicating with RL agent so that action, such as scaling virtual machines, adding or removing containers, allocating or reclaiming memory, changing the number of processing units, migrating workloads between clouds, prioritizing high value BI queries, workload balancing between cloud nodes, etc., can be performed. Action can then be used to check the success of the done action in the next step, depending on the action performed, by using the reward function. Positive reward can be based on any number of factors, like providing a response faster, better use of resources, lower costs or achieving SLAs. These are the points where queries are getting hold up, resources are not being utilized and thus quality of service (SVC Services) might be getting impacted and a penalty (may be) [9] [10].

The suggested BI optimization model based on Reinforcement Learning approach is divided into 5 different layers. There has been the BI workload monitoring layer, which is always activated in collecting data related to users' requests, query running time, dashboards use, data processing workload and infrastructure utilization data. The second layer, will be the state space construction layer where the raw monitoring data will be fed into system state that RL model knows. The third one is RL decision engine that decides the most appropriate resource management action which in such cases,

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

can be achieved by algorithms such as Q-learning, Deep Q-Networks, Actor Critic method and Proximal Policy Optimization. The fourth layer is a cloud resource orchestration layer which actually takes the selected action, to communicate with cloud services, container platforms, virtual machines and workload schedulers. The 5th layer is feedback and evaluation layer to evaluate the impact of action and the learning model can be changed with the application of reward mechanism.

It's an innovative research on Intelligent Learning and real world cloud resource management. BI systems must be quick and accurate as business users could rely on the results of the analysis to make decisions in time. When BI reporting is lagging, it can have an impact on your business planning, customer service, financial management, strategic decision making - and more. Cloud expenses are issues organisations have to consider in the meantime. The majority of the cloud-based platforms solutions are "pay as you go" - if there is Resource Inefficiency, then this can be a waste of money. By using an RL based model, the organisation's performance and cost has been able to be scaled-up/scaled-down as well as the ability to work load distributed more effectively.

Reinforcement learning has another advantage over it, which is that it could be used in pro- active optimizations. Not only an "if my bandwidth threshold is exceeded, use rule based system", but to be used to make decisions based on throughput learnt during past behaviour of the workload so that it would never reach such a point that its performance would be degraded. For instance, if BI queries are likely to be made at a specific time of day you can leave the resources allocated accordingly, or if BI queries are likely to be made in a specific period of reporting then the resources can be allocated accordingly. This will guarantee the responsiveness of the system, as well as prevent any sudden decrease in performance and/or any system failure.

But, it also comes with some challenges in the design of effective RL based cloud BI optimization framework. To ensure the reward function is well designed and encompasses a variety of objectives, such as: latency reduction, cost control, resource efficiency and SLA adherence. The state space should contain sufficient information to make it a good description of the cloud environment, but without adding about too much information to make the model too complicated. The learning process should not also make decisions that are unstable that the impact BI services. With this in mind, this study is going to try to define a framework for implementable, measurable and business like Reinforcement Learning.

In conclusion, Reinforcement Learning shows a great potential in dynamically optimizing cloud resources in BI systems. RL can assist in optimisation of cloud resources, cost reduction, increase resource utilisation and facilitate business analytics, which can in real-time determine what resources should be allocated based on workloads. To cope with the adaptive optimization scenario in the current cloud based BI environment, a framework to monitor the BI workloads, make intelligent decision, coordinate resources and even a feedback mechanism are suggested in this paper.

II. RELATED WORK

With the ever-changing nature of the workloads in the cloud, the study of cloud resource optimization is also voluminous since modern cloud platforms have to manage fluctuating workloads while keeping their performance, cost efficiency, and compliance to service-level agreements intact. Previous works focused mainly on two kinds of techniques, namely the rulebased autoscaling and threshold policies, the heuristic scheduling and predictive workload modeling. They can be effective for stable workloads, but struggle to deal with workloads with high change rates and with multiple objectives, such as latency and utilization, that have to be optimized simultaneously. Hence, in the recent studies, one can see the need to concentrate on machine learning and reinforcement learning for performing adaptive cloud resource management [3].

It is concluded that Reinforcement Learning is the suitable technique to be used for cloud auto-scaling and it is possible to let the system learning by interacting with the environment on the basis of the reward/punishment it receives rather than fixed rule. In addition, Deep Reinforcement Learning (DRL) [4] is applied to this research to learn the system from low-level to high-level, and high dimensional cloud environment.

However, the majority of literature available is on the general aspects of cloud applications, microservices, serverless platforms or virtual machine scheduling. So little work is being done on the workload side of Business Intelligence where query latency, responsiveness of the dashboards, priority of analytical workloads and reliability of reporting are important. In this study, BI specific optimization framework will be proposed that will correlate the workload

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

monitoring, RL decision making and cloud orchestration and provide feedback on the performance to dynamically optimize the BI resources [11] [12].

III. REINFORCEMENT LEARNING FOR DYNAMIC CLOUD RESOURCE BI OPTIMIZATION

The proposed Reinforcement Learning for Dynamic Cloud Resource BI Optimization framework is to adapt the cloud resources intelligently for workloads management in cloud environment of the Business Intelligence applications in a feedback driven manner. Generally speaking, BI systems tend to be characterized by a large number of users, real time (or near real time) decision support, creating dashboards, doing analytical queries and having a large number of structured and semi-structured data. These workloads are dynamic, meaning that they can vary during the day, month or even the business cycle depending on the query volume and frequency, end user activity, data volume and reporting needs. Thus, there needs to be a flexible resource allocation model. The proposed framework continuously monitors the cloud-BI environment and learns the behavior of the above workloads and makes decision on resource management that maximizes the workloads' performance while minimizing cost and maintaining the service quality.

These 5 layers are: BI workload monitoring layer, state-space construction layer, reinforcement learning decision engine, cloud resource orchestration layer and feedback-evaluation layer. There are multiple layers and each of them plays its role, and is a part of a lifelong learning process. This framework's basic idea is to turn the reactive resource management into intelligent and proactive resource optimizations in the cloud system.

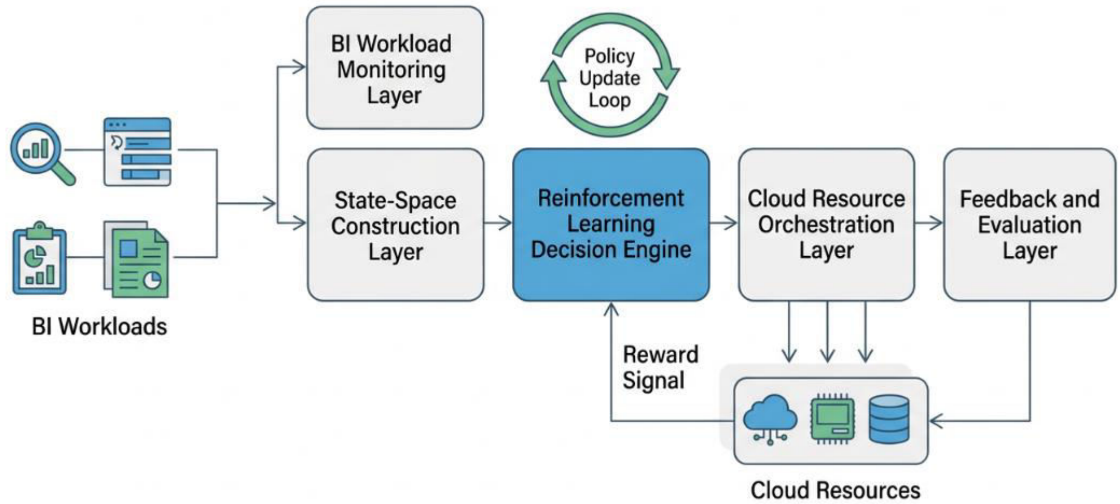

Figure 2: Proposed Reinforcement Learning-Based BI Optimization Framework

3.1 BI Workload Monitoring Layer

The top-level of the framework is the BI workload monitoring level. This layer is needed to get realtime data from the BI platform and the cloud infrastructure. The backbone of Reinforcement Learning is monitoring as it involves observing the environment. Layer information monitored include CPU Usage, Memory Used, Disk I/O, Network Bandwidth, Dashboard refreshes time, Number of Users, Queue length, Workloads arrival rate, failed requests and delayed requests.

In a BI environment not every workload is created equally. For instance, a real-time executive dashboard might need quicker processing than a weekly report that is scheduled. Similarly, financial operations monitoring, in the form of fraud detection and financial analysis, might not be as latency sensitive as exploration of historical data. Monitoring layer also monitors the characteristics of workload such as type of query, priority level, expected query response time, amount of data touched and service level agreement requirements. This will enable the RL agent to know the monetary value of each workload, and the technical condition of the system.

Use the monitoring layer in cloud monitoring solutions, BI server logs, query engines, container metrics and virtual machine metrics and cost-monitoring dashboards. All the data so collected is then passed on to the next one which will convert the data to the representation of the state in structured form.

|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

3.2 State-Space Construction Layer

The second layer is one of construction of state space. The raw monitoring data can't be directly used by a Reinforcement Learning model, but has to be transformed into meaningful states. A state represents the cloud-BI environment at a certain point in time. In this scenario, the state can come up with variables relating to technical, workload, cost and performance.

The typical state vectors can be: Percentage of CPU utilized, Memory utilization, I/O (storage), Network utilization, Number of pending BI queries, Average query response time, Dashboard latency, Resource cost per unit time, SLA violation rate, Energy consumption level. Can also involve workload aspects such as complexity of query, number of users, report priority and expected completion date.

This could be an 85% CPU, 78% Memory, high query queue length, high dashboard latency and a normal cost budget. This state could be utilized for implementing decisions to increase the resources by RL agent. The CPU could be low, the memory stable and the query traffic could be decreasing in another state. The agent could opt to reduce resources at that time to avoid cloud overages.

The pre-processing of data is done by the state-space construction layer too. Removes noise from the filters, normalizes the metrics, supports the absence of certain metrics and groups of similar workloads as needed This is very important as the states are represented correctly and consistently; otherwise learning decisions may be made based on wrongly. Is able to learn meaningful relationships between workload demand, resource use and system performance with a good definition for state space, RL model.

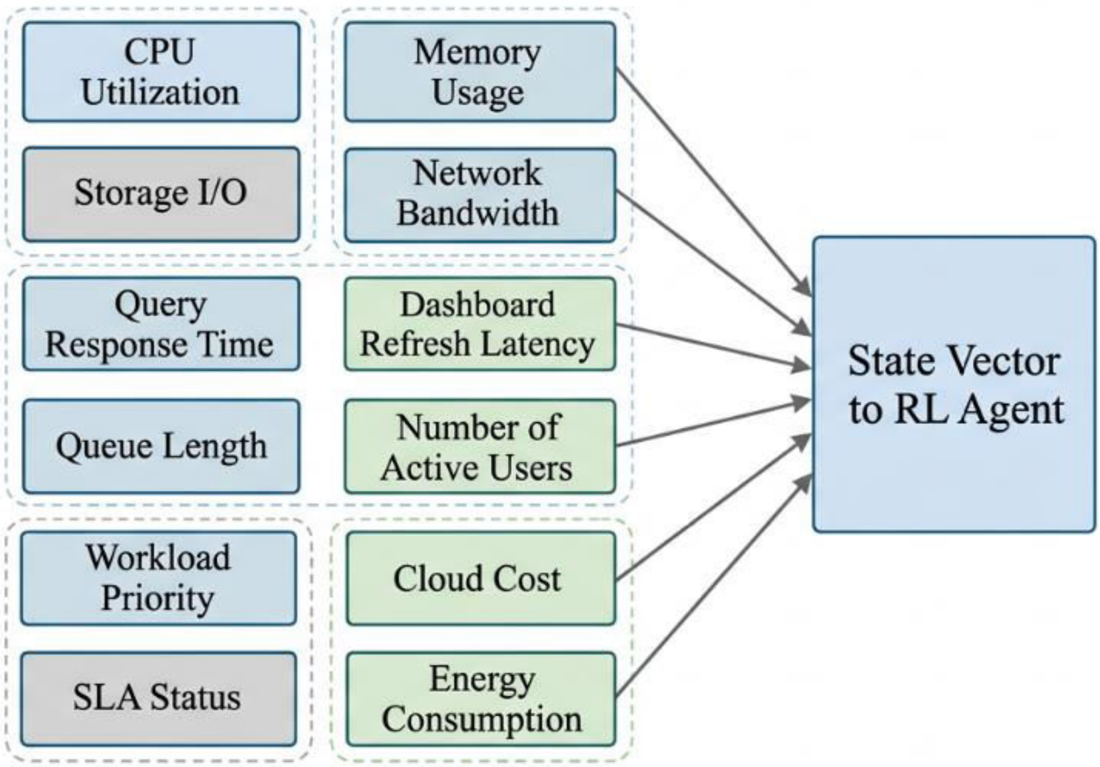

Figure 3: State-Space Representation for BI Workload Monitoring

3.3 Reinforcement Learning Decision Engine

Reinforcement learning decision engine is the key component of the proposed framework. It is the ‘Smart element' and is able to ‘learn' how to optimise the resources by conversing with the cloud environment. The RL agent looks at the state, decides on the next action, gets a feedback from the reward function and updates its policy to decide on the next action.

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

It differs in the environment and different RL algorithm will be used to achieve it. For smaller, simpler cloud environments, the use of Q-learning might be possible, as it is easy to implement, and suitable for the discrete state-action space. But, modern cloud-BI platforms are typically complex and have large state spaces. In this particular use case, Deep Q-Networks (DQN), Actor-Critic (AC) or Proximal Policy Optimization (PPO) can all be used. These algorithms could be capable of taking a higher dimensional input, and could learn more flexible resource allocation policies.

Actions space: Space of actions that the RL agent can take. These can involve scaling up or down the number of VM's, scaling up or down containers horizontally, add memory, change priority of queries run, delay low priority batch reporting operations or redistribute the dashboard request among available servers. The action space needs to be crafted so that the agent will be able to help optimise the system without performing 'bad' or 'destabilising' actions.

For example, when the queues of queries are big, and the response time of the dashboard is long, the RL agent can take an action to deploy more containers and/or provide more processing resources. During downtimes, the agent can scale down the number of instances, to save costs. With the one node being very busy and the other not, then the agent can be responsible for load balancing or workload migration. Such decisions not only rely on the hard-thresholds, but are also learnt in a few interactions with the environment.

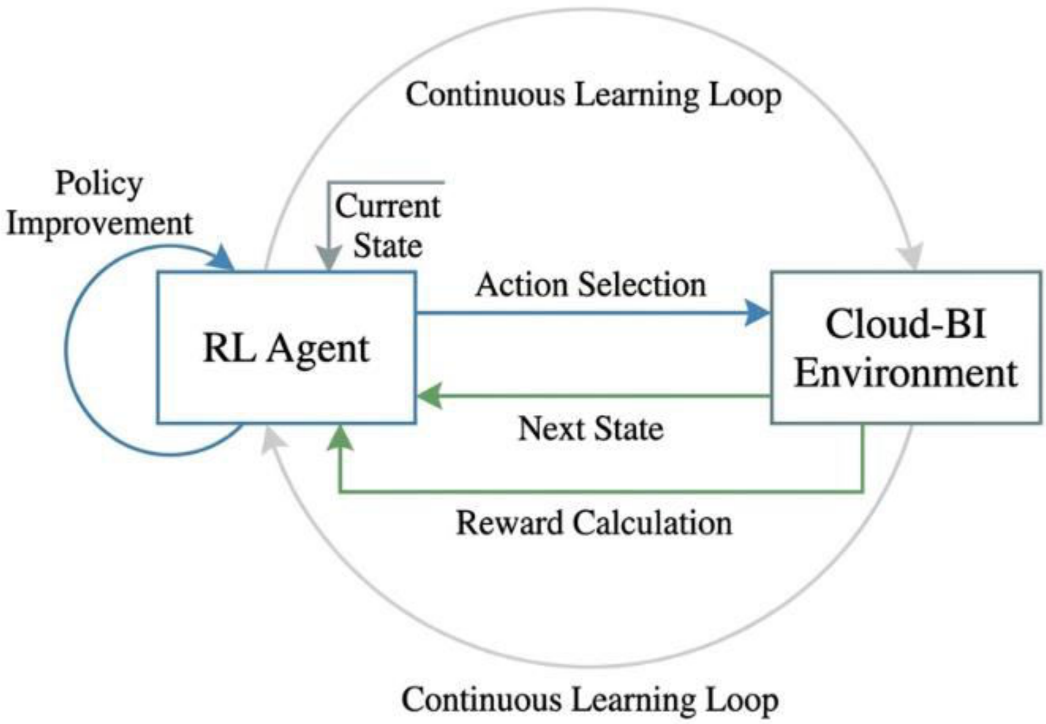

Figure 4: Reinforcement Learning Interaction Cycle for Cloud-BI Environment

3.4 Reward Function Design

Probably one of the most important parts of the framework is the reward function as it is based on the reward function that a learning conduct of the RL agent is guided. The problem is that the reward function might be incorrect, it might optimize some and optimize some other thing as well. Optimising goals of Cloud-BI can be different, hence reward function can be a combination of the performance, cost and resource usage, SLA and energy.

The reward function that is proposed can be a weighted multi-objective function. Positive rewards are given for the agent when he/she makes the query response time shorter, refreshes the dashboard faster, uses more resources, doesn't break the SLA and saves operating cost. In a situation where latency is high, there is excessive resource wastage, query failures or SLA violations and scaling actions, negative rewards are applied to the system.

The reward function can, for instance, contain the following components: latency reward, cost reward, utilization reward, SLA reward and energy reward. The more "fast" the agent is at responding to a query, the greater the amount

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

of reward is to the agent as long as it does not cost the agent much for their resources to increase. However if it grows faster, and leaves resources unused, then the reward is not deducted, but the cost penalty is deducted from the reward. If the agent attempts to reduce the resources used by too much, and is slowing down the dashboard, then the agent is negatively rewarded. This is a system that provides justice in the decision making process - not optimisation for either side.

The reward system is also one way that the agent can learn how to achieve the long-term rewards. This will involve an additional cost, however in the event that it were done prior to a peak workload, it would avert potential SLA violation over the life of the workload. Such switches and their utilization for carrying out actions that will pay-off in the long term can be learnt by the RL model.

3.5 Cloud Resource Orchestration Layer
The decision is made by the RL agent and the cloud resource orchestration layer makes the decision in real cloud. It is a Smart execution bridge – i.e. it is a link between Smart Decision Engine and Cloud infrastructure. Integrates with virtual machines, containers, load balancers, storage, query and scheduling.

Then, based on the RL agent's determination, additional virtual machines can be added or scaling up, for example, a Kubernetes cluster to add more containers, at the orchestration layer. This layer will enable the agent to reduce idle instances of this layer – not when critical BI queries are still running – during resource reduction. In the event that a migration is necessary, then the orchestration layer will transfer some of the activities to non-occupied nodes. Can be integrated with BI query scheduler, so that it can be used for prioritizing high priority analytical tasks if query prioritization is needed.

Some safety controls should be there in the orchestration layer, as well. Cloud resource actions can have an effect on live BI services and each potential for a resource action should be tested prior to deciding to take a certain action. For instance a set of critical reports should not be 'resources' that are to be terminated by the system. It shouldn't also have too many scaling operations in a short amount of time as too many scaling operations may also cause instability and additional cost. So, at the orchestration level adding policy constraint (maximum and minimum resource limits) and cooldown periods rollback mechanisms is possible.

3.6 Feedback and Evaluation Layer
The feedback/evaluation layer is the last layer. It provides a comparison of all the actions undertaken on the effectiveness of BI performance and the cloud resource efficiency. The evaluation layer is used to make comparisons between the condition of the system "before" and "after" the action. It validates query latency improvement, resources utilization and cost reduction and if the SLA requirements have been met.

This feedback is converted in to the reward signal, which is again passed back to the RL decision engine. The one that is used by this agent to improve the learning policy is this reward. The longer he or she is in the job, the more he or she will be able to make this decision the more work there is. This is feedback – feedback that is continuous – and the framework is adaptive. The system is not a standard one, but has flexibility in the decision making process by applying the actual operational experience.

It is possible to report on evaluation layers' performance also. It can provide metrics like, average response time, throughput, resource utilization rate, cost savings, number of SLA violations, scaling frequency, energy efficiency etc. These indicators can provide the administrators with an insight on the performance assessment of RL based Autoscaling framework with the traditional rule based Autoscaling framework.

IV. FRAMEWORK EVALUATION AND PERFORMANCE ASSESSMENT

4.1 Evaluation Approach
The following traditional provisioning approaches will be compared to the proposed Reinforcement Learning (RL) based provisioning approach: Static provisioning, Threshold based provisioning, and Heuristic scheduling: effort should be made to evaluate the approaches in one of the following ways: a simulated cloud or real cloud BI solution with the following types of workloads: dashboard requests, analytical queries, scheduled reports, and high-priority business tasks; and varying levels of demand. This assessment is primarily to see if the RL agent can make better decisions to optimize the performance of the response time, cost, resource utilization and reliability of services.

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

4.2 Performance Metrics

There are a number of quantitative measures of effectiveness of the framework. The first one is query response time – how long it could take the BI system to respond to the query from a user. The response time is the better the analytical performance (the shorter the better). The second is the latency of the dashboard refresh that is applicable to real-time/near real-time BI applications. The third one is resource utilization (such as CPU, Memory, Storage I/O and network). Now looking at constructing an optimization model, one thing to attempt to do is to not "over-model" or to "under-model" the model. The fourth – charges for using a certain amount of resources during a specified period of time, (cloud cost) A fifth rate is 5th (SLA violation rate); how many times SLA violated while the system was running. Some other metrics include throughput, frequency of scaling, energy consumption and workload completion time.

4.3 Comparative Evaluation

RL-based framework should be compared to baseline models, and demonstrated to be useful. This can result in waste of resources and/or delay in provisioning in particular if a Static Provisioning methodology is employed that is workload unaware. They scale up/down based on the thresholds and if it is a threshold based autoscaling, they allocate/dedicate resource based on the thresholds. However, the RL approach could be utilized for more adaptive decisions with knowledge learnt from different workload scenarios. But if it offers less latency, SLA as well as lower costs than the original approach, then it may prove to be more useful for BI workloads that are dynamic in nature.

4.4 Reward Function Evaluation

In addition, it needs to be assessed if the reward function has an impact on the learning behaviour of the agent, because it has a direct influence. Rwdt is NOT a performance/just cost based reward function. Cost of cloud is increased? With overscaling comes impact on the response time – similar to that for scaling up? The same applies to seek to lower costs - it can result into BI query performance below par. Therefore, it is critical to measure and analyze the reward function that should be leveraged to adjust the latency, cost, utilization, SLA and energy efficiency factors.

4.5 Expected Evaluation Outcome

The ultimate goal is to develop a model that will be able to more permanently structure and enhance resources management than the classic models. It should result in a more responsive BI system during periods of heavy workload, in less idle during periods of low workloads and it should provide better service during periods of workload fluctuations. Overall, the evaluation shows that the cloud resources can be adaptatively optimized while considering cost and performance aware of modern Business Intelligence systems with the help of Reinforcement Learning.

V. CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this research article, Reinforcement Learning approach to Dynamically Optimize cloud resources in Business Intelligence environment, is presented. These cloud-based BI systems are constantly under pressure from the constant and often changing volumes of data, as well as other workloads—whether it's changing user behavior, complex analytical queries, real-time dashboards or scheduled reports. Traditional resource management methods such as static provisioning or threshold-based autoscaling, can be difficult in these types of dynamic conditions. They may under provision on resources, causing high latency, SLA violations or over-provision resources and lead to excess cloud spending, and low utilization.

To avoid such shortcomings, it is proposed to add an intelligent learning based decision model. It features five layers – BI workload monitoring, state space construction, reinforcement learning decision engine, cloud resource orchestration and cloud feedback evaluation. Here, the reward signal can be learned, and its effects are reflected on the decision making policy while the cloud workload state, BI workload state and allocation can always be done depending on the result of monitoring. It optimises various objectives, such as query response time, latency of dashboards, resource utilisation, cloud cost, SLA compliance and energy efficiency.

Based on the results of the study it can be concluded that "Reinforcement Learning" is an amazing tool to be used for assisting adaptive and autonomous management of today's BI-platforms. It is not a hard coded rules system, but can flexibly be learnt from the historical workload pattern and the real time workload. Can also scale up and down during peak and low demand periods and even prioritize critical BI workloads, if needed – even run cloud in a cost aware manner. The general model proposed is good when it comes to building up intelligent and responsive cloud infrastructures of BI oriented towards the business.

International Journal of Science, Research and Technology (IJSRAT)
|Bimonthly | Peer-Reviewed | Scholarly Journal|
|ISSN 2454-3853|www.ijsrat.com|editor@ijsrat.com|Vol.9, Issue 1, January-February 2026|
DOI: 10.15662/IJSRAT.2026.0901008

5.2 Future Work

The framework can be further expanded to conduct further research. Secondly, information of enterprise cloud workload could be input into the model, and then the real performance of the model could be tested. Second, the performance of the advanced learning algorithms such as the Deep Reinforcement Learning (DRL) algorithms such as Deep Q-Networks, Actor-Critic and Proximal Policy Optimization (PPO) can be compared for the most efficient learning policy. Thirdly, the future work could be done by using multi-agent reinforcement learning that is multiple agents would coordinate in computing, storage, networking and query scheduling.

Other optimisations could be included, e.g. security aware optimisations, carbon aware optimisations. It is also important to consider availability of renewable energy, workload (data) sensitivity, reduction of carbon emissions etc. factors when building the framework. Finally, using predictive analytics and reinforcement learning can help find the load trends proactively, making cloud-based BI resource management sustainable, reliable and proactive.

REFERENCES

[1] P. Mishra, M. Putrevu, G. R. Ganger, and A. Slominski, “Optimizing Cloud Workloads: Autoscaling with Reinforcement Learning,” *IBM Research*, 2024. [Online]. Available: https://research.ibm.com/publications/optimizing-cloud-workloads-autoscaling-with-reinforcement-learning

[2] Amazon Web Services, “PERF02-BP05: Scale your compute resources dynamically,” *AWS Well-Architected Framework*. [Online]. Available: https://docs.aws.amazon.com/wellarchitected/latest/performance-efficiency-pillar/perf_compute_hardware_scale_compute_resources_dynamically.html

[3] T. Cui, R. Yang, C. Fang, and S. Yu, “Deep reinforcement learning-based resource allocation for content distribution in IoT-edge-cloud computing environments,” *Symmetry*, vol. 15, no. 1, 217, 2023.

[4] K. Saidi and D. Bardou, “Task scheduling and VM placement to resource allocation in cloud computing: challenges and opportunities,” *Cluster Computing*, 2023.

[5] Y. Xu and A. H. Mohammed, “An energy-aware resource management method in cloud-based internet of things using a multi-objective algorithm and crowding distance,” *Telecommunication Systems*, vol. 34, no. 1, e4673, 2023.

[6] M. I. Khaleel, “Hybrid cloud-fog computing workflow application placement: joint consideration of reliability and time credibility,” *Multimedia Tools and Applications*, 2023.

[7] K. K. Gola et al., “Multi-objective hybrid capuchin search with genetic algorithm based hierarchical resource allocation scheme with clustering model in cloud computing environment,” *Concurrency and Computation: Practice and Experience*, vol. 35, no. 7, e7606, 2023.

[8] A. Amini Motlagh, A. Movaghar, and A. M. Rahmani, “A new reliability-based task scheduling algorithm in cloud computing,” *International Journal of Communication Systems*, vol. 35, no. 3, e5022, 2022.

[9] D. Parfenov, I. Bolodurina, L. Kuznetsova, L. Zabrodina, and N. Yanishevskaya, “Application of bioinspired methods for solving the problem of resource allocation in cloud platforms,” in *2021 International Conference on Information Technology and Nanotechnology (ITNT)*, pp. 1–7, IEEE, 2021.

[10] P. Gupta et al., “Hybrid Whale optimization algorithm for resource optimization in cloud e-healthcare applications,” *Computers, Materials & Continua*, vol. 71, no. 3, pp. 5381–5396, 2022.

[11] P. Durgadevi and S. Srinivasan, “Resource allocation in cloud computing using SFLA and cuckoo search hybridization,” *International Journal of Parallel Programming*, vol. 48, pp. 549–565, 2020.

[12] S. K. Kayalvili and M. Selvam, “Hybrid SFLA-GA algorithm for an optimal resource allocation in cloud,” *Cluster Computing*, vol. 22, Suppl. 2, pp. 3165–3173, 2019.