www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 08, August 2024, pp : 950-962

e-ISSN : 2583-1062

Impact Factor : 5.725

DYNAMIC WORKLOAD PREDICTION IN A CHANGING CLOUD ENVIRONMENT AND ALLOCATE RESOURCES IN REAL-TIME

Neha Sharma[1]

[1]School of engineering and technology DPGITM Gurgaon, India.

ABSTRACT

In todays changing world of computing which includes cloud computing, edge computing and distributed systems, managing workloads that have rapid changes, in resource demands has become a critical challenge. To address this challenge it is important to develop scheduling mechanisms that can adapt quickly without putting strain on the underlying systems. This research paper introduces a scheduling mechanism specifically designed to meet these needs. It explores the complexities of workload management examines existing research in this area and identifies a gap; the lack of scheduling mechanisms that can seamlessly handle changing workloads without adding burdens. Our research is driven by the increasing demand for scheduling solutions in evolving computing environments where dynamic resource allocation's crucial for optimizing system performance. Our proposed scheduling approach is based on a crafted algorithm that aims to minimize system impact while remaining highly adaptable to changes in workload patterns. To aid understanding this paper provide an analysis along with a flowchart illustrating how the algorithm operates, as well as tables that provide insights, into how our scheduling system effectively manages various workload scenarios.

Keywords - Cloud Computing; Dynamic Workloads; Resource Allocation Strategies; Task Scheduling)

1. INTRODUCTION

The field of computing is currently experiencing a transformation, characterized by innovation, widespread connectivity and the ever-changing nature of technology. This paper find ourselves in an era where computing has expanded beyond boundaries and now influences every aspect of our lives reshaping how this paper work, communicate, and solve complex problems. In todays computing environments, which are marked by the growing presence of cloud computing, edge computing and distributed systems efficiently managing workloads has become a priority [1-5]. Dynamic workloads The term dynamic workloads is used to describe workload behavior in terms of resource demands over time which is often driven by user activity or application needs or by external influences. As efficient resource use and system throughput are key concerns for modern enterprises, anticipating and reacting to such workload characteristics becomes important for system designers and users alike. In this paper we look into understanding and developing strategies to anticipate workloads and thereby providing means to cope with emerging problems due to dynamic workload behavior in contemporary computing environments [6-11].

In the realm of cloud computing numerous users and applications rely on resources. The key task involves ensuring access, to these resources while maximizing their distribution efficiently. This crucial equilibrium greatly influences the effectiveness and cost efficiency of cloud services. Another layer of complexity is added by Edge computing aiming to process data at the networks edge close to its source. In this context the question arises about how to distribute workloads across a variety of edge devices each with its unique capabilities and limitations [12-16].

Workload prediction has a rich history of research in diverse domains and technologies. Past research has comprehensively studied techniques, models including machine learning, statistical models and hybrid models. These works focused on predicting the patterns and variability of workloads. They explained the advantages and disadvantages of existing schemes and contributed towards understanding of accurately predicting workload variations. Some of them studied the impacts of user behaviour, characteristics of applications and environmental events on workload characteristics. With the emergence of cloud computing, predicting resource demand and optimizing cloud resources to provide services at an optimal cost became of interest. Furthermore real time use cases, such as content delivery networks and edge computing were proposed and prediction models were implemented. In combination, these prior studies constitute the knowledge base that can help understand the challenges and opportunities of workload prediction and also guide us in this research [17-22].

Effective dynamic workload prediction offers benefits across computing environments with significant implications, for optimizing resources and system performance. In cloud computing accurate predictions enable providers to allocate resources striking a balance between performance and cost efficiency through efficient resource utilization. The predictive capabilities of workload models also prove invaluable in real time applications like video streaming and online gaming where anticipating spikes in demand enables scaling to ensure an responsive user experience. In the

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 08, August 2024, pp : 950-962
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 5.725

realm of edge computing, where resources re often limited dynamic workload prediction plays a role in efficiently allocating computing power. This technology applications extend further into the Internet of Things (IoT) enabling resource management in sensor networks and connected devices by forecasting varying workloads. Consequently the ranging applications and advantages of workload prediction highlight its critical role, in enhancing the efficiency cost effectiveness and overall performance of modern computing systems [22-28].

By undertaking this research our aim is to make an impact, in the field of computing. This paper strive to provide knowledge and resources that can help individuals navigate the challenges and possibilities it offers. This paper is organized as follows:

Section I gives an intro, Section 2 provides a Introduction, Section 3 provides a comprehensive literature survey, Section 4 discusses the proposed methodology , section 5 discusses the obtained results & Finally section 6 conclusion & future scope of the presented study..

2. BACKGROUND

A. Adaptive Workload Forecasting Model

A new approach, known as the Auto Adaptive Neural Network (AANN) has been introduced to forecast workload, in dynamic cloud environments. To achieve this an enhanced learning algorithm called Auto Adaptive Differential Evolution (AADE) has been developed and applied to train a feed forward network for workload prediction. The AADE training algorithm dynamically optimizes neuron connections and aids in learning workload patterns from data. The effectiveness of the proposed model has been evaluated using two datasets; NASA and Saskatchewan traces. The results demonstrate a reduction in root mean error and improved prediction accuracy compared to existing methods [29-36].

The key contributions of this research can be summarized as follows;

- Introducing a network model specifically designed for workload forecasting in dynamic cloud environments.
- Applying an version of the differential evolution learning algorithm to train the network effectively.
- Implementing adaptation across three dimensions; mutation, crossover and control parameter tuning thereby achieving optimization of the network model.
- Conducting experimental analysis of the proposed workload prediction approach using two datasets; NASA HTTP and Saskatchewan HTTP internet traffic traces.
- The findings indicate enhancements in predicting workload demands, at cloud data centers.

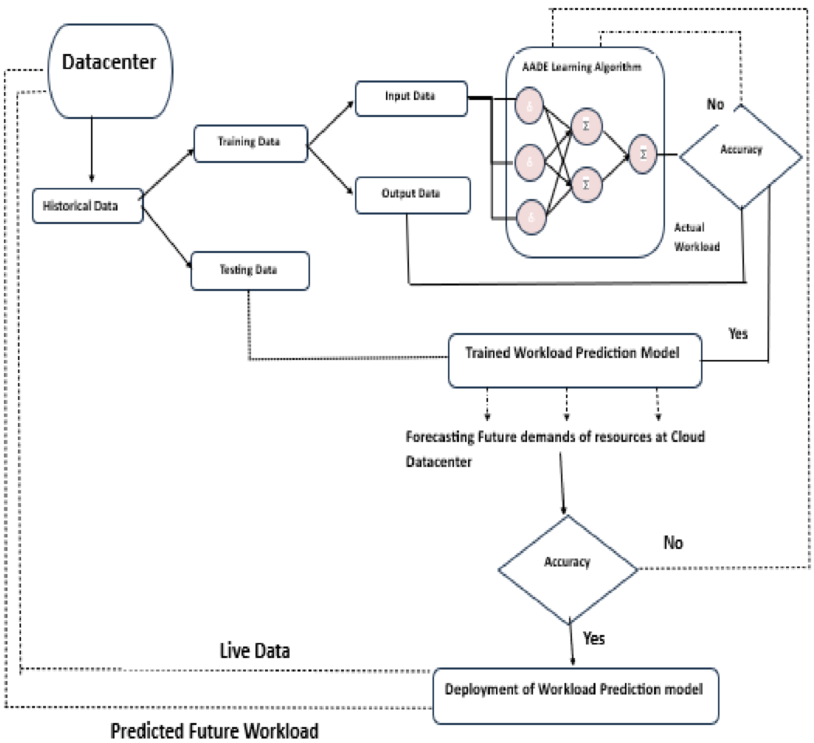

Figure 1. Proposed Workload Forecasting model

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 08, August 2024, pp : 950-962
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 5.725

3. LITERATURE SURVEY

The area of workload scheduling for changing workloads has received attention from both researchers and practitioners. Researchers have. Implemented scheduling mechanisms to tackle the challenges posed by dynamic workloads. In this section this paper will. Analyze the existing research in this field highlighting contributions, limitations, and areas where research gaps are apparent [37-44].

Scheduling Mechanism	Characteristics	Limitations
Static Scheduling Strategies	- Based on fixed configs -Historical workload data	-Inflexible for dynamic workloads -Poor adaptability to changes
Dynamic Scheduling Algorithms	-Real-time allocation - Adapt to changing loads -Examples: Round Robin, Weighted Fair Queuing fine-tuning, SJN algorithm	- May introduce computational overhead, limiting scalability - Limited granularity for resource allocation, - Efficiency and scalability challenges in large-scale deployments
Machine Learning-Based approach [26-31]	-Leverage historical data -Predictive models -Reinforcement learning, -deep learning	Require extensive training and may not adapt rapidly to unforeseen workload changes

3.1 Strategies for Workloads;

Initially efforts in workload scheduling relied on strategies. These strategies allocated resources based on predetermined configurations and historical workload patterns. Although suitable for workloads they struggled to effectively adapt to dynamic changes. The limitations of these strategies became evident in scenarios where workloads experienced variations, such as online services that faced fluctuating user demand.

Table 2 describes Strategies for Workloads

Scheduling Approach	Adaptability to Dynamic Workloads	Overhead	Scalability	Resource Efficiency
Static Scheduling	Limited	Low	Good	Poor
Historical Workload Data	Moderate	Low	Good	Moderate

3.2 Algorithms for Dynamic Scheduling

Recognizing the need for solutions researchers have proposed dynamic scheduling algorithms [45]. These algorithms aim to allocate resources based on real time characteristics of the workload providing flexibility in resource management. Examples include heuristics like Round Robin, Weighted Fair Queuing as more sophisticated approaches like the Shortest Job Next (SJN) algorithm. While these dynamic algorithms offer adaptability they often introduce overhead which limits their scalability and efficiency, in large scale environments.

Table 3 describes various Algorithms for Dynamic Scheduling

Scheduling Approach	Adaptability to Dynamic Workloads	Overhead	Scalability	Resource Efficiency
Round Robin	Limited	Low	Good	Poor
Weighted Fair Queuing	Moderate	Moderate	Moderate	Moderate
Shortest Job Next (SJN)	Moderate	Moderate	Moderate	Moderate
Real-time Monitoring	Good	High	Limited	Moderate

3.3 Approaches Based on Machine Learning

Workload scheduling has been investigated using machine learning techniques , such, as reinforcement learning and deep learning. These methods utilize data and predictive models to determine how resources should be allocated. Although they show potential in optimizing resource usage they necessitate training. May not quickly adapt to unexpected changes, in the workload [46-57].

www.ijprems.com
editor@ijprems.com
INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 08, August 2024, pp : 950-962
e-ISSN : 2583-1062
Impact Factor : 5.725

Table 4 describes Approaches Based on Machine Learning

Scheduling Approach	Adaptability to Dynamic Workloads	Overhead	Scalability	Resource Efficiency
Reinforcement Learning	Good	High	Limited	Good
Deep Learning	Good	High	Limited	Good
Predictive Models	Good	Moderate	Moderate	Moderate

4. RESEARCH ANALYSIS

4.1 Research Gap

There is a gap, in the research on creating models that can adapt to changes in computing environments without using too much computational power. This study aims to address this gap by introducing a mechanism, for predicting workloads. Existing solutions often have limitations that hinder their practicality in computing environments;

- Overhead and Scalability; Many dynamic scheduling mechanisms result in overhead especially when dealing with large scale deployments. This additional burden can negate the advantages of scheduling. Restrict the effectiveness of these solutions in resource constrained environments [58-62].
- Adaptability; While certain existing approaches demonstrate adaptability to changes in workloads they may lack the granularity to adjust resource allocation. Striking the balance between adaptability and efficiency remains a challenge [63-65].
- Resource Efficiency; In evolving workloads there is a need for scheduling mechanisms that not adapt to changes but also optimize resource utilization. Numerous current solutions prioritize adaptability over resource efficiency resulting in underutilized or provisioned resources [66-70].
- Real time Decision Making; Workloads in computing environments can undergo changes necessitating real time decision making capabilities. Some existing solutions may not respond enough to spikes or drops, in workload demand.

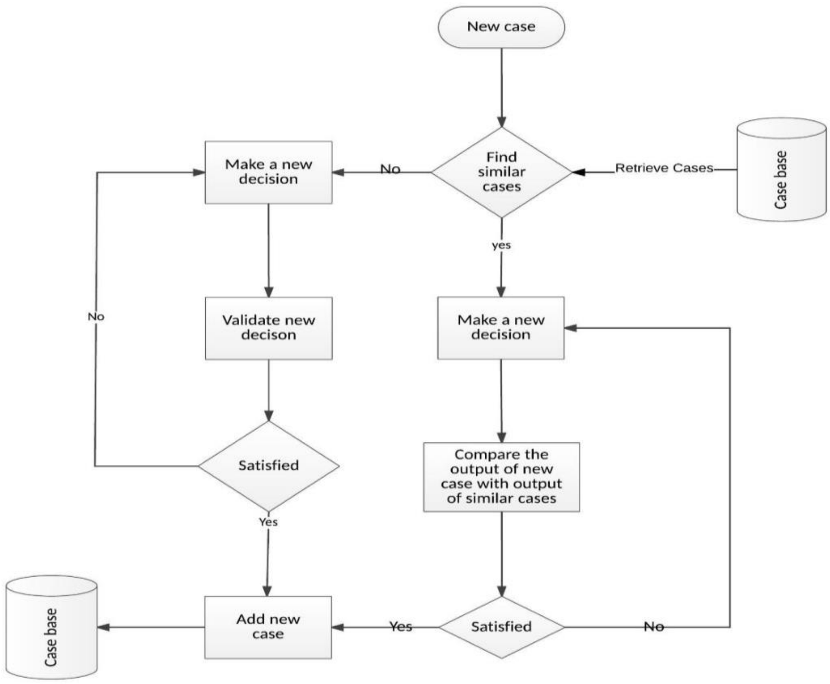

Figure 2 Flow chart of the process of cloud adoption decision model

4.2. Motivation for the Research:

The reason, for conducting this research lies in the growing need for scheduling methods that can effectively handle the difficulties brought on by changing workloads. In todays environment, businesses and service providers are looking to make the most of their resources cut down on operational expenses and improve system performance. Being able to allocate resources dynamically has evolved from being an obstacle, to becoming a crucial strategic requirement.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 08, August 2024, pp : 950-962
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 5.725

Our motivation stems from the desire to bridge the divide, between scheduling methods and an ideal solution that combines flexibility with additional workload. This paper acknowledge the need for a scheduling system that can effectively allocate resources in time while keeping computational demands to a minimum. By addressing this necessity our research endeavors to enhance the efficiency of workload scheduling and resource management, in computing environments [71].

5. PROPOSED APPROACH

To effectively address the difficulties that come with fluctuating workloads we have devised a scheduling mechanism in this paper. The primary objective of this mechanism is to minimize system impact while simultaneously accommodating the shifting patterns of workloads. Our approach integrates adaptability, resource efficiency and real time decision making to cater to the demands of computing environments [44-45]. The development in IoT has been influenced by energy constrained on sensor ndoes [72-75].

5.1. Basic Algorithm

The basic Allocation of Resources Based on Real Time Workload consists of following steps:-

Input- Real time collection of workload data such, as CPU usage, memory requirements and network traffic.

- Historical workload data for analyzing trends.
- Availability of system capacity and resources.

Output- Decisions on resource allocation for each workload.

Step 1- Data Collection

- Continuously gather real time workload data from all applications and resources.
- Keep a record of data for trend analysis. Recognizing patterns.

Step 2- Pattern Recognition

- Analyze data to identify patterns and trends in workloads.
- Employ methods and machine learning techniques to detect both irregular patterns.

Step 3- Prediction

- Predict behavior of workloads based on patterns.
- Utilize models to estimate the resource needs for time intervals.

Step 4- Monitoring Resource Pool

- Continuously monitor the system resources such as CPU, memory, storage and network bandwidth.

Step 5- Decision Making

- For each decision point regarding resource allocation assess the real time workload patterns.
- Allocate resources to workloads based on anticipated resource requirements.
- Implement resource allocation policies using insights, from predictions (auto scaling, provisioning or load balancing).

Step 6- Executing Resource Allocation

- Put the resource allocation decisions into action, in time.
- Make adjustments to resource allocation as workload patterns change.

Step 7- Continuous Feedback Loop

- Consistently monitor the system and workloads.
- Collect performance data to assess how effective the resource allocation is.

Step 8- Adaptation

- Modify the resource allocation policies and models based on feedback and changing workload patterns.
- Improve accuracy by retraining models

Step 9- time Reporting and Alerts

- Deliver real time reports on resource utilization and allocation.
- Raise alerts for patterns or shortages of resources.

Step 10- Performance Optimization

- Continuously optimize resource allocation to enhance system performance cost efficiency and satisfaction with workloads.

Step 11- End of Cycle

- Repeat Steps 1 to 10, in a cycle to ensure that real time resource allocation remains optimal and adaptable.

At the heart of our proposed approach is a crafted scheduling algorithm. This algorithm makes real time decisions, about resource allocation based on the state of workloads taking into account workload patterns for predictive insights

5.2. Proposed Algorithm (Dynamic Workload Aware Resource Allocator DWARA)

Objective; The aim is to allocate resources in a cloud environment by predicting varying workloads. This optimization process focuses on enhancing performance and resource utilization.

Steps of the Algorithm;

1. Workload Prediction Module;

   Develop a module that accurately forecasts workloads using historical data, machine learning or statistical models. The module should be adaptable and capable of self learning to adapt to changing patterns.

2. Resource Monitoring;

   Continuously monitor the resource utilization, in the cloud environment by capturing real time data on CPU availability, memory usage, storage capacity and network bandwidth.

3. Dynamic Threshold Setting;

   Dynamically set resource utilization thresholds based on predicted workloads. Establish levels of resource usage for types of workloads to optimize performance and avoid contention.

4. Workload Classification;

   Categorize workloads into classes based on their resource requirements, priority levels and predicted duration. This classification enables a adaptable resource allocation strategy.

5. Adaptive Resource Allocation;

   Allocate resources based on predicted workloads while adjusting thresholds dynamically. Prioritize workloads or those with priority levels by allocating resources, in real time to meet changing demands.

6. Load Balancing;

   To optimize system performance it is important to implement mechanisms that evenly distribute workloads, across resources. This prevents any resource from becoming overloaded and causing bottlenecks.

7. Feedback Mechanism;

   A feedback loop should be established to continuously evaluate the accuracy of workload predictions compared to resource usage. This feedback can then be used to refine and improve the prediction model over time making the algorithm more adaptable.

8. Resource Reservation for Workloads;

   During peak demand periods it is crucial to identify and reserve resources for critical workloads. By doing essential processes can receive the resources without contention or interference from other tasks.

9. Real time Adaptation;

   Enable the ability to adjust resource allocations in time, as the workload changes. This involves releasing resources that're no longer necessary and reallocating them to address the needs of emerging workloads.

10. Considerations for Security and Privacy;

security and privacy measures into the algorithm ensuring that sensitive information is handled securely and that controlled access, to resources is based on predefined policies.

Performance Metrics:

- Resource Utilization Efficiency: Evaluate how effectively resources are utilized.
- Workload Prediction Accuracy: Assess the precision of the workload prediction module.
- Adaptability: Evaluate the algorithm's ability to adapt to changing workload patterns.
- Response Time: Measure the time taken to allocate resources in response to workload changes.
- Security Compliance: Assess the algorithm's adherence to security and privacy policies.

5.3. Architecture

Our scheduling mechanism is designed to integrate with existing computing environments and consists of key components;

- Resource Monitor; This component continually monitors the resource utilization of the computing environment gathering data, on CPU, memory, storage and network usage.
- Workload Analyzer; The workload analyzer evaluates the characteristics of the workload such as task types their resource requirements and priority. It also utilizes workload data to identify patterns and trends.
- Scheduling Engine; The scheduling engine serves as the core of our mechanism. It processes data from the resource monitor and workload analyzer in time to make allocation decisions. It employs algorithms that balance adaptability and efficiency.
- Resource Allocator; The resource allocator executes the allocation decisions made by the scheduling engine to ensure resources are provisioned or de provisioned as required.

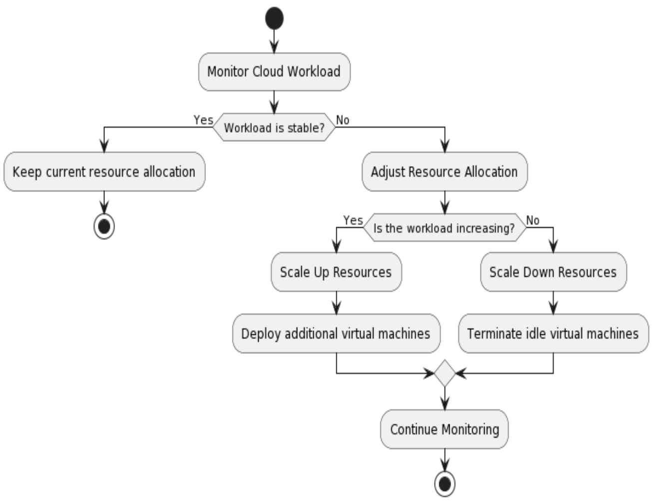

Figure 3- This flowchart provides a representation of how the algorithm adapts to varying workload conditions.

Resource Allocation

Resource allocation involves distributing resources to tasks or components, within a system with the goal of optimizing performance and meeting specific objectives. The approach can vary depending on the context whether its in computing environments, project management or manufacturing. Here are some general methods used for resource allocation;

- **Planning**; In this approach a single authority or system manages the distribution of resources. While it can be efficient for coordination purposes there may be challenges in terms of scalability and responsiveness to changes
- **Distributed Allocation;** Resources are independently assigned by units or components. This approach offers flexibility and adaptability to changing conditions. May require coordination mechanisms.
- **Priority Based Allocation;** Assigning priorities to tasks or processes ensures that high priority items receive resources first. This method is commonly used in real time systems where certain tasks require attention.
- **Dynamic Allocation;** Resources are allocated based on real time demand and workload. Dynamic allocation adapts to changes in the systems environment. Optimizes resource utilization.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 08, August 2024, pp : 950-962
e-ISSN : 2583-1062
Impact Factor : 5.725

- **Load Balancing;** Tasks are evenly distributed across resources to prevent bottlenecks and optimize system performance. Load balancing is crucial, in distributed computing and cloud environments.
- **Reservation of Resources;** allocate resources beforehand, for important tasks. This guarantees that vital processes receive the required resources without any conflicts.
- **Allocation based on Queues;** Tasks or processes are lined up in a queue and resources are assigned according to the order in the queue. This approach is commonly used when tasks need to wait for resources.
- **Use of Optimization Algorithms**; Implement algorithms that analyze system parameters, workload and resource availability to determine the strategy for allocating resources. This may involve techniques such, as linear programming, genetic algorithms or other optimization methods.

6. IMPLEMENTATION AND RESULTS

Details of the Implementation;

To ensure an integration of our scheduling mechanism, into computing environments this paper have taken into consideration components and factors. Here is a comprehensive account of our implementation process;

B. Software Elements;

- Incorporating the Scheduling Algorithm; This paper have integrated our scheduling algorithm into the existing software infrastructure. This integration involves making modifications or extensions to the software stack so that it can effectively utilize time decision making capabilities.
- Monitoring Tools for Resources; This paper utilize tools and agents for collection of data on CPU usage, memory utilization, network traffic and storage I/O. This data is then analyzed by the scheduling mechanism.
- Gathering Workload Characteristics; To make decisions about resource allocation this paper gather information about the types of tasks or processes running in the environment their resource requirements and their priority levels. This data plays a role in determining how resources should be allocated.
- Repository for Historical Data; This paper maintain a repository that holds workload data. This repository serves as a reference point for the workload analyzer component to identify trends and patterns, in workload behavior.

C. Hardware Considerations;

- Resource Pooling; This paper set up the hardware infrastructure to support resource pooling, which allows us to dynamically allocate and deallocate resources as needed. This can involve using virtualization technologies or containerization to ensure that resources are isolated.
- Scaling Mechanisms; To handle scalability this paper incorporate mechanisms, for both vertical scaling. This ensures that our infrastructure can adapt to workload sizes and resource demands.

D. Real time Decision Making;

- Continuous Monitoring; Our scheduling mechanism constantly monitors resource utilization and workload characteristics in time. This ongoing monitoring is crucial for making decisions, about resource allocation.
- Adaptive Algorithms; This paper employ adaptive scheduling algorithms that take into account both conditions and historical data. These algorithms consider workload trends. Adjust resource allocations accordingly.

7. CONTRIBUTIONS

Our research has made contributions, to the field of workload scheduling with several key findings to highlight.

- Minimizing Overhead; Our proposed scheduling mechanism aims to strike a balance, between adaptability and resource efficiency resulting in computational burden. By allocating resources in time our approach effectively reduces unnecessary waste of computing resources and mitigates any impact on system performance.
- Flexibility; This paper have developed an adaptive scheduling algorithm that can swiftly respond to changing workload conditions. Leveraging real time data and historical patterns our algorithm optimizes decisions regarding resource allocation ensuring that resources are directed where they are most needed.
- Real time Decision Making; Our mechanism excels in making decisions, which's crucial in modern computing environments where workloads can change rapidly. By monitoring resource utilization and workload characteristics our approach ensures informed allocation decisions at the moment.

- Scalability; Through testing and benchmarking this paper have demonstrated the scalability of our solution. It performs efficiently both in scale and larger scale environments accommodating a range of workloads and varying resource demands.

- Transparency and Visualization; To enhance transparency and facilitate comprehension this paper have employed aids such, as flowcharts, tables, graphs and charts to illustrate how our scheduling mechanism operates and performs.

8. CONCLUSION

In conclusion our research paper presents a solution to address the complexities of managing evolving workloads. Through minimizing prioritizing adaptability and leveraging time decision making capabilities our scheduling mechanism holds great promise in enhancing resource allocation and system performance in modern computing environments [76-79].

This paper firmly believe that the contributions outlined in this research paper have the potential to significantly impact workload scheduling practices by enabling organizations to optimize resource utilization reduce costs and improve system performance within the dynamic landscape of modern computing environments.

As technology progresses there will continue to be a growing demand, for scheduling mechanisms that can effectively handle evolving workloads. Our research provides a basis, for investigation, in this important field aiming to push the boundaries of workload scheduling and resource management techniques to new heights [80-83].

9. FUTURE WORK

Our research represents an advancement, in workload scheduling. There are several areas that could be explored in the future;

- Incorporating Machine Learning; It would be beneficial to conduct research on integrating machine learning techniques to enhance capabilities and optimize adaptability.
- Exploring Auto scaling Strategies; It would be worth investigating auto scaling strategies that can dynamically adjust computing resource capacity based on changes, in workload. This could result in utilization of resources.
- Energy Efficiency; To further enhance our research this paper can explore ways to optimize workload scheduling not for performance but for energy efficiency, in line with sustainability objectives.
- Cloud Implementations; By adapting our scheduling mechanism to suit native environments and serverless computing architectures this paper can unlock new possibilities for improving resource allocation efficiency.
- Industry Adoption; Encouraging the adoption of scheduling mechanisms in the industry remains a challenge. Future efforts should focus on strategies for integrating our approach into real world computing environments.

ACKNOWLEDGMENT

We would like to express our appreciation to Ms. Shraddha Arora, for their invaluable guidance, expertise, and unwavering support throughout the course of this research.

We are truly grateful for the feedback received from reviewers and colleagues during the peer review process. We would also like to extend our thanks to the institutions and organizations that provided resources and funding.

To our families and friends, we extend a heartfelt thank you for your understanding, patience, and encouragement. Your unwavering support has been a source of inspiration on this journey. This research wouldn't have been possible without the collective efforts of everyone mentioned.

10. REFERENCES

[1] Lamba, M., Munjal, G., Gigras, Y., & Kumar, M. (2023). Breast cancer prediction and categorization in the molecular era of histologic grade. Multimedia Tools and Applications, 82(19), 29629-29648.

[2] Arora, S., & Chatterjee, N. (2022, December). Automated Evaluation Approach for Time-Dependent QA Pairs on Web Crawler-Based QA System. In 2022 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI) (pp. 1-6). IEEE.

[3] Lamba, M., Munjal, G., & Gigras, Y. (2018). Feature Selection of Micro-array expression data (FSM)-A Review. Procedia computer science, 132, 1619-1625.

[4] Pandey, M., Arora, M., Arora, S., Goyal, C., Gera, V. K., & Yadav, H. (2023). AI-based Integrated Approach for the Development of Intelligent Document Management System (IDMS). Procedia Computer Science, 230, 725-736.

[5] Lamba, M., Gigras, Y., & Dhull, A. (2021). Classification of plant diseases using machine and deep learning. Open Computer Science, 11(1), 491-508.

[6] Arora, S., Pandey, M., Arora, M., Gupta, K., Sharma, V., & Nagpal, L. (2024). Digitization of Health Insurance Documents for The Cashless Claim Settlement Using Intelligent Document Management System. Procedia Computer Science, 235, 1319-1331.

[7] Lamba, M., Munjal, G., & Gigras, Y. (2021). A hybrid gene selection model for molecular breast cancer classification using a deep neural network. International Journal of Applied Pattern Recognition, 6(3), 195-216.

[8] Chaudhary, S., Indu, S., & Chaudhury, S. (2018). Video-based road traffic monitoring and prediction using dynamic Bayesian networks. IET Intelligent Transport Systems, 12(3), 169-176.

[9] Lamba, M., Munjal, G., & Gigras, Y. (2023). Ranking of classification algorithm in breast Cancer based on estrogen receptor using MCDM technique. International Journal of Information Technology & Decision Making, 22(02), 803-827.

[10] Roy, M., Boby, R. A., Chaudhary, S., Chaudhury, S., Roy, S. D., & Saha, S. K. (2016, October). Pose estimation of texture-less cylindrical objects in bin picking using sensor fusion. In 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 2279-2284). IEEE.

[11] Lamba, M., Munjal, G., & Gigras, Y. (2021). A MCDM-based performance of classification algorithms in breast cancer prediction for imbalanced datasets. International Journal of Intelligent Engineering Informatics, 9(5), 425-454.

[12] Hayat, A. A., Chaudhary, S., Boby, R. A., Udai, A. D., Roy, S. D., Saha, S. K., & Chaudhury, S. (2022). Vision based identification and force control of industrial robots (Vol. 404, pp. 1-178). Springer.

[13] Lamba, M., Munjal, G., & Gigras, Y. (2023). Computational studies in breast Cancer. Research Anthology on Medical Informatics in Breast and Cervical Cancer, 434-456.

[14] Sharma, A., Chaudhary, S., Roy, S. D., & Chand, P. (2015, December). Three dimensional reconstruction of cylindrical pellet using stereo vision. In 2015 Fifth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG) (pp. 1-4). IEEE.

[15] Lamba, M., Munjal, G., & Gigras, Y. (2022). Supervising healthcare schemes using machine learning in breast Cancer and internet of things (SHSMLIoT). Internet of Healthcare Things: Machine Learning for Security and Privacy, 241-263.

[16] Pandey, M., Arora, M., Arora, S., Goyal, C., Gera, V. K., & Yadav, H. (2023). AI-based Integrated Approach for the Development of Intelligent Document Management System (IDMS). Procedia Computer Science, 230, 725-736.

[17] Lamba, M., Munjal, G., & Gigras, Y. (2023). Identifying breast cancer molecular class using integrated feature selection and deep learning model. International Journal of Bioinformatics Research and Applications, 19(1), 19-42.

[18] Arora, S., Pandey, M., Arora, M., Gupta, K., Sharma, V., & Nagpal, L. (2024). Digitization of Health Insurance Documents for The Cashless Claim Settlement Using Intelligent Document Management System. Procedia Computer Science, 235, 1319-1331.

[19] Lamba, M., Munjal, G., & Gigras, Y. (2020). Computational studies on breast cancer analysis. Journal of Statistics and Management Systems, 23(6), 999-1009.

[20] Kumar, P., Arora, S., & Arora, S. (2023). Defending Against Adversarial Attacks in Text-based Fake News Detection: A Machine Learning Survey1.

[21] Lamba, M., Munjal, G., & Gigras, Y. (2024). Two-Stage Gene Selection Technique For Identifying Significant Prognosis Biomarkers In Breast Cancer. International Journal of Computing and Digital Systems, 16(1), 85-100.

[22] Kamboj, D., Vashisth, S., & Arora, S. (2024). A Novel Approach for Traffic Sign Detection: A CNN-Based Solution for Real-Time Accuracy. Int. J. Intell. Syst. Appl. Eng, 12, 400-409.

[23] Lamba, M., Munjal, G., & Gigras, Y. (2021). ECABC: Evaluation of classification algorithms in breast cancer for imbalanced datasets. In Data Driven Approach Towards Disruptive Technologies: Proceedings of MIDAS 2020 (pp. 379-388). Springer Singapore.

[24] Lamba, M., Munjal, G., & Gigras, Y. (2024). IBCBML: interpreting breast cancer biomarker using machine learning. Health and Technology, 1-22.

[25] Arora, S., & Lamba, M. (2024). 3D brain image based tumor classification using ensemble of reinforcement transfer-based belief neural networks. Multimedia Tools and Applications, 1-28.
[26] Lamba, M., Munjal, G., & Gigras, Y. (2023, August). Prediction of Breast Cancer Grade Using Explainable Machine Learning. In International conference on soft computing for problem-solving (pp. 733-745). Singapore: Springer Nature Singapore.
[27] Lamba, M., & Munjal, G. (2024). SLRRT: Sign Language Recognition in Real Time. Supervised and Unsupervised Data Engineering for Multimedia Data, 1-25.
[28] Bhateja, N., Sethi, N., & Kumar, D. (2018, December). Study of ant colony optimization technique for coalition formation in multi agent systems. In 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET) (pp. 1-4). IEEE.
[29] Sethi, N., Bajaj, S. B., & Jaglan, V. (2021, December). Multi-Operator based Saliency Detection. In 2021 International Conference on Computational Performance Evaluation (ComPE) (pp. 507-512). IEEE.
[30] Sethi, N., & Bajaj, S. B. Neural network based image object detection and tracking for security and surveillance.
[31] Malhotra, A., & Sethi, N. (2023). Conceding Sentiment Prognosis on Twitter Data. Journal of Computers, Mechanical and Management.
[32] Sethi, N., Kaushal, S., & Bhateja, N. (2021). Hand Gesture Detection Using Segmentation. International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN, 2347-5552.
[33] Sethi, N., Bajaj, S. B., & Jaglan, V. (2021). Distortion Free Image Generation. Grenze International Journal of Engineering & Technology (GIJET), 7(1).
[34] Sethi, N., Bajaj, S. B., Verma, J. K., & Shrivastava, U. (2021). Google Stock Movement: A Time Series Study Using LSTM Network. In Multidisciplinary Functions of Blockchain Technology in AI and IoT Applications (pp. 70-87). IGI Global.
[35] Sethi, M. N., & Malhotra, A. (2023). Efficiency Engine: Designing and Implementing an Academic Management System. International Journal of Innovative Research in Engineering & Management, 10(3), 115-120.
[36] Bhateja, N., Sethi, N., Jain, S., & Mishra, Y. (2020). Lawn mower–An automated machine. International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN, 2347-5552.
[37] Juneja, T., Bajaj, S. B., & Sethi, N. (2022, May). Synthetic Time Series Data Generation Using Time GAN with Synthetic and Real-Time Data Analysis. In The International Conference on Recent Innovations in Computing (pp. 657-667). Singapore: Springer Nature Singapore.
[38] Sethi, N., Bhateja, N., Singh, J., & Mor, P. (2020, March). Fibonacci retracement in stock market. In 5th International conference on next generation computing technologies (NGCT-2019).
[39] Bhateja, N., Sethi, N., & Kaushal, S. (2021). Machine learning and its role in diverse business systems. Research Journal of Science and Technology, 13(3), 213-217.
[40] Sethi, N., Bhateja, N., Kaushal, S., & Singh, J. (2020). Launching WordPress-Blogs and Domains. International Journal of Innovative Research in Computer Science & Technology (IJIRCST), ISSN, 2347-5552.
[41] Sethi, N., Bhateja, N., Kaushal, S., & Singh, J. (2020). Launching WordPress-Blogs and Domains. International Journal of Innovative Research in Computer Science & Technology (IJIRCST), ISSN, 2347-5552.
[42] Sethi, N., Bhateja, N., Sethi, N., & Sinha, S. (2020). Recognizing sentiment prediction on twitter data. International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN, 2347-5552.
[43] Sethi, N. A Comprehensive Study on Data Science and its Practical Applications. International Journal on Recent and innovation trends in computing and communication, (5), 573-575.
[44] Sethi, N. (2017). IoT-Architecture and its Technical Issues. International Journal of Advanced Research in Computer Science, 8(5).
[45] Kaushal, S., Narang, A., Sethi, N., & Bhateja, N. (2021). GST calculation for invoice generation in IT filing. Research Journal of Science and Technology, 13(3), 173-176.
[46] Kaushal, S., Narang, A., Sethi, N., & Bhateja, N. (2021). GST calculation for invoice generation in IT filing. Research Journal of Science and Technology, 13(3), 173-176.
[47] Sethi, N., & Bhateja, N. A STUDY ON ENVIRO-COMPUTING: KEY CHALLENGES.
[48] Raghuvanshi, T., Chaudhary, S., Jain, V., Agarwal, S., & Chaudhury, S. (2015, July). Automated monocular vision based system for picking textureless objects. In Proceedings of the 2015 Conference on Advances In Robotics (pp. 1-6).

[49] Hayat, A. A., Chaudhary, S., Boby, R. A., Udai, A. D., Dutta Roy, S., Saha, S. K., & Chaudhury, S. (2022). Force Control and Assembly. In Vision Based Identification and Force Control of Industrial Robots (pp. 115-151). Singapore: Springer Singapore.

[50] Chaudhary, S., Zakhami, S., & Roy, S. D. (2019, July). Visual Feedback based Trajectory Planning to Pick an Object and Manipulation using Deep learning. In Proceedings of the 2019 4th International Conference on Advances in Robotics (pp. 1-6).

[51] R. Dogra, S. Rani, B. Sharma, and S. Verma, "Essence of scalability in wireless sensor network for smart city applications," in IOP Conference Series: Materials Science and Engineering, IOP Publishing, 2021, p. 012094. Accessed: Jul. 28, 2024. [Online]. Available: https://iopscience.iop.org/article/10.1088/1757-899X/1022/1/012094/meta

[52] R. Dogra, S. Rani, B. Sharma, S. Verma, D. Anand, and P. Chatterjee, "A novel dynamic clustering approach for energy hole mitigation in Internet of Things-based wireless sensor network," Int. J. Commun. Syst., vol. 34, no. 9, p. e4806, Jun. 2021, doi: 10.1002/dac.4806.

[53] R. Dogra, S. Rani, S. Verma, S. Garg, and M. M. Hassan, "TORM: Tunicate Swarm Algorithm-based Optimized Routing Mechanism in IoT-based Framework," Mob. Netw. Appl., vol. 26, no. 6, pp. 2365–2373, Dec. 2021, doi: 10.1007/s11036-021-01833-2.

[54] V. G. Menon, S. Verma, S. Kaur, and P. S. Sehdev, "Internet of Things-Based Optimized Routing and Big Data Gathering System for Landslide Detection," Big Data, vol. 9, no. 4, pp. 289–302, Aug. 2021, doi: 10.1089/big.2020.0279.

[55] D. Pant, S. Verma, and P. Dhuliya, "A study on disaster detection and management using WSN in Himalayan region of Uttarakhand," in 2017 3rd International conference on advances in computing, communication & automation (ICACCA)(Fall), IEEE, 2017, pp. 1–6. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8344703/

[56] M. J. Piran, S. Verma, V. G. Menon, and D. Y. Suh, "Energy-Efficient Transmission Range Optimization Model for WSN-Based Internet of Things.," Comput. Mater. Contin., vol. 67, no. 3, 2021, Accessed: Jul. 28, 2024. [Online].

[57] S. R. Pokhrel, S. Verma, S. Garg, A. K. Sharma, and J. Choi, "An efficient clustering framework for massive sensor networking in industrial Internet of Things," IEEE Trans. Ind. Inform., vol. 17, no. 7, pp. 4917–4924, 2020.

[58] S. Sharma, V. M. Mishra, M. M. Tripathi, S. Verma, and S. Kaur, "SEEDI: Sink-mobility based Energy Efficient Data dissemination in Internet of Medical Things," IEEE Sens. J., 2024, Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10555519/

[59] S. Verma, "Energy-efficient routing paradigm for resource-constrained Internet of Things-based cognitive smart city," Expert Syst., vol. 39, no. 5, p. e12905, Jun. 2022, doi: 10.1111/exsy.12905.

[60] S. Verma and A. Adhya, "Routing in UAVs-assisted 5G wireless sensor network: Recent advancements, challenges, research GAps, and future directions," in 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), IEEE, 2022, pp. 422–428. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9853082/

[61] S. Verma and A. Adhya, "DAAPEO: Detect and Avoid Path Planning for UAV-Assisted 5G Enabled Energy-Optimized IoT," in IEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2023, pp. 1–6. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10225891/

[62] S. Verma, S. Akhtar, A. Adhya, and V. G. Menon, "Towards energy-efficient UAV-assisted 5G internet of underwater things," in Proceedings of the 5th International ACM Mobicom Workshop on Drone Assisted Wireless Communications for 5G and Beyond, Sydney NSW Australia: ACM, Oct. 2022, pp. 1–6. doi: 10.1145/3555661.3560855.

[63] S. Verma, S. Bhatia, S. Zeadally, and S. Kaur, "Fuzzy-based techniques for clustering in wireless sensor networks (WSNs): Recent advances, challenges, and future directions," Int. J. Commun. Syst., vol. 36, no. 16, p. e5583, Nov. 2023, doi: 10.1002/dac.5583.

[64] S. Verma and S. Kaur, "Toward Green and Secure Communication in IoT-Enabled Maritime Transportation System," IEEE Internet Things Mag., vol. 6, no. 2, pp. 32–36, 2023.

[65] S. Verma, S. Kaur, A. Adhya, G. Kaddoum, and B. Brik, "CROP: Cluster-based routing using optimized framework for IoT-based precision agriculture," in ICC 2023-IEEE International Conference on Communications, IEEE, 2023, pp. 4725–4730. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10279747/

[66] S. Verma, S. Kaur, G. Dhiman, and A. Kaur, "Design of a novel energy efficient routing framework for wireless nanosensor networks," in 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), IEEE, 2018, pp. 532–536. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8703308/

[67] S. Verma, S. Kaur, S. Garg, A. K. Sharma, and M. Alrashoud, "AGRIC: Artificial Intelligence-Based Green Routing for Industrial Cyber-Physical System Pertaining to Extreme Environment," IEEE Internet Things J., 2023, Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10256079/

[68] S. Verma, S. Kaur, and A. D. Gupta, "A Brief Survey on Simulators for Designing Routing Protocol for Wireless Sensor Network," in 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE, 2023, pp. 1–6. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10307096/

[69] S. Verma, S. Kaur, R. Manchanda, and D. Pant, "Essence of blockchain technology in wireless sensor network: a brief study," in 2020 International Conference on Advances in Computing, Communication & Materials (ICACCM), IEEE, 2020, pp. 394–398. Accessed: Jul. 28, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9212970/

[70] S. Verma, S. Kaur, D. B. Rawat, C. Xi, L. T. Alex, and N. Z. Jhanjhi, "Intelligent framework using IoT-based WSNs for wildfire detection," IEEE Access, vol. 9, pp. 48185–48196, 2021.

[71] S. Verma, S. Kaur, A. K. Sharma, A. Kathuria, and M. J. Piran, "Dual sink-based optimized sensing for intelligent transportation systems," IEEE Sens. J., vol. 21, no. 14, pp. 15867–15874, 2020.

[72] S. Verma, N. Sood, and A. K. Sharma, "Cost-effective cluster-based energy efficient routing for green wireless sensor network," Recent Adv. Comput. Sci. Commun. Former. Recent Pat. Comput. Sci., vol. 14, no. 4, pp. 1040–1050, 2021.

[73] S. Verma, N. Sood, and A. K. Sharma, "Design of a novel routing architecture for harsh environment monitoring in heterogeneous WSN," IET Wirel. Sens. Syst., vol. 8, no. 6, pp. 284–294, Dec. 2018, doi: 10.1049/iet-wss.2018.5025.

[74] S. Verma, N. Sood, and A. K. Sharma, "Genetic algorithm-based optimized cluster head selection for single and multiple data sinks in heterogeneous wireless sensor network," Appl. Soft Comput., vol. 85, p. 105788, 2019.

[75] S. Verma, N. Sood, and A. K. Sharma, "A novelistic approach for energy efficient routing using single and multiple data sinks in heterogeneous wireless sensor network," Peer--Peer Netw. Appl., vol. 12, no. 5, pp. 1110–1136, Sep. 2019, doi: 10.1007/s12083-019-00777-5.

[76] S. Verma, N. Sood, and A. K. Sharma, "QoS provisioning-based routing protocols using multiple data sink in IoT-based WSN," Mod. Phys. Lett. A, vol. 34, no. 29, p. 1950235, Sep. 2019, doi: 10.1142/S0217732319502353.

[77] S. Verma, S. Zeadally, S. Kaur, and A. K. Sharma, "Intelligent and secure clustering in wireless sensor network (WSN)-based intelligent transportation systems," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 8, pp. 13473–13481, 2021.

[78] S. Verma and K. Sharma, "Energy efficient zone divided and energy balanced clustering routing protocol (EEZECR) in wireless sensor network," Circuits Syst. Int. J. CSIJ, 2014.

[79] S. Verma, R. Mehta, D. Sharma, and K. Sharma, "Wireless sensor network and hierarchical routing protocols: A review," Int. J. Comput. Trends Technol. IJCTT, vol. 4, no. 8, Art. no. 8, 2013.

[80] S. Sakshi, S. Verma, P. Chaturvedi, and S. A. Yadav, "Least Significant Bit Steganography for Text and Image hiding," in 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), IEEE, 2022, pp. 415–421. Accessed: May 07, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9853052/

[81] R. Singh, S. Verma, S. A. Yadav, and S. V. Singh, "Copy-move Forgery Detection using SIFT and DWT detection Techniques," in 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), IEEE, 2022, pp. 338–343. Accessed: May 07, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9853192/

[82] A. Bhavani, S. Verma, S. V. Singh, and S. A. Yadav, "Smart Traffic Light System Time Prediction Using Binary Images," in 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), IEEE, 2022, pp. 367–372. Accessed: May 07, 2024. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9853071/

[83] V. G. Menon, S. Verma, S. Kaur, and P. S. Sehdev, "Internet of things-based optimized routing and big data gathering system for landslide detection," Big Data, 2021.