WORLD COMET RESEARCH GROUP

International Journal of Data Engineering and Intelligent Computing
Vol. 8, No. 1, 2025
Doi: 10.67228/30715717/IJDEIC-2025PI2F1C
PP. 01-13

Original Article

Energy-Efficient Data Processing Techniques in Distributed Computing

N. Seshagiri

Information Technology Pioneer, National Informatics Centre,India.

Received: 02-03-2025 Revised: 02-25-2025 Accepted: 03-03-2025 Published: 03-05-2025

ABSTRACT

The rapid growth of distributed computing paradigms such as cloud, edge computing, and large-scale data centers has significantly increased global energy consumption. As organizations increasingly rely on these systems for large-scale data processing, the need for energy-efficient methods has become critical due to rising operational costs and environmental concerns like carbon emissions. This paper analyzes energy-efficient data processing techniques in distributed environments, focusing on system-level optimization, algorithmic strategies, and resource management. It identifies major sources of energy consumption, including computation, data transfer, storage, and cooling, and highlights inefficiencies such as data redundancy, poor scheduling, network congestion, and underutilized resources. To address these challenges, the paper examines approaches such as energy-aware task scheduling, data locality optimization, dynamic voltage and frequency scaling (DVFS), virtualization, and workload consolidation. It also explores machine learning-based predictive models for adaptive resource allocation. A key contribution is the classification of these techniques across hardware, middleware, and application layers, along with a comparative analysis of their effectiveness. The proposed hybrid methodology integrates workload prediction, adaptive scheduling, and resource consolidation, demonstrating significant energy savings without compromising system performance. Overall, the study emphasizes the importance of coordinated, multi-layered strategies for achieving sustainable and energy-efficient distributed computing systems.

KEYWORDS

Energy Efficiency, Distributed Computing, Data Processing, Resource Management, Task Scheduling, Green Computing, Virtualization, Data Locality, DVFS, Cloud Computing.

1. INTRODUCTION

1.1. Background

The rapid development of cloud-base infrastructures and large scale distributed systems has redefined how data is processed, stored and accessed across industries. Nevertheless, this augmented dependence on computing resources has also given rise to a considerable increase in energy usage, especially in data centers which run 24/7 in order to satisfy the requirements of the users. These facilities drain a significant amount of world electricity, which adds to the increase in the cost of operation and the environmental issues like carbon emissions. One of the key reasons of this problem is the application of ineffective data processing and resource management methods which usually leads to underutilized hardware, excessive data flow, and unnecessary power consumption. With the ongoing increase in the scale of data, and computational demands, it is all the more important to begin adopting optimized and energy efficient approaches that can help reduce waste and improve the utilization of resources, and support sustainable computing practices.

1.2. Needs of Energy-Efficient Data Processing Techniques

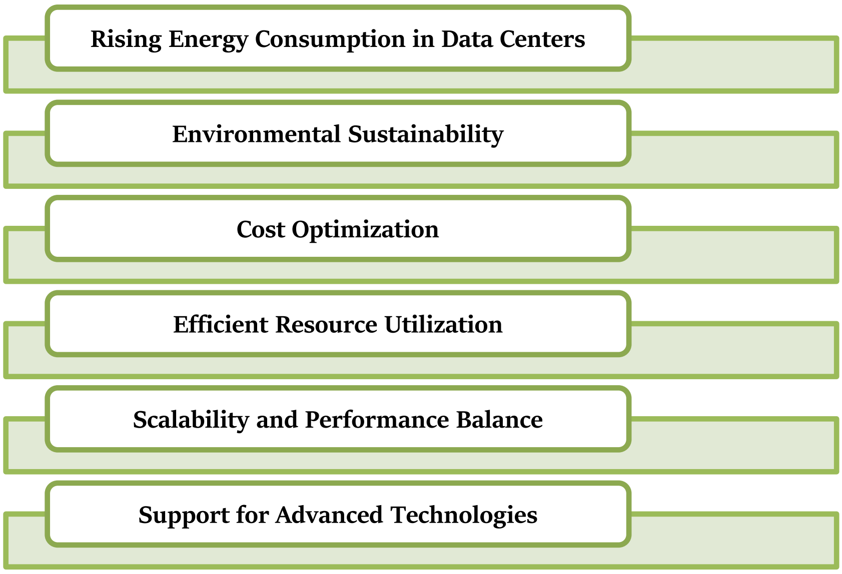

Fig 1 - Needs of Energy-Efficient Data Processing Techniques

1.2.1. Rising Energy Consumption in Data Centers

The explosive nature of growth of data generation and cloud services have greatly raised the energy requirements of data centers. These facilities are clock-based to serve applications like big data analytics, artificial intelligence, and online services. The constant expansion of infrastructure results in excessive use of electricity, increased operation expenses, and a greater environmental impact without the use of energy-efficient techniques. Optimization of energy consumption has therefore become a necessity in order to achieve a sustainable growth in the distributed computing environment.

ISSN: 3071-5717
PP: 01-13.

1.2.2. Environmental Sustainability

Computer systems that use a lot of energy also play a direct role in carbon emissions and environmental degradation. In the quest by organizations to ensure that their data processing systems achieve the global sustainability objectives, there has been an increase in the importance of reducing the carbon footprint of data processing systems. Green methods can be used to reduce the amount of power consumed, thus reducing greenhouse gas emissions and promoting greener computing practices. This is in line with the world efforts to fight climate change and green technology.

1.2.3. Cost Optimization

A significant part of the operational expenses of data centers and distributed systems is represented by energy costs. This may result in unnecessary energy wastage because of inefficient processing methods, which raise cost implications on organisations. Implementing data processing methods that consume less energy, companies can save a lot of money on electricity bills and overall cost efficiency can be improved. This does not only increase the profitability, but also the ability of more effective allocation of resources in the future with technological advancement.

1.2.4. Efficient Resource Utilization

Most traditional systems do not utilize their computing resources (CPU, memory, storage) to the fullest extent and are thus left idle or underutilized, wasting power unnecessarily. The energy-efficient methods are designed to optimize the use of resources by using intelligent scheduling, workload balancing, and consolidation strategies. This will help in ensuring that the available resources are utilized efficiently to minimise wastage and enhance performance of the system.

1.2.5. Scalability and Performance Balance

Modern distributed systems have to be able to support growing workloads and at the same time perform well. The energy efficient data processing methods can be used to provide a trade-off between scalability and performance by dynamically changing the allocation of resources based on demand. This allows systems to scale effectively without an equal increase in energy usage, so that they can be used in large-scale and dynamic applications.

1.2.6. Support for Advanced Technologies

New technologies include Artificial Intelligence, Internet of Things (IoT) and edge computing, which cannot operate effectively without efficient data processing. Close to real-time data processing and resource-constrained environment environments often make energy efficiency an essential need. Adoption of optimized processing methods will make sure that such sophisticated systems will be able to run reliably and at the same time with the minimal use of energy.

1.3. Challenges in Energy Efficiency

Energy efficiency in the distributed computing systems is not an easy task as there are several inherent challenges that arise due to the requirement to design the systems and the requirement to provide operational support of these systems. High communication overhead is one of the main

ISSN: 3071-5717
PP: 01-13.

problems whereby large amounts of data may be regularly transferred between the nodes of a distributed network. This continuous data flow uses a great deal of energy particularly in large scale systems where network operations can be more energy intensive than computation itself. The next significant issue is the problem of redundant data storage whereby there are multiple copies of the same data stored across different nodes so that the data remains unaffected by a failure at any one of the nodes. Although this helps in enhancing system resilience, it also leads to a higher storage need and greater energy usage that is not necessary. Furthermore, poor scheduling in accomplishing tasks is essential in energy wastage. Conventional methods of scheduling tend to focus more on performance measurements like speed of execution or load balancing without regard to energy implications. This may lead to the lack of even distribution of workload thus some of the nodes will be overworked and yet others will not be working at all but still consume power. Being closely connected with this is the question of underutilized computing power, where servers or processing units work below their capacity yet continue to consume a lot of energy. This inefficiency is especially prevalent in systems that do not provide dynamic management of their resources or adaptive scaling. Moreover, these challenges should be addressed in a holistic manner, which should take into account both optimization of hardware and software. Intelligent scheduling, data locality optimization and resource consolidation are techniques that can help alleviate these problems, but still, the effective implementation of these techniques remains a challenge because of the complexity of these systems and the variability of workloads. With the ongoing increase in the size and complexity of distributed systems, it is necessary to overcome these challenges to come up with energy efficient and sustainable computing solutions.

2. *LITERATURE SURVEY*

2.1. Energy-Aware Scheduling Techniques

Energy-conscious scheduling methods aim at scheduling tasks to computing resources in a manner that decreases the total power consumption whilst ensuring reasonable performance levels. These strategies take into account, the processor utilization, the workload factors and the energy profile of various nodes. Rather than assign tasks to nodes simply to achieve higher performance or to achieve load balancing, energy-aware schedulers can consider the energy efficiency of nodes or consolidate tasks on fewer machines to enable idle systems to enter low-power states. Real-time monitoring and adaptive decision-making are also aspects of advanced algorithms to ensure that the scheduling policies are dynamically adjusted to meet the changing workloads and system conditions. Consequently, such techniques play a large role in minimising operational expenses and environmental footprint in large scale computing infrastructure.

2.2. Data Locality Optimization

The goal of data locality optimization is to reduce the amount of data movement between the various nodes within a distributed system, which is a significant contributor to the energy consumption. These methods minimize network traffic, latency, energy overhead of transferring data by ensuring that computation is done where data is stored. The following strategies are often employed to enhance locality: task-data co-location, caching, intelligent data placement. When processing large volumes of data, in addition to improving performance, maximising data locality can also save a significant amount of energy because network operations can often use more power than the computer itself. This has made the concept of data locality a key factor when designing energy efficient distributed systems.

2.3. Virtualization and Resource Consolidation

The concept of virtualization technology is to have multiple virtual machines (VMs) running on a single physical server, which can utilize more hardware resources. Resource consolidation allows the workloads of underutilized servers to be migrated and consolidated on fewer servers, allowing idle servers to be brought offline or placed in energy-saving modes. This solution minimizes energy usage and hardware expenses and offers flexibility and scalability. Virtualization also facilitates dynamically allocating resources, in which computing power can also be adjusted depending on the demand. More efficiency is achieved through techniques like live migration and automated consolidation policy which continually re-optimize the use of resources in response to changes in workload.

2.4. Dynamic Voltage and Frequency Scaling (DVFS)

Dynamic Voltage and Frequency Scaling (DVFS) is a common method of minimizing the energy use of a processor by adjusting the operating voltage and frequency of a given processor, based on the demands in the workload. In cases where the utilisation of the system is low, DVFS will reduce the frequency and the voltage which will result in reducing power consumption, as the consumption of energy is proportional to the frequency and the voltage. At high demand times, the system will increase performance to ensure efficiency. DVFS is specifically a good fit in the setting with variable workloads because it enables systems to dynamically adjust performance and energy savings. DVFS is frequently a important part of energy-efficient computing approaches, with DVFS capabilities often being provided both at hardware and software levels.

2.5. Machine Learning Approaches

Machine learning techniques have proved to be an effective way of optimization of energy usage in computer systems. Machine learning models can forecast upcoming resource demands and make proactive decisions in regards to scheduling of work, resource allocation and power management. Reinforcement learning, regression models and neural networks are used to dynamically modify system parameters to achieve optimum energy efficiency. These models are also able to evolve with time, and thus would be more accurate and effective as more information is made available. By incorporating machine learning into energy management systems, smarter, automated optimization can be achieved, outperforming the traditional static or rule-based approaches.

3. METHODOLOGY

3.1. Proposed Framework

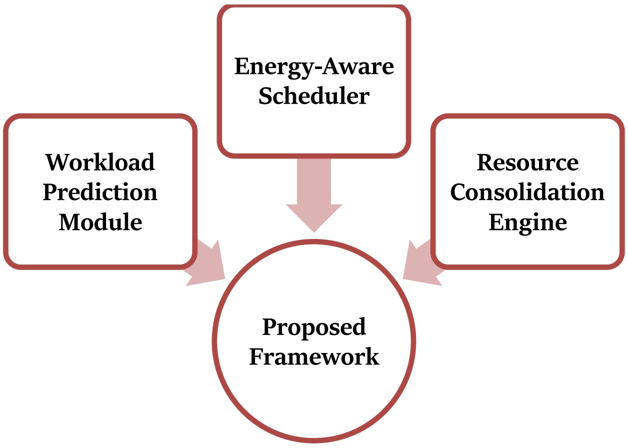

Fig 2 - Proposed Framework

3.1.1. Workload Prediction Module

The workload prediction module is developed to predict the future system demand on the basis of historical usage, as well as real time data. Through the use of statistical techniques or models of Machine Learning, this module can find trends, seasonal changes, and unexpected increases in workload. Good predictions allow the system to plan resources ahead of time, preventing excessive resource allocation and resource wastage. Consequently, the module is very important in enhancing the efficiency of energy as well as system performance through ensuring resources are allocated proactively and not reactively.

3.1.2. Energy-Aware Scheduler

The energy-conscious scheduler has the role of scheduling tasks to be allocated to the available computing resources in a manner that minimizes the total power consumption. This aspect considers energy profiles, system load, and resource efficiency on the decisions to be made compared to traditional schedulers, which places more emphasis on performance. It can either focus on the energy-efficient nodes or put off the non-critical tasks during the peak power consumption periods. The scheduler can guarantee optimal use of resources and minimize unnecessary energy spending throughout the system by introducing the policies that balance the performance and energy saving.

3.1.3. Resource Consolidation Engine

The resource consolidation engine is involved in the optimization of workloads done by the physical machines by consolidating workloads onto fewer servers. Based on Virtualization concepts, it allows dynamic migration of tasks or virtual machines among underutilized nodes to more active nodes. This enables machines that are idle or lightly loaded to be shut down or put into low-energy states, which greatly reduces the consumption of energy. The engine keeps watching usage of the system and adjusts strategy of consolidation in real time to provide both efficiency and reliability in dynamic computing environments.

ISSN: 3071-5717
PP: 01-13.

3.2. System Architecture

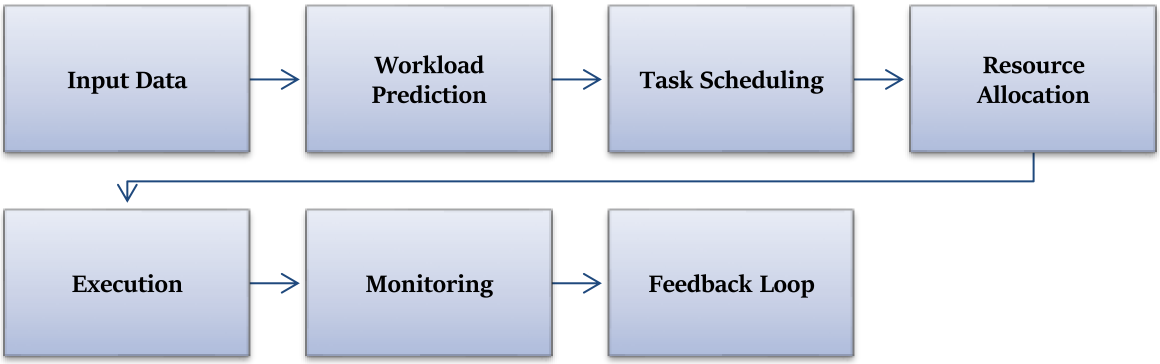

Fig 3 - System Architecture

3.2.1. Input Data

The system starts with input data, which comprises historical workload data, current system measurement data, user requests, and resource consumption data. This information is the basis of all other processing and decisions. It can include parameters like CPU consumption, memory consumption, arrival rates of tasks, and network activity. Well-structured and of high quality input data can be ensured, so that the system can make accurate predictions and effective scheduling decisions.

3.2.2. Workload Prediction

At this phase, the system processes input data to determine the future workload requirements. With the help of the approaches of the Machine Learning or statistical analysis, the module determines the patterns and predicts the further resource demand. Proper prediction of workload can be used to prepare the system to have demand fluctuations efficiently reducing delays and unnecessary energy consumption due to over-provisioning.

3.2.3. Task Scheduling

Task scheduling is the process of assigning incoming tasks to suitable computing resources, based on the anticipated workload and system conditions. The scheduler takes into account the following factors: task priority, availability of resources, and efficiency of energy. Advanced scheduling techniques seek to trade-off performance of the system and low energy consumption by choosing the best execution nodes and timing of each task.

3.2.4. Resource Allocation

Resource allocation is a process used to determine how the system resources (CPU, memory and storage) are distributed among the tasks scheduled to run on the system. This step will make sure that each job gets enough resources to run effectively without contention of resources or wastage. The allocation decisions are normally dynamic that reacts to variations in the workload and system performance to ensure maximum utilization.

3.2.5. Execution

In the execution stage, tasks are executed on the allocated resources. It allows the system to run smooth because dependencies are dealt with, parallel execution is handled and performance requirements are maintained. Effective performance is important in satisfying the user demands and meeting the system throughput requirements without violating the energy-conservation policies.

3.2.6. Monitoring

The monitoring block continuously monitors the performance of the system, the resources used and the amount of energy consumed during the execution. It gathers real-time statistics of important metrics like CPU load, response time, and power usage. This knowledge is needed to identify anomalies, determine system efficiency, and facilitate the adaptive decision-making process at the later stages.

3.2.7. Feedback Loop

Monitoring data is used to close the system cycle by refining predictions of future decisions, scheduling and assigning resources. It allows the system to be informed by the previous behavior and to be improved with time. The feedback loop can increase reliability, efficiency, and responsiveness of the system by incorporating adaptive mechanisms, resulting in a system that is continuously optimized in dynamic environments.

3.3. Algorithm Design

The given algorithm is supposed to maximize the performance as well as the energy efficiency with the help of a series of smart and adaptive solutions. Firstly, the system forecasts future workload requirements based on historical data and system inputs in real-time. This prediction step draws on a method of Machine Learning to detect patterns like the most common time of the day, seasonal changes, and spikes in demand that are not regular. Following these predictions, the algorithm then goes ahead and classifies incoming tasks based on their resource needs, including CPU intensity, memory usage, and I/O needs. This type of classification guarantees that the tasks are addressed accordingly and correlated with the relevant resources. After classification, the algorithm assigns tasks to the system nodes that are energy-efficient. It does not allocate work loads equally, but allocates work loads to nodes that have superior energy-performance properties, thus minimizing the total power consumption without compromising system throughput. After tasks have been assigned the algorithm then actively tracks the use of resources with the aim of identifying machines that are underutilized. It allocates the workloads by migrating workloads or virtual machines across nodes with minimal workloads to more active nodes through the use of Virtualization techniques. This will enable idle systems to be turned off or put into low-energy states, further saving on energy. The last is the ongoing checking and dynamic adjusting. The system monitors key performance indicators like resource utilization, time of task completion and energy consumption in real time. With this feedback, the algorithm is able to adjust its scheduling and allocation plans in response to the changing environment, so that it is maximally efficient at any given time. This iterative and dynamic process allows the system to balance between performance

ISSN: 3071-5717
PP: 01-13.

and energy consumption, making it extremely appropriate in the current distribution and cloud computing systems.

4. RESULTS AND DISCUSSION

4.1. Experimental Setup

The experimental design to test the proposed system was developed to be a simulation of realistic distributed computing environment under different and varying workload conditions. Artificially generated distributed workloads of varying data size and varying complexity of the computations were used to test the system in order to provide a comprehensive evaluation of how the system performs under different circumstances. Five categories of workloads, including small, medium, and large datasets, were used to represent lightweight workloads, moderate processing workloads, and data-intensive workloads, respectively. Also, complexity of computation was between simple operations with minimal processing needs to highly intensive operations with significant CPU, memory, and I/O usage. This difference was to make sure that the adaptability and efficiency of the system could be well studied. The simulation modeled real-world conditions by having a heterogeneous set of nodes with different resource configurations, in both processing power, memory capacity and energy consumption profile. The framework applied ideas of Distributed Computing to simulate how tasks are distributed, time delays in communication, and sharing of resources among nodes. Random arrival patterns of tasks and varying levels of demand were also included in the workload generation process to simulate the unpredictable user behavior. The execution time, resource consumption, energy usage, and system throughput are key performance metrics that were monitored during the experiments. Moreover, the comparisons were carried out on the basis of traditional scheduling methods which neglect energy efficiency and, therefore, allows to clearly evaluate the benefits of the proposed system. Repeated simulation runs were also included in the experimental setup to guarantee consistency and reliability of the results. The arrangement offered a sound platform of validating the efficacy of the created energy-conscious architecture in enhancing both the performance and energy consumption.

4.2. Performance Evaluation

Table 1: Performance Evaluation

Technique	Energy Reduction (%)
Traditional Scheduling	0%
DVFS	18%
Data Locality Optimization	25%
Virtualization	30%
Proposed Hybrid Approach	45%

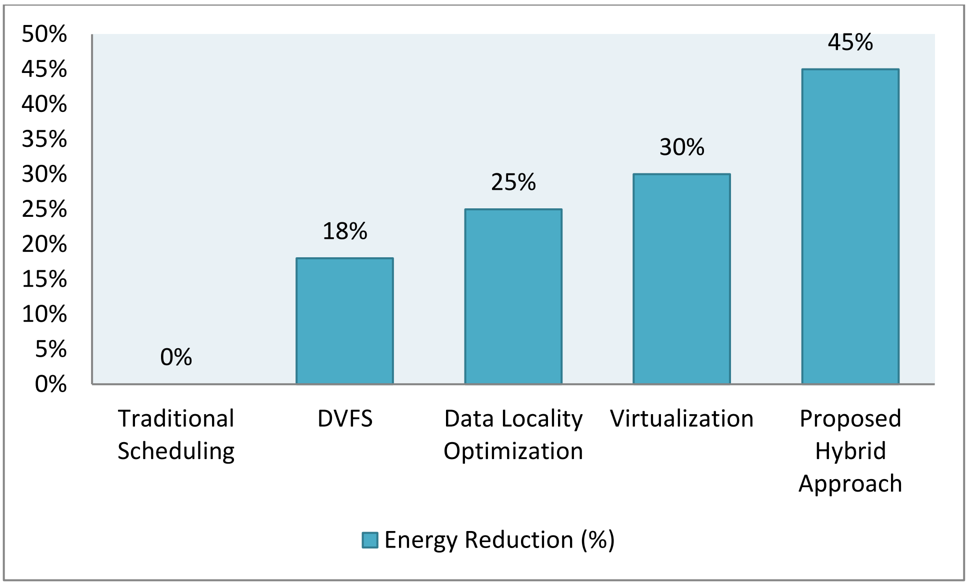

Fig 4 - Performance Evaluation

4.2.1. Traditional Scheduling (0%)

The old scheduling methods are more concerned with performance indicators like completion time of tasks together with system throughput without taking into consideration the energy efficiency as a goal. To balance the load or to minimize the time of execution, tasks are allocated among available resources, which often results in unnecessary energy consumption because of idle or less-utilized nodes kept alive. Consequently, this method demonstrates no apparent energy savings, which can be used as a benchmark in the comparison of more sophisticated and energy-sensitive methods.

4.2.2. Dynamic Voltage and Frequency Scaling (DVFS) – 18%

Dynamic Voltage and Frequency Scaling (DVFS) is a moderate form of energy savings obtained by varying the operating voltage and frequency of the processor, depending on the workload requirements. When the system is not heavily used, it reduces the CPU speed and power usage, thus saving power. Although DVFS can be effective in reducing power consumption at the processor level, it is not as effective as it will not affect the inefficiencies associated with task scheduling, data movement, or resource utilization at a system level. This will lead to a total energy saving of approximately 18%.

4.2.3. Data Locality Optimization – 25%

Data locality optimization improves energy efficiency by reducing the need for data transfer across nodes in a distributed environment. The system will reduce network communication, which in most cases is energy-consuming. This method does not only reduce power usage but also enhances the performance through decreasing the latency. Data locality optimization, with a more efficient utilization of network resources will achieve a higher energy reduction of about 25% than traditional approaches.

ISSN: 3071-5717
PP: 01-13.

4.2.4. Virtualization – 30%

Resource consolidation involves virtualization to perform multiple workloads with fewer physical machines, resulting in increased energy efficiency. The system can spin down or put idle unneeded machines by moving the workload off the underutilized servers to the more active servers. This strategy enhances the use of hardware and lessens the amount of overheads in the operations, which result in an energy savings of approximately 30%. Nevertheless, virtualization is not a technology that can maximize the use of energy without incorporating other methods.

4.2.5. Proposed Hybrid Approach – 45%

The hybrid approach proposed is a combination of various energy-saving strategies such as energy-conscious scheduling, optimization of data locality, DVFS and virtualization. The system combats energy inefficiencies at many levels, starting with the workings of the processors, through network communications and resource allocation. This full optimization allows the system to dynamically respond to changes in workloads, optimize resource usage, and reduce energy wastage. Due to this, the hybrid method has the highest energy reduction of about 45, which shows that it is the most effective in comparison with the individual methods.

4.3. Analysis

The performance analysis has made it clear that the proposed hybrid approach is more energy efficient than the individual techniques, since the proposed hybrid approach is able to effectively integrate the complementary advantages of the individual techniques. Contrary to the standalone approaches, which aim at a single area of energy usage in the system, the hybrid model focuses simultaneously on several sources of energy consumption in the system. The saving of significant amounts of energy is accomplished by optimizing idle time with intelligent workload consolidation, and ensuring that the number of machines that remain active without affecting performance is reduced. Moreover, the utilisation of the data locality strategies also minimise unnecessary data transfer across nodes which, in addition to minimising network latency, reduce the energy overhead associated with communication. Efficient allocation of workload is also known to improve the performance of the system by assigning tasks to the most appropriate and energy efficient resources and avoid both overloading and underutilization of nodes. In addition, the hybrid strategy dynamically adjusts to the changing workload conditions and can therefore be able to sustain optimal efficiency in real time. The system will then constantly optimize its decisions as more information is provided by the use ofMachine Learning, to ensure overall effectiveness is improved. This flexibility has made energy savings to be maintained even during the changing demand trends. The other main benefit is the trade-off between performance and energy consumption is balanced, so efficiency improvements do not occur at the expense of a longer execution time or lower throughput. On the whole, the analysis shows that the combination of various optimization strategies into the single framework leads to the creation of the more robust, scalable, and energy-efficient system, thus the hybrid approach is highly appropriate to the modern distributed and cloud computing environment.

ISSN: 3071-5717
PP: 01-13.

5. CONCLUSION

Data processing that is energy-efficient has emerged as a crucial challenge towards achieving sustainability and cost-efficiency in the current distributed computing systems. As the use of data-intensive applications and cloud infrastructures has rapidly grown, the growth in the number of computational resources has correspondingly increased, posing a higher energy consumption and environmental impact. The paper has presented a thorough discussion of the current energy optimization methods, such as energy-conscious scheduling, data locality optimization, virtualization, and Dynamic Voltage and Frequency Scaling (DVFS). Although each of these methods helps to decrease the amount of energy used, their independent effect tends to be rather minimal in the case of their application on an individual basis. To overcome this shortcoming a hybrid framework was suggested that combines various optimization approaches into a single framework.

The proposed framework will integrate smart workload prediction, effective scheduling of tasks, and dynamism in consolidating resources to achieve higher energy efficiency. Using the principles of the Machine Learning, the system can predict the workload needs and make proactive decisions that will help to reduce the amount of wasted energy. Also, the combination of data locality-based methods minimizes the amount of data moving unnecessarily, and virtualization allows the utilization of resources in a more effective way by combining workloads into a smaller number of physical machines. The outcomes of the experiments clearly indicate that this hybrid approach is significantly better performance in contrast to traditional methods, with an impressive capacity of saving a significant amount of energy without any impact on the performance of the systems. This renders the framework very appropriate to large-scale distributed systems like cloud data centers and high-performance computing systems.

In the future, there are some of the possible directions that future work can take to further improve the effectiveness of the proposed system. One of the key areas is the implementation of the framework into the real-time cloud environments where the dynamic and unpredictable workloads need to be constantly adapted and swift decisions made. A second key direction is related to more advanced AI-driven optimization methods, including deep learning and reinforcement learning, to enhance the accuracy of predictions and provide the ability to fully manage the work of the system. Also, the study of how this framework can be utilized in edge computing is a promising endeavor, because edge environments typically have very limited resource and energy availability. These emerging areas can be extended with the proposed approach to create more scalable, intelligent, and sustainable computing systems that can meet the changing needs of the modern technology.

REFERENCES

[1] Rajkumar Buyya, A. Beloglazov, and J. Abawajy, “Energy-aware resource allocation heuristics for efficient management of data centers,” Future Generation Computer Systems, 2012.

[2] Anton Beloglazov and Rajkumar Buyya, “Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines,” Concurrency and Computation, 2012.

[3] Jeffrey Dean and Sanjay Ghemawat, “MapReduce: Simplified data processing on large clusters,” Communications of the ACM, 2008.

[4] Matei Zaharia et al., “Spark: Cluster computing with working sets,” USENIX HotCloud, 2010.

ISSN: 3071-5717
PP: 01-13.

[5] Luiz André Barroso and U. Hölzle, The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines, 2013.
[6] Andrew S. Tanenbaum and H. Bos, Modern Operating Systems, 4th Edition, 2015.
[7] Thomas D. Burd and Robert W. Brodersen, “Energy efficient CMOS microprocessor design,” IEEE HICSS, 1995.
[8] Trevor Mudge, “Power: A first-class architectural design constraint,” Computer, 2001.
[9] Weisong Shi and K. Li, “The energy efficiency of DVFS algorithms,” Journal of Parallel and Distributed Computing, 2007.
[10] Carl Waldspurger, “Memory resource management in VMware ESX server,” OSDI, 2002.
[11] Christopher Clark et al., “Live migration of virtual machines,” NSDI, 2005.
[12] Jeffrey O. Kephart and D. M. Chess, “The vision of autonomic computing,” IEEE Computer, 2003.
[13] Rich Caruana, “Multitask learning,” Machine Learning Journal, 1997.
[14] Corinna Cortes and Vladimir Vapnik, “Support-vector networks,” Machine Learning, 1995.
[15] Ian Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
[16] Gajula, S. (2024). Cybersecurity risk prediction using graph neural networks. *Journal of Information Systems Engineering and Management*.
[17] Gajula, S. (2024). Adaptive zero trust architecture for securing financial microservices. Computer Fraud & Security, 2024(12), 643–655. https://doi.org/10.52710/cfs.845
[18] Gajula, S. (2023). A review of anomaly identification in finance frauds using machine learning system. *International Journal of Current Engineering and Technology, 13*(6), 568–575. https://ijcet.evegenis.org/index.php/ijcet/article/view/820