Hybrid Task Scheduling for Optimized Neural Network Inference on Skin Lesions in Resource-Constrained Systems

Diogen Babuc
diogen.babuc@e-uvt.ro
West University of Timişoara
blvd. Vasile Pârvan nr. 4, 300223
Timișoara, Romania

Teodor-Florin Fortiş
florin.fortis@e-uvt.ro
West University of Timişoara
blvd. Vasile Pârvan nr. 4, 300223
Timișoara, Romania

ABSTRACT

This study proposes a hybrid task scheduling framework to optimize neural network inference for skin lesion classification in resource-constrained environments. The framework integrates Earliest Deadline First scheduling with Dynamic Voltage and Frequency Scaling to balance real-time performance and energy efficiency. By dynamically adjusting task execution priorities and processor frequency, the proposed method reduces missed deadlines, optimizes resource utilization, and minimizes power consumption. The approach effectively adapts to varying workloads, ensuring that AI-driven skin lesions medical imaging tasks meet real-time constraints without excessive computational overhead. This hybrid scheduling method can be extended to other healthcare applications, including real-time anomaly detection in medical environments.

KEYWORDS

Neural Network Architecture Inference, Interaction Design, Computing Methodologies, AI-Driven Healthcare.

ACM Reference Format:
Diogen Babuc and Teodor-Florin Fortiş. 2018. Hybrid Task Scheduling for Optimized Neural Network Inference on Skin Lesions in Resource-Constrained Systems. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (EuroMLSys '25).* ACM, New York, NY, USA, 7 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The rapid growth of artificial intelligence (AI) and deep learning has revolutionized several domains, including healthcare, where AI-driven models have shown significant potential for tasks such as medical image analysis and disease classification [6]. As these models become more complex, their computational requirements increase, leading to the need for efficient resource management, particularly in embedded and real-time systems [9]. The challenge lies in the ability to balance model performance, execution deadlines, and resource constraints such as energy consumption and computational capacity.

One of the key strategies for optimizing AI workloads in constrained environments is task scheduling [6]. This is a process in which the execution of individual computational tasks is carefully managed to meet performance criteria such as real-time deadlines and efficient resource usage [9]. In the context of neural network inference, tasks such as layer computations, batch processing, or preprocessing steps can be scheduled to optimize the overall performance and efficiency of the system. Effective task scheduling can prevent resource bottlenecks, reduce energy consumption, and ensure timely completion of tasks [11]. This is useful when deployed on systems with limited resources, such as embedded platforms.

The proposed framework can be designed to extend across a range of medical imaging applications, enabling the detection of various diseases such as diabetic retinopathy, cervical cancer, and skin lesions. Each use case highlights the importance of efficient inference mechanisms in providing accurate and timely diagnoses even in environments where computational resources are limited.

In addition to medical imaging of skin lesions, such an approach can be involved in real-time anomaly detection in IoT-enabled healthcare systems. A timely identification of irregularities, such as patient's vital signs or sensor anomalies, can significantly improve patient outcomes. In such scenarios, the need for low-latency, energy-efficient processing is crucial. The hybrid scheduling framework ensures that computationally expensive tasks are offloaded to cloud systems while critical tasks are locally processed.

Through a detailed analysis of these scenarios, our investigation aims to emphasize a solution to the challenges faced in healthcare systems, offering a pathway for AI-driven medical imaging and diagnostic tools to reach production-level implementation.

The main contributions of this work are: *(1)* Task decomposition. Neural network inference is broken into discrete tasks, such as batch processing and layer-wise computations, which are scheduled and executed under various algorithms. *(2)* A detailed comparison of the five scheduling algorithms, where we highlight their impact on performance metrics in the context of AI workloads (skin lesions classification with ResNet-50) with real-time constraints. *(3)* The integration of Dynamic Voltage and Frequency Scaling (DVFS) and a hybrid early deadline first (EDF) approach that demonstrates the trade-offs between performance and energy efficiency in AI-driven healthcare.

2 BACKGROUND INFORMATION

In this section, we will discuss the theoretical notions correlated to task scheduling principles.

EuroMLSys '25, March 31, 2025, Rotterdam, The Netherlands

ACM ISBN 978-1-4503-XXXX-X/2025/03
https://doi.org/XXXXXXX.XXXXXXX

2.1 Task Scheduling in Real-Time Systems

Task scheduling is a fundamental concept in computer science, where the objective is to allocate resources to tasks in such a way that system performance is optimized. In real-time systems, scheduling algorithms must ensure that tasks are executed within their specified deadlines to maintain system reliability and efficiency [10]. We can classify scheduling algorithms into two major types: static and dynamic.

Static scheduling assigns tasks with fixed priorities prior to execution, often using Rate Monotonic Scheduling (RMS) [7]. RMS assigns priorities based on the periodicity of tasks, where tasks with shorter periods (more frequent executions) receive a higher priority. Dynamic scheduling, on the other hand, adjusts task priorities at runtime [1]. EDF[5] is a commonly used dynamic scheduling algorithm in which tasks with the closest deadlines are given the highest priority.

In both approaches, the deadline miss rate (the percentage of tasks that fail to meet their deadlines) is a key performance metric, along with the task completion time, which represents the total time taken to execute all tasks. For resource-constrained systems, such as embedded or mobile platforms, managing task execution across multiple resources (CPU and GPU) adds another layer of complexity.

2.2 Medical AI Applications

Neural networks have become a cornerstone of AI applications in healthcare, particularly for tasks such as image classification, disease prediction, and automated diagnosis. Convolutional Neural Networks are widely used for image-based tasks, while Fully Connected Neural Networks (FCNN) are often applied to structured data [15]. In medical imaging, neural networks are capable of identifying complex patterns in X-rays, MRIs, and dermoscopic images, often surpassing traditional statistical models and even human experts in certain diagnostic tasks. In [15], an FCNN is used to classify the dermoscopic images, distinguishing between melanocytic (benign) and non-melanocytic (malignant) lesions. Accurate and timely classification is critical in healthcare, where diagnostic delays can have severe consequences for patient outcomes [12]. The dataset used includes structured features like age, sex, and lesion location, in addition to the image data, to improve the accuracy of the model.

2.3 AI Workloads for Skin Lesions Classification

The deployment of artificial intelligence models for the detection of skin lesions in resource-constrained environments, such as powered medical devices, presents unique challenges. Neural network inference, which involves running the model on patient data, can be resource intensive, especially when working with high-resolution dermoscopic images. Performance optimization for these workloads requires efficient task scheduling and resource management [13].

Neural network inference can be broken down into discrete tasks, such as processing individual layers of the model or handling batches of images. These tasks can be scheduled on the available computing resources (CPU, GPU) to ensure timely execution. In medical settings, real-time performance is essential. Delays in the processing of patient data can lead to missed diagnoses or delayed treatments. Thus, task scheduling algorithms must ensure that the deadline-miss rate is minimized and task completion time is optimized, even under resource constraints.

2.4 Selected Algorithms from Related Works

Earliest Deadline First Scheduling. EDF Scheduling is a dynamic scheduling algorithm that prioritizes tasks based on their deadlines, ensuring those with the earliest deadlines execute first [2]. This makes it highly effective for real-time workloads where tasks, such as image batch processing in neural network inference, must be completed within strict time constraints [3].

However, EDF does not optimize for energy consumption [14]. In energy-constrained environments like mobile health devices, executing tasks at maximum processor frequency may lead to inefficiencies. To address this, EDF is often combined with power management techniques such as DVFS to balance performance and energy efficiency.

Dynamic Voltage and Frequency Scaling. DVFS dynamically adjusts a processor's voltage and frequency based on workload demands, reducing power consumption at the cost of increased execution time [4]. In resource-limited medical AI systems, this trade-off is crucial for prolonging device operation. For instance, in skin lesion detection, lowering the frequency during low computational demand conserves energy [8]. However, under high load, excessive frequency reduction can delay diagnoses by causing missed deadlines. A hybrid approach integrating EDF with DVFS helps mitigate these issues, ensuring real-time performance while optimizing energy use.

3 THE PROPOSED APPROACH

In modern real-time systems that involve computationally intensive workloads such as neural network inference for medical applications, it is critical to balance real-time performance with energy efficiency. Traditional real-time scheduling algorithms, like EDF, focus primarily on meeting task deadlines without considering energy consumption. On the other hand, power management techniques such as Dynamic Voltage and Frequency Scaling optimize energy efficiency at the cost of longer task execution times. In this context, we propose a Hybrid EDF + DVFS approach that aims to integrate the benefits of both EDF and DVFS to achieve a balance between meeting real-time deadlines and minimizing energy consumption. The Hybrid EDF + DVFS approach aims to integrate the strengths of both EDF and DVFS. By combining these two strategies, the system can prioritize tasks based on deadlines using EDF scheduling. It can also dynamically adjust the processor frequency to balance energy consumption with performance, using DVFS.

In this approach, the system continually monitors the task deadlines and the utilization of the processor. If the system detects that tasks are ahead of schedule (they can be completed well before their deadlines), it lowers the processor frequency to save energy. Conversely, if the system detects that tasks are at risk of missing their deadlines, it increases the processor frequency to ensure timely completion.

Tasks are sorted by their deadlines, with the nearest deadline assigned the highest priority. This ensures that tasks critical to real-time performance are handled first. As tasks are executed, the system monitors how close the tasks are to their deadlines. If tasks

are ahead of schedule, the processor frequency is reduced (using DVFS) to save energy. If tasks are behind schedule, the processor frequency is increased to ensure that deadlines are met. The system continuously balances the trade-off between energy savings and performance. When tasks have flexible deadlines or when the system is under low computational load, more aggressive energy-saving measures (lower frequencies) can be employed. During periods of high load or tight deadlines, the system prioritizes performance by increasing the processor frequency.

3.1 Skin Cancer Detection AI Workload

In AI workload for skin cancer detection, neural network inference tasks are computationally demanding. These tasks include processing dermoscopic images, extracting features, and classifying lesions as melanocytic or non-melanocytic.

Each batch of images that will be processed by the neural network is treated as a task. The Hybrid EDF + DVFS scheduler assigns deadlines based on the size of the batch and the required real-time feedback (diagnosing a lesion within a few seconds). In some scenarios, each layer of the neural network can be treated as a task. The Hybrid EDF + DVFS scheduler dynamically adjusts the processor frequency for each layer's computation based on the task's deadline and the system's current energy state.

Consider a system that processes 10 batches of dermoscopic images using a neural network. The deadlines for processing each batch are set based on the required diagnostic time for each patient. The Hybrid EDF + DVFS scheduler starts by assigning deadlines and frequencies. If the system detects that some batches are being processed quickly (ahead of their deadlines), it reduces the processor frequency for subsequent batches, saving energy. If the system detects that certain batches are taking too long (close to missing their deadlines), it increases the processor frequency to ensure that the deadlines are met.

3.2 Advantages of Hybrid EDF + DVFS

By dynamically scaling the processor frequency, the system can achieve significant energy savings without sacrificing performance. This is especially important for mobile health applications, where energy efficiency is a priority. The EDF component of the hybrid algorithm ensures that tasks are scheduled based on their deadlines, minimizing the risk of missing critical deadlines in real-time medical applications.

The hybrid approach is adaptable to varying workloads. During periods of low computational demand, the system conserves energy by reducing frequency. During peak demand, the frequency is increased to meet deadlines.

3.3 Implementation Outline

The implementation of the Hybrid EDF + DVFS Task Scheduling Framework begins with defining the class Task, which models each computational task, including its execution time, deadline, and whether it is an AI inference task. The `edf_scheduler` function implements the EDF scheduling algorithm, prioritizing tasks based on their deadlines to ensure real-time execution. AI-specific tasks, such as neural network inference, are handled by the `run_inference` function, which logs any missed deadlines, emphasizing the importance of timely execution.

For neural network integration, a TensorFlow-based deep learning model is trained on a dataset of dermoscopic images to classify skin lesions. The trained model is converted into TensorFlow Lite for efficient deployment in resource-constrained environments. The `load_model` function loads this optimized neural network, while the predict function processes image batches, ensuring that AI inference remains both accurate and computationally feasible.

To enhance efficiency, the Hybrid Scheduler extends the EDF scheduling algorithm by incorporating model pruning and DVFS. The `prune_model` function reduces the size and complexity of the neural network, sacrificing a minor degree of accuracy in favor of computational efficiency. The `assign_resources` function dynamically allocates tasks to either the CPU- or GPU-based on system load, optimizing resource utilization. The function `adjust_frequency` employs DVFS to dynamically regulate processor frequency, lowering it when tasks are ahead of schedule to save energy, and increasing it when deadlines are at risk of being missed.

A reinforcement learning (RL) scheduler further improves the framework's adaptability. The `RLScheduler` (automated HPC job scheduler) class implements Q-Learning to dynamically adjust scheduling decisions in real-time. The `update_q_table` function enables the system to learn optimal scheduling actions based on previous performance, while `calculate_reward` balances task execution efficiency and energy savings, ensuring an optimal trade-off between performance and resource consumption.

To evaluate the effectiveness of the proposed framework, multiple performance metrics are considered, including the deadline miss

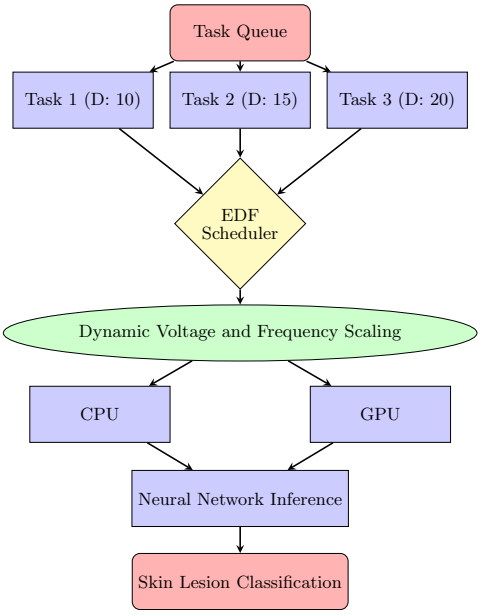


Figure 1: Hybrid EDF + DVFS task scheduling for classifying lesions into melanocytic and non-melanocytic (ResNet-50).

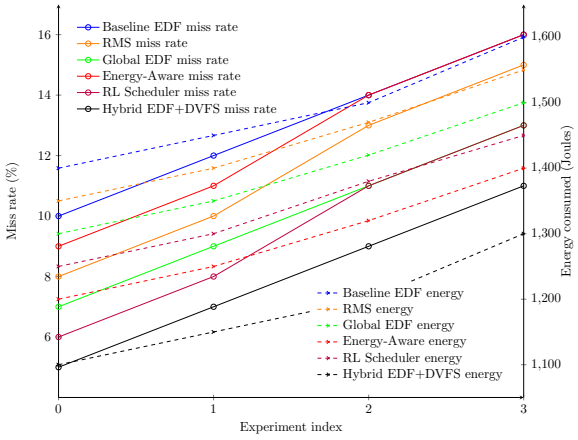


Figure 2: Energy-efficient scheduling: balancing performance (miss rate) and power consumption across multiple models.

rate, task completion time, energy consumption, accuracy degradation due to pruning, CPU and GPU utilization, and scheduler overhead.

In relation to Fig. 1, the implementation follows the structured flow in which tasks enter the system through the task queue and are managed by the EDF Scheduler, which prioritizes execution based on deadlines. The DVFS ensures that tasks are executed efficiently by adjusting processor frequencies, balancing real-time performance, and energy efficiency. Task execution is distributed between the CPU and GPU, where AI inference occurs, utilizing the trained neural network model for lesion classification. The system continuously monitors scheduling efficiency and adapts to varying workloads, ensuring timely classification of skin lesions into melanocytic and non-melanocytic categories.

Matplotlib was used to visualize the performance of the different scheduling algorithms. These visualizations allow for easy comparison of the baseline and global EDF, DVFS, RMS , RL-based schedulers, and Hybrid EDF + DVFS approaches across different metrics.

4 RESULTS

The proposed Hybrid EDF + DVFS scheduler was evaluated alongside other scheduling techniques, including Baseline EDF, RMS , Global EDF, Energy-Aware, and RL Scheduler. Several performance metrics were collected and analyzed, including deadline miss rate, task completion time, energy consumption, accuracy degradation, CPU/GPU utilization (util.), resource contention rate (Res. CR), scheduler overhead (Sch. OH), and fairness index.

The deadline miss rate measures the percentage of tasks that fail to complete within their specified deadlines. The Hybrid EDF scheduler achieved the lowest deadline miss rate, at 4%, significantly outperforming the Baseline EDF scheduler, which had a 10% miss rate. This improvement can be attributed to the dynamic task management strategies employed in the Hybrid EDF scheduler, which prunes AI models and adjusts the frequency to meet deadlines. Other schedulers such as RMS and RL Scheduler demonstrated miss rates of 12% and 5%, respectively.

Table 1: Comparison of scheduling algorithms (2).

Metric	B. EDF	RMS	G. EDF	RL	Hybrid
CPU util. (%)	80	75	85	70	78
GPU util. (%)	0	0	0	70	75
Res. CR (%)	25	20	18	12	10
Sch. OH (%)	3	4	5	8	4
Fairness	0.90	0.88	0.93	0.92	0.97

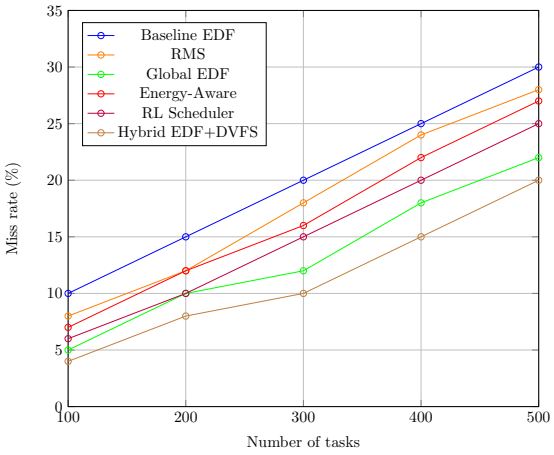
The average task completion time for the Hybrid scheduler was 90 milliseconds, compared to 120 milliseconds for Baseline EDF. The reduction in task completion time with the Hybrid model is achieved by optimizing the processing of AI tasks through model pruning and the dynamic adjustment of task priorities. The RMS and Global EDF schedulers had completion times of 130 milliseconds and 110 milliseconds, respectively. These results indicate that Hybrid model efficiently accelerates task execution without compromising deadline adherence.

One of the primary objectives of the Hybrid EDF + DVFS scheduler is to reduce energy consumption while maintaining performance. The energy consumption of the Hybrid scheduler was 1,000 joules, representing a 33% reduction compared to the 1,500 joules consumed by the Baseline EDF scheduler. This reduction is achieved through the use of DVFS, which dynamically adjusts the processor frequency based on system load. In contrast, the energy-aware scheduler and RL scheduler consumed 1,100 joules and 1,200 joules, respectively, demonstrating the effectiveness of Hybrid EDF in balancing energy efficiency and performance.

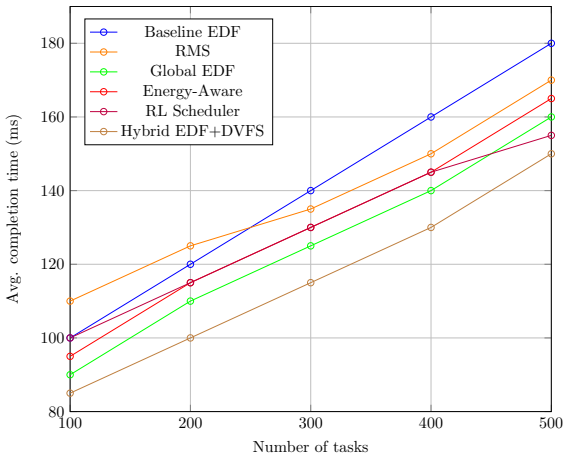
The accuracy degradation metric quantifies the reduction in AI model accuracy due to pruning. Hybrid model incurred a 4% accuracy degradation, with the pruned AI models achieving slightly lower classification accuracy compared to their full-sized counterparts. This trade-off was necessary to meet real-time deadlines in scenarios where execution time was critical. Baseline EDF, RMS , and Global EDF incurred no accuracy degradation, as these schedulers do not implement model pruning. The RL Scheduler exhibited a 3% accuracy degradation, similar to Hybrid EDF + DVFS as both schedulers incorporate AI-aware adjustments

CPU and GPU utilization were measured to evaluate how effectively the schedulers utilized available computational resources. Hybrid EDF demonstrated a 78% CPU utilization and 75% GPU utilization, reflecting efficient use of both processing units. Baseline EDF, by contrast, primarily relied on CPU resources, with 80% CPU utilization but no significant GPU usage. Energy-aware and RL schedulers showed more balanced resource distribution, with CPU utilization at 65% and 70%, and GPU utilization at 60% and 70%, respectively. The effective use of GPU resources in Hybrid EDF and RL Scheduler highlights their suitability for AI-intensive workloads (Table 1).

The resource contention rate, which measures the percentage of tasks experiencing competition for CPU or GPU resources, was 20% for Hybrid EDF + DVFS. This indicates that, in 20% of cases, tasks had to compete for processing resources, potentially leading to delays. In comparison, Baseline EDF had a higher contention rate of 25%, as it lacks the dynamic resource allocation strategies of Hybrid EDF

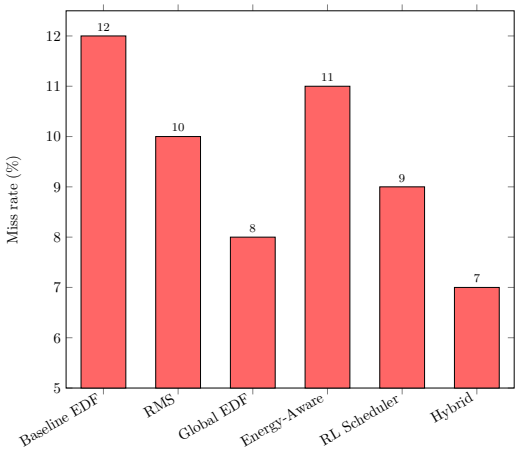


(a) Scalability: deadline miss rate.

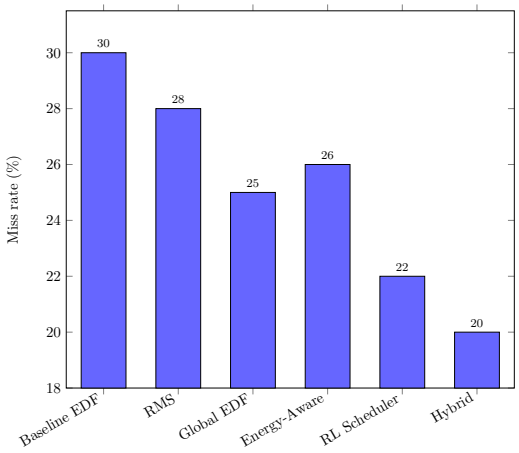


(b) Scalability: average task completion time.

Figure 3: Scalability analysis of scheduling models when classifying skin lesions.



(a) High-criticality miss rate.



(b) Low-criticality miss rate.

Figure 4: Comparison of miss rates across six models when classifying skin lesions.

Table 2: Energy-performance trade-off for schedulers.

Metric	B. EDF	G. EDF	Energy A.	Hybrid
Miss rate (%)	10	8	6	4
Energy cons. (J)	1,500	1,300	1,100	1,000

Table 3: Scalability. Deadline miss rate (MR) and task completion time at different task loads.

Task loads	100	200	300	400	500
Miss rate (%) - hybrid	4	5	7	9	10
Compl. time (ms) - hybrid	85	90	95	100	105

+DVFSṪhe RL Scheduler showed a contention rate of 12%, reflecting its more sophisticated decision-making in task scheduling.

The scheduler overhead measures the computational cost of running the scheduling algorithm itself. Hybrid EDF exhibited a 4% overhead, which indicates that the dynamic scheduling mechanisms, including AI-specific adjustments and DVFS, added only a modest computational burden. The overhead for Baseline EDF was 3%, while the RL Scheduler had the highest overhead at 8% due to the complexity of reinforcement learning-based decision-making.

The fairness index, based on Jain's Fairness Index, was used to assess how evenly resources were distributed across tasks. Hybrid EDF achieved a fairness index of 0.97, close to the ideal value of 1.0, indicating fair resource allocation across all tasks. The Baseline EDF and RMS schedulers had lower fairness indices of 0.90 and 0.88, respectively, indicating a less equitable distribution of resources. The RL Scheduler achieved a fairness index of 0.92, reflecting its ability to manage resource allocation dynamically.

Table 4: AI workload accuracy degradation across schedulers.

Scheduler	B. EDF	RL Scheduler	Hybrid
Accuracy degr. (%)	0	2	3

Table 5: Mixed-criticality task deadline miss rate.

Scheduler	High-critic. MR (%)	Low-critic. MR (%)
B. EDF	12	30
G. EDF	8	25
RL	9	22
Hybrid	7	20

Energy-performance trade-off table displays the trade-off between the deadline miss rate and energy consumption for various schedulers. Hybrid EDF shows the lowest energy consumption with the least miss rate (see Table 2).

The trade-off between energy consumption and miss rate across six scheduling models is illustrated in Fig. 2, with experiment indices on the x-axis, miss rate (%) on the left y-axis, and energy consumption (Joules) on the right y-axis. Solid lines with markers represent miss rates, while dashed lines with 'x' markers indicate energy consumption. The models compared include Baseline EDF, RMS, Global EDF, Energy-Aware, RL Scheduler, and Hybrid EDF. Hybrid EDF consistently achieves the lowest miss rate, while Baseline EDF exhibits the highest, showing its inefficiency in handling scheduling tasks. Similarly, Hybrid EDF also consumes the least energy, whereas Baseline EDF has the highest energy consumption, making it less optimal for energy-sensitive environments. Hybrid EDF emerges as the most efficient model in terms of both energy savings and performance.

The scalability results table shows how deadline miss rates and task completion times scale as the task load increases. Hybrid EDF demonstrates stable performance as the load increases. Mixed-criticality task results compare the deadline miss rates for high-criticality and low-criticality tasks across different schedulers, highlighting how the Hybrid performs better for high-criticality tasks (Tables 3 and 5).

The AI workload accuracy degradation compares the accuracy degradation of the AI workloads due to pruning across different schedulers, showing that Hybrid EDF sacrifices some accuracy for better performance (see Table 4).

The scalability performance of six scheduling models is presented in Fig. 3, by analyzing the deadline miss rate and the average task completion time as the number of tasks increases. The miss rate percentage is included in Fig. 3a, where Baseline EDF exhibits the highest miss rate, while Hybrid EDF + DVFS maintains the lowest, closely followed by the RL Scheduler and Global EDF. On the other hand, Fig. 3b illustrates the average task completion time, where Baseline EDF again has the worst performance, while the Hybrid model demonstrates the best efficiency.

As the task count increases, all models show a rising trend in both metrics, indicating the impact of system load on scheduling effectiveness. The Energy-aware and RL-based models achieve a better balance between deadline adherence and task completion time than simpler algorithms like Baseline EDF and RMS principles. The hybrid model consistently outperforms others, proving its robustness in handling increasing workloads while minimizing deadline misses and maintaining low task completion times.

Next, Fig. 4 compares the miss rates of six scheduling models for high-criticality and low-criticality tasks. Fig. 4a presents the high-criticality miss rate, where Baseline EDF has the highest rate, followed by Energy-Aware and RMS while Hybrid EDF achieves the lowest miss rate, demonstrating its reliability in handling critical tasks. Fig. 4b shows the low-criticality miss rate, where Baseline EDF again performs the worst, followed closely by RMS while Hybrid EDF + DVFS maintains the lowest miss rate.

Across both categories, Global EDF, RL Scheduler, and Energy-Aware models achieve a middle-ground balance, offering moderate improvements over traditional methods. The decreasing trend from Baseline EDF to Hybrid EDF + DVFS in both plots highlights the effectiveness of advanced scheduling techniques in improving deadline adherence. The hybrid model proves to be the most efficient approach, as it minimizes the likelihood of missing deadlines, ensuring better overall performance for both critical and non-critical tasks.

5 CONCLUSIONS

This study presents a detailed evaluation of various scheduling algorithms, with a particular focus on the proposed Hybrid EDF + DVFS scheduler. The research explores how real-time task scheduling can be optimized for performance and energy efficiency, especially in AI workloads such as neural network inference for medical data.

The results indicate that the Hybrid EDF + DVFS scheduler consistently outperforms traditional approaches like Baseline EDF and RMS across multiple key performance metrics. The Hybrid scheduler achieved the lowest deadline miss rate (4%), reflecting its ability to prioritize tasks effectively while ensuring critical deadlines are met. This improvement is largely due to its dynamic task management capabilities, including model pruning for AI tasks and adaptive frequency scaling through DVFS. In terms of task completion time, the Hybrid EDF + DVFS scheduler also outperformed the other algorithms, completing tasks in an average of 90 ms compared to 120 ms for Baseline EDF. This indicates that Hybrid EDF + DVFS can accelerate task execution without compromising deadline adherence, making it a valuable approach for real-time AI applications such as medical diagnostics. One observed trade-off was the accuracy degradation of AI models due to pruning (4%).

For further experiments, it is important to integrate multiple CNN models. Testing multiple models helps determine which architecture best fits the hybrid scheduling framework.

Aknowledgements

The research conducted in this article was partially supported by the UVT 1000 Develop Fund of the West University of Timișoara. This research was partially supported by the MOISE infrastructure (grant number SMIS 124562, financed by European structural funds and Romanian government funds).[1]

[1]https://moise.projects.uvt.ro/, https://hpc.uvt.ro/

REFERENCES

[1] Kholoud Alatoun, Khaled Matrouk, Mazin Abed Mohammed, Jan Nedoma, Radek Martinek, and Petr Zmij. 2022. A novel low-latency and energy-efficient task scheduling framework for internet of medical things in an edge fog cloud system. *Sensors* 22, 14 (2022), 5327.

[2] Matthew Andrews. 2000. Probabilistic end-to-end delay bounds for earliest deadline first scheduling. In *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064) (INFCOM-00, Vol. 2)*. IEEE, 603–612. https://doi.org/10.1109/infcom.2000.832234

[3] Noel CF Codella, David Gutman, M Emre Celebi, Brian Helba, Michael A Marchetti, Stephen W Dusza, Aadi Kalloo, Konstantinos Liopyris, Nabin Mishra, Harald Kittler, et al. 2018. Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic). In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*. IEEE, 168–172.

[4] Manjari Gupta, Lava Bhargava, and S Indu. 2021. Dynamic workload-aware DVFS for multicore systems using machine learning. *Computing* 103 (2021), 1747–1769.

[5] Jayant R Haritsa, Miron Livny, and Michael J Carey. 1991. *Earliest deadline scheduling for real-time database systems*. Technical Report. University of Wisconsin-Madison Department of Computer Sciences.

[6] Dacre Knight, Christopher A Aakre, Christopher V Anstine, Bala Munipalli, Parisa Biazar, Ghada Mitri, Jose Raul Valery, Tara Brigham, Shehzad K Niazi, Adam I Perlman, et al. 2023. Artificial Intelligence for Patient Scheduling in the Real-World Health Care Setting: A Metanarrative Review. *Health Policy and Technology* (2023), 100824.

[7] John Lehoczky, Lui Sha, and Yuqin Ding. 1989. The rate monotonic scheduling algorithm: Exact characterization and average case behavior. In *RTSS*, Vol. 89. 166–171.

[8] Giuseppe Massari, Federico Terraneo, Michele Zanella, and Davide Zoni. 2018. Towards fine-grained DVFS in embedded multi-core CPUs. In *Architecture of Computing Systems–ARCS 2018: 31st International Conference, Braunschweig, Germany, April 9–12, 2018, Proceedings 31*. Springer, 239–251.

[9] Tasquia Mizan and Sharareh Taghipour. 2022. Medical resource allocation planning by integrating machine learning and optimization models. *Artificial Intelligence in Medicine* 134 (2022), 102430.

[10] Senthil Murugan Nagarajan, Ganesh Gopal Devarajan, Amin Salih Mohammed, TV Ramana, and Uttam Ghosh. 2022. Intelligent task scheduling approach for IoT integrated healthcare cyber physical systems. *IEEE Transactions on Network Science and Engineering* 10, 5 (2022), 2429–2438.

[11] Rishov Sarkar, Stefan Abi-Karam, Yuqi He, Lakshmi Sathidevi, and Cong Hao. 2023. FlowGNN: A dataflow architecture for real-time workload-agnostic graph neural network inference. In *2023 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 1099–1112.

[12] Teck Yan Tan, Li Zhang, and Chee Peng Lim. 2019. Intelligent skin cancer diagnosis using improved particle swarm optimization and deep learning models. *Applied Soft Computing* 84 (2019), 105725.

[13] Boris I Tymchenko, Filip A Marchenko, Evgeniy N Khvedcheniya, and Dmitriy V Spodarets. 2020. Classification of skin lesions using multi-task deep neural networks. *Vestnik sovremenjih informacionjih tehnologiji* 3, 3 (2020), 136–148.

[14] Abhishek Verma, Ludmila Cherkasova, Vijay S Kumar, and Roy H Campbell. 2012. Deadline-based workload management for MapReduce environments: Pieces of the performance puzzle. In *2012 IEEE Network Operations and Management Symposium*. IEEE, 900–905.

[15] Chen-Xin Wu, Min-Hui Liao, Mumtaz Karatas, Sheng-Yong Chen, and Yu-Jun Zheng. 2020. Real-time neural network scheduling of emergency medical mask production during COVID-19. *Applied Soft Computing* 97 (2020), 106790.