TASK Quarterly 30 (1) 2026

Comparison of HPC and Cloud Environments for Developing Energy Efficient Services

Paweł Czarnul [1,2*], **Oksana Diakun** [1,2], **Grzegorz Koszczał** [1,2], **Adam Krzywaniak** [2], **Jerzy Proficz** [1,2†], **Piotr Sokołowski** [1,2]

[1]Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
[2]Centre of Informatics – Tricity Academic Supercomputer and networK (CI TASK), Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland

[*]pawel.czarnul@pg.edu.pl [†]jerzy.proficz@pg.edu.pl

All authors contributed equally to this work and are listed in alphabetical order by surname.

https://doi.org/10.34808/tq2026/30.1/b

Abstract

The paper presents the characteristics and a comparison of HPC and cloud computing environments in the context of building and deploying services that support energy efficiency. The analysis covers typical hardware configurations, types of workloads (taking into account domain-specific features, application and software requirements), as well as the characteristics of use cases based on CPUs, GPUs, and their combinations. The scale of resource utilization and task execution performance in both environments is also discussed. The authors consider these aspects in the context of the potential migration and adaptation of the DEPO (Dynamic Energy–Performance Optimizer) software from an HPC environment to a cloud environment, with particular emphasis on performance-energy optimization. The paper also discusses the limitations resulting from differences between these environments and presents the hardware-software configurations in the cloud environment supported by the proposed cloud-based variant of the DEPO software.

Keywords:

high-performance computing, energy-efficient computing

1. Introduction

In recent years, the rapid development of computational technologies has led to a clear convergence of two data processing paradigms — High Performance Computing (HPC) environments and cloud environments. Traditionally, HPC systems have been used primarily in areas requiring very high computational power, such as scientific simulations, large-scale data analysis, or modeling of complex physical phenomena. These environments are characterized by high efficiency, scalability, and dedicated hardware infrastructure optimized for intensive computational tasks.

Cloud environments, on the other hand, originally developed mainly to provide flexible IT resource sharing in the service model (IaaS), are increasingly gaining importance in high-performance computing applications as well. The development of cloud infrastructure, integration with graphics cards, availability of specialized instances, and increased reliability of services make cloud computing a real alternative to traditional supercomputers in selected scenarios. At the same time, the cloud enables flexible resource scaling, rapid service deployment, and reduced capital costs.

Alongside the growing demand for computational power, the importance of energy efficiency issues are also increasing. Both HPC centers and cloud data centers consume significant amounts of electricity, which translates into operational costs and environmental impact. In this context, solutions that support energy consumption monitoring, management, and optimization are gaining importance. Integrating energy-efficient services with existing computing environments represents an important step toward the sustainable development of computational infrastructure.

The aim of this article is to compare High Performance Computing (HPC) and cloud environments from the perspective of building and deploying energy-efficiency-oriented services. Rather than focusing solely on performance characteristics, the comparison emphasizes architectural, operational, and management aspects that directly influence the design and effectiveness of energy-aware monitoring, control, and optimization mechanisms.

In particular, the paper analyzes how differences in hardware architecture, software stack organization, resource allocation models, and execution semantics affect the availability and usability of energy-related metrics, the ability to control power and performance parameters, and the feasibility of deploying continuous energy-performance optimization services. While HPC environments typically rely on tightly coupled, bare-metal execution with low-level access to hardware interfaces, cloud environments introduce virtualization, abstraction layers, and elastic resource management, which significantly alter the optimization space.

These considerations are discussed in the context of migrating and adapting the DEPO [1, 2] (Dynamic Energy–Performance Optimizer) tool from an HPC-centric execution model toward cloud and hybrid HPC-cloud environments. The paper highlights both the opportunities and limitations arising from this transition, including constraints on hardware visibility, deployment models, and orchestration mechanisms, as well as new possibilities enabled by cloud-native service architectures.

The structure of the paper is as follows. Section 2 reviews related work on energy-efficient computing in HPC, cloud, and hybrid environments. Section 3 compares the key characteristics of HPC and cloud platforms relevant to energy-aware service design. Section 4 discusses mechanisms and services supporting energy efficiency at the infrastructure, orchestration, and application levels. Finally, Section 5 summarizes the findings and outlines directions for future work.

2. Related work

As an example of research focusing on cloud environments, Tian et al. [3] analyze the relationship between performance and power consumption in heterogeneous cloud data centers. The study formulates job scheduling and service rate selection as an optimization problem that minimizes a weighted sum of average response time and power usage. Server performance is modeled using queuing theory, while power consumption includes both static and dynamic components dependent on the service rate. The authors examine systems with fixed service rates as well as DVFS-enabled servers with discrete frequency levels, reflecting practical processor constraints. To support scalability, a distributed optimization approach based on dual decomposition is proposed, allowing scheduling and speed-scaling decisions to be made locally at individual servers. Experimental results show that DVFS-aware scheduling can reduce power consumption by up to 50% while maintaining controllable performance degradation.

While the previous work focuses on energy-performance trade-offs in cloud data center services, Bukhari shifts the focus toward energy-aware scheduling of data-intensive scientific workflows in large-scale HPC environments [4]. The work focuses on reducing turnaround time and energy consumption by addressing I/O bottlenecks through speculative and locality-aware scheduling techniques. An event-based system is proposed to proactively prefetch data to compute nodes

based on historical execution patterns, using Bayesian inference to anticipate future tasks. In addition, the thesis explores energy modeling through machine learning, employing variational autoencoders to analyze the relationship between system configurations, performance, and energy usage. The proposed methods are evaluated using real scientific workflows, demonstrating the interaction between scheduling strategies, data access behavior, and energy efficiency in high-performance data analysis environments.

Moving further into HPC-oriented energy optimization, Abdelhafez et al. examine energy–performance trade-offs in heterogeneous high-performance systems using parallel genetic algorithms executed on hybrid servers [5]. The authors propose a Heterogeneous Parallel Island Genetic Algorithm (HPIGA) designed to exploit multicore CPUs and multiple GPU accelerators concurrently through workload distribution. The study formulates a bi-objective optimization problem targeting execution time and energy consumption, with the number of islands assigned to each device as the decision variable. Experimental evaluation is conducted on an HPC-class hybrid server equipped with an Intel Icelake CPU and NVIDIA A40 GPUs, using system-level power measurements. The results are analyzed through Pareto fronts, illustrating trade-offs between performance and energy efficiency across different parallel execution strategies.

Focusing on power-constrained HPC systems, Kumbhare et al. address energy-aware scheduling through dynamic power management for value-oriented schedulers [6]. The study builds on the concept of time-dependent job-value functions, where system productivity is defined as the accumulated value of completed jobs rather than raw performance metrics such as FLOPS. The authors analyze limitations of existing static power allocation strategies under varying system-wide power constraints, highlighting inefficiencies in resource utilization and job completion. To address these issues, they propose dynamic power reallocation strategies that adjust the power budgets of running jobs to accommodate newly arriving high-value workloads. The approach relies on application-specific power–performance models and is evaluated through large-scale simulations using real HPC workload traces, demonstrating improvements in system productivity, resource utilization, and job completion rate under constrained power budgets.

Complementing HPC-oriented studies, Kumar and Pandey focus on energy-efficient resource management in cloud data centers through dynamic virtual machine consolidation [7]. The work addresses the trade-off between reducing energy consumption and maintaining service level agreements in virtualized IaaS environments. An adaptive overload detection technique is proposed, where upper utilization thresholds are dynamically adjusted based on statistical analysis of historical CPU usage. The approach combines this mechanism with established VM selection and power-aware placement policies to migrate workloads and switch idle hosts into low-power states. The proposed method is evaluated using CloudSim simulations with real-world PlanetLab workload traces, demonstrating reduced energy consumption while maintaining acceptable SLA violation levels.

Bridging cloud and HPC paradigms, Nguyen-Duc et al. address energy-aware virtual machine allocation in HPC cloud environments through a hybrid heuristic and machine learning approach [8]. The work formulates VM placement as a multidimensional vector bin packing problem, considering multiple resource dimensions as well as VM execution time. The authors introduce BFD-NN, which extends the Best Fit Decreasing heuristic with an evaluation function whose coefficients are dynamically adapted using a combination of parallel genetic algorithms and neural networks trained on historical workload data. This approach enables the scheduler to account for both resource utilization efficiency and temporal alignment of VM lifetimes. Simulation-based evaluation using parallel workload traces shows that BFD-NN reduces total energy consumption compared to classical bin packing heuristics, while maintaining comparable scheduling complexity.

Taken together, the reviewed works highlight that energy efficiency has been extensively studied in both cloud and HPC environments, but largely through domain-specific perspectives. Cloud-oriented studies primarily focus on virtualization, VM consolidation, and service-level trade-offs, emphasizing elastic resource management and workload variability. In contrast, HPC-focused research addresses power constraints, job scheduling, and application-level performance–energy trade-offs, often assuming tightly coupled workloads and static resource allocations. Hybrid approaches, such as HPC clouds and heterogeneous platforms, attempt to bridge these paradigms by combining virtualization with HPC scheduling and accelerator-aware execution. However, existing studies typically evaluate energy efficiency within a single computing model and under distinct assumptions. A systematic comparison of HPC and cloud environments from the perspective of developing energy-efficient services remains limited, motivating the need for an analysis that contrasts their architectural characteristics, resource management mechanisms, and energy optimization opportunities.

Table 1: HPC vs. Cloud Computing Comparison

Feature	HPC (High Performance Computing)	Cloud Computing
Primary Goal	Solving one complex problem very fast.	Managing many tasks for many users.
Connectivity	Tightly coupled (extremely fast networking).	Loosely coupled (standard networking).
Resource Use	Runs directly on "bare metal" hardware.	Uses Virtual Machines (VMs) or containers.
Cost Model	Usually fixed (owned/leased clusters).	Pay-as-you-go (utility model).

3. Characteristics of HPC and Cloud environments

There are many different types of environments which provide massive computing power. They have many characteristics to meet a variety user needs. Some of them, like HPC, are focused on performance, defined as short and constant latency, while in others, such as cloud computing, more factors are taken into account, some of which are measurable, such as availability time or scalability, while some of them are not directly measurable, for example security, flexibility or even ease of use. The latter need to be defined by proper characteristics and in turn metrics indirectly. In general, HPC systems are designed to meet one well-defined goal, but in cloud computing the exact goal depends strongly on the user needs. In Table 1, we present the main characteristics of both environments, and in the following sections we describe them further with focus on the aspects that are the most important in terms of energy optimization.

3.1. HPC environment

Minimizing latency entails a few important design decisions. First of all, the hardware is dedicated to one purpose, so there is no need to share resources between different users or applications. The user usually pays for the explicit hardware, so there no simple way to change it. This means that if some optimization is need, only the parameters can be adjusted. This necessitates the use of low level mechanisms to control hardware behavior, like DVFS or power capping.

3.2. Cloud environment

Virtualization allows physical resources to be separated from the users, so the hardware assigned to the user can easily be changed. Therefore, cloud providers usually have a few hardware configurations dedicated to different types of workloads, so the main problem is to detect what type of workload the user actually runs and assign the most suitable hardware configuration. This is especially important from the energy optimization point of view, because the other hardware configurations may have distinct efficiency for various types of workloads. The providers should decide where to place the workload, taking into account service requirements, and also their costs, like for example energy consumption. To make this problem a little bit easier, available hardware parameters are usually fixed.

4. Mechanisms and services supporting energy efficiency

4.1. Infrastructure level

Even if at other levels there are not any power management mechanisms, at this level there are almost always some mechanisms that can be used to control power consumption. Moreover, these mechanisms cannot be disabled, because they are usually used to protect hardware from overheating, so the only possibility to optimize them is to adjust their settings. These are commonly used by various services in the upper levels, like orchestration or even application. These services need to have access to hardware metrics, which provide information about power consumption and often also performance, so various interfaces are provided. These are usually vendor specific APIs, such as Data Center GPU Manager (DCGM) for NVIDIA GPUs, or Host System Management Port (HSMP) for AMD CPUs, but there are also some somewhat universal solutions, like Running Average Power Limiter (RAPL) for Intel CPUs, which is also partially supported by AMD processors or even perf API for Linux, which supports various CPU metrics on different architectures.

4.2. Orchestration level

Solutions at this level are usually related to scheduling. In the HPC environment, there are plugins for popular schedulers like SLURM which allow power consumption of jobs to be optimized. One example of this type of plugin is Energy Aware Runtime [9–11]. For cloud built

with Kubernetes, there is, for example, Kepler [12]. Additionally, OpenStack has some energy saving strategies implemented, for example in the Watcher service[1]. This is a very basic solution, it simply tries to put all the workloads on the smallest number of nodes, and switch off the rest. It does not require any energy measurements, so there is no need to have access to any specialized hardware capabilities. However, the rest of the solutions at this level usually require some energy related metrics to make correct decisions and APIs to set power configurations. These metrics can be obtained from the infrastructure level, sometimes directly, but sometimes with the help of agents or telemetry systems like Prometheus or Telegraf.

4.3. Applications / services level

At this layer, there are many valuable metrics that can be used to optimize energy efficiency. For example, in the cloud environment are metrics related to SLA. Naturally, in some situations, power consumption can be reduced without breaking the SLA agreement, but in some cases the situation is the opposite, when in order to comply with the SLA, more power is needed. The latter case can happen when hardware or operating system level power management mechanisms have not got enough information to make correct decisions [13]. This solution can be adapted to various applications in the cloud environment, like web services, if metrics can be provided that reveal information about the actual performance of the service.

Another solution, one that is more dependent on the application type, is to manipulate the operation order. For example, in machine learning workloads, there are various computations that can be done in a different order, and in some cases this can cause significant power savings [14].

5. Conclusion and future work

In this paper, we analyzed and compared High Performance Computing (HPC) and cloud environments from the perspective of designing, deploying, and operating energy-efficient services. The comparison focused on architectural characteristics, resource management models, and available software and hardware mechanisms that directly affect the feasibility and effectiveness of energy-aware monitoring, control, and optimization. Particular attention was paid to differences in hardware visibility, execution models, and deployment constraints, which play a critical role in energy–performance optimization.

The discussion was grounded in the context of migrating and extending the DEPO [1] (Dynamic Energy–Performance Optimizer) tool from a traditional HPC environment toward cloud and hybrid HPC–cloud infrastructures. While HPC systems provide direct access to low-level power and performance metrics and enable fine-grained control through mechanisms such as DVFS and power capping, cloud environments introduce additional abstraction layers that limit hardware visibility but offer greater flexibility in resource allocation, scalability, and service-oriented deployment models. These differences necessitate distinct design choices when developing energy-aware services for each environment.

The analysis showed that effective energy-performance optimization in cloud environments requires a combination of infrastructure-level telemetry, orchestration-aware mechanisms, and application- or service-level performance indicators. In this context, cloud-native monitoring, scheduling, and deployment frameworks play a crucial role in compensating for reduced access to low-level hardware controls, while also enabling continuous and scalable optimization across heterogeneous workloads.

Future work will focus on extending the presented analysis in several directions. Firstly, the cloud-enabled variant of DEPO will be evaluated and adapted to additional cloud infrastructures, software stacks, and deployment configurations beyond the CI TASK environment. Secondly, further research will address hybrid optimization scenarios, particularly coordinated CPU–GPU energy–performance tuning, which remains a challenge in both HPC and cloud settings. Finally, tighter integration with cloud orchestration mechanisms and higher-level service metrics is envisaged, enabling more adaptive and context-aware energy optimization strategies suitable for large-scale, heterogeneous computing platforms.

[1] https://docs.openstack.org/watcher/latest/strategies/saving_energy.html

Acknowledgments

The research was supported [in part] by project “Cloud Artificial Intelligence Service Engineering (CAISE) platform to create universal and smart services for various application areas”, No. KPOD.05.10-IW.10-0005/24, as part of the European IPCEI-CIS program, financed by NRRP (National Recovery and Resilience Plan) funds.

References

[1] A. Krzywaniak, P. Czarnul, and J. Proficz, “Depo: A dynamic energy-performance optimizer tool for automatic power capping for energy efficient high-performance computing,” *Software: Practice and Experience*, vol. 52, no. 12, pp. 2598–2634, 2022.

[2] D. Szmidka, A. Krzywaniak, P. Czarnul, and J. Proficz, “Dynamic

energy-performance optimization tool extension with periodic power cap tuning," in *2025 IEEE 31th International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 1–4, 2025.

[3] Y. Tian, C. Lin, and K. Li, "Managing performance and power consumption tradeoff for multiple heterogeneous servers in cloud computing," *Cluster computing*, vol. 17, no. 3, pp. 943–955, 2014.

[4] S. AZ Bukhari, *Energy-Aware Speculative Scheduling*. PhD thesis, University of Leicester, 2025.

[5] A. Abdelhafez, R. R. Manumachu, and A. Lastovetsky, "Parallel genetic algorithms on hybrid servers: Design, implementation, and optimization for performance and energy," *Swarm and Evolutionary Computation*, vol. 98, p. 102110, 2025.

[6] N. Kumbhare, A. Akoglu, A. Marathe, S. Hariri, and G. Abdulla, "Dynamic power management for value-oriented schedulers in power-constrained hpc system," *Parallel Computing*, vol. 99, p. 102686, 2020.

[7] S. Kumar and M. Pandey, "Energy aware resource management for cloud data centers," *International Journal of Computer Science and Information Security*, vol. 14, no. 7, p. 844, 2016.

[8] T. Nguyen-Duc, N. Quang-Hung, and N. Thoai, "Bfd-nn: best fit decreasing-neural network for online energy-aware virtual machine allocation problems," in *Proceedings of the 7th Symposium on Information and Communication Technology*, pp. 243–250, 2016.

[9] J. Corbalan, L. Alonso, J. Aneas, and L. Brochard, "Energy optimization and analysis with ear," in *2020 IEEE International Conference on Cluster Computing (CLUSTER)*, pp. 464–472, 2020.

[10] J. Corbalan, O. Vidal, L. Alonso, and J. Aneas, "Explicit uncore frequency scaling for energy optimisation policies with ear in intel architectures," in *2021 IEEE International Conference on Cluster Computing (CLUSTER)*, pp. 572–581, 2021.

[11] J. Corbalan, L. Alonso, C. Navarrete, and C. Guillen, "Soft cluster powercap at supermuc-ng with ear," in *2022 IEEE 13th International Green and Sustainable Computing Conference (IGSC)*, pp. 1–8, 2022.

[12] S. Choochotkaew, T. Chiba, M. Amaral, R. Nakazawa, S. Trent, E. K. Lee, U. Devi, T. Eilam, and H. Chen, "Best-effort power model serving for energy quantification of cloud instances," in *2024 32nd International Conference on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, pp. 1–4, IEEE, 2024.

[13] V. Anagnostopoulou, M. Dimitrov, and K. A. Doshi, "Sla-guided energy savings for enterprise servers," in *2012 IEEE International Symposium on Performance Analysis of Systems & Software*, pp. 120–121, 2012.

[14] J.-W. Chung, Y. Gu, I. Jang, L. Meng, N. Bansal, and M. Chowdhury, "Reducing energy bloat in large model training," in *Proceedings of the ACM SIGOPS 30th Symposium on Operating Systems Principles*, SOSP '24, (New York, NY, USA), p. 144–159, Association for Computing Machinery, 2024.