Survey on Energy Consumption Optimization Approach in Container-Based Cloud Environments

Francesco Pace Napoleone[1], Marian-Catalin Mutu-Costan[1], Ihor Ryzhuk[1], and Ewan Soueidan[1]

Université de Toulouse, France
{francesco.pace-napoleone, marian.mutu-costan, ihor.ryzhuk, ewan.soueidan}@univ-tlse3.fr

Abstract. Growing demand for cloud services has shifted data center design from performance-centric to energy-efficient architectures. Containerization, a lightweight alternative to virtual machine, enables flexible resource allocation and reduced overhead. This survey reviews energy optimization techniques in container-based cloud environments by systematically selecting studies from 2010-2020 and categorizing them by scheduling scope, optimization method, energy-saving strategy, and Docker maturity. Our analysis covers early VM consolidation methods that achieved up to 83% energy savings, predictive scheduling, and advanced container orchestration. Overall, container-based optimizations, particularly those leveraging AI and hybrid heuristics, significantly reduce energy consumption while maintaining Quality of Service (QoS), pointing to promising future research directions.

Keywords: Cloud Computing · Containerization · Energy Optimization · Microservices · Resource Allocation

1 Introduction

1.1 From Monoliths to Containers: Energy Implications

Cloud computing evolved to support scalable platform usage, moving from single-file applications to client/server architectures with monolithic backends, then to microservices with containers, and now transitioning to micro-frontends. Containers run directly on the host using Linux CGroups, bypassing the hypervisor overhead inherent in VMs (e.g., Proxmox, VMWare), which allows for full utilization of host CPU resources and simplifies telemetry monitoring. This lightweight nature is evident in Docker files that simply copy files and run builds, making containers easier to integrate into CI/CD pipelines.

However, this increased reliance on cloud-based infrastructures has led to a significant rise in energy consumption. In 2006, the electricity costs for IT infrastructures in the United States alone were estimated at $4.5 billion [4]. Energy consumption optimization has since become a critical concern, especially as cloud data centers now account for approximately 1-1.5% of global electricity use [11]. Despite efficiency improvements, the demand for digital services continues to grow, pushing the need for more sustainable solutions.

1.2 Energy-Aware Containerized Cloud Environments

As energy consumption in cloud infrastructures continues to rise, optimizing energy efficiency through containerized environments has become increasingly important. Early research focused on heuristic-based approaches to optimize virtual machine (VM) placement, achieving energy savings of up to 83% while maintaining only a 1.1% service level agreement (SLA) violation rate [4]. More recently, research has shifted from VM-based allocation towards containerized environments, where energy efficiency is influenced by scheduling strategies, workload distribution, and infrastructure optimizations.

The paper "Survey on Energy Consumption Optimization Approach in Container Based Cloud Environments" further highlights that containerization not only drives scalability and reproducibility but also plays a crucial role in optimizing energy consumption. It explores strategies for efficient resource allocation, reducing power overhead, and ensuring that the benefits of container-based deployments extend beyond performance to sustainability in cloud infrastructures. In this work, we present a state-of-the-art review on Energy Consumption Optimization Approaches in Container-Based Cloud Environments. Our survey of the available literature—predominantly spanning from 2010 to 2020—reveals that foundational research primarily focused on energy measurement, basic optimization strategies, and energy visualization techniques [4].

2 Literature Review Approach

The research process began by defining our scope together with our supervisor, Professor Jean-Marc Pierson. We focused on creating a state-of-the-art review addressing energy consumption optimization within Container and Cloud Computing.

Initially, we defined a structured "Search Pipeline" illustrated in Figure 1, ensuring systematic identification and evaluation of relevant papers.

Fig. 1. Search Pipeline Flowchart

We developed an initial search query: *((energy OR resource) AND container)*, including "resource" to account for better resource utilization directly influencing energy efficiency.

Clear exclusion criteria were established beforehand (Table 1), filtering out irrelevant work and focusing exclusively on impactful research.

Direct Exclusion of Paper If
Work is not in English
Work is not a scientific paper
Work has fewer than 10 citations
Work is not related to containers/cloud

Table 1. Exclusion Criteria

This approach yielded 34 relevant papers, sourced across various publishers, as depicted in Figure 2.

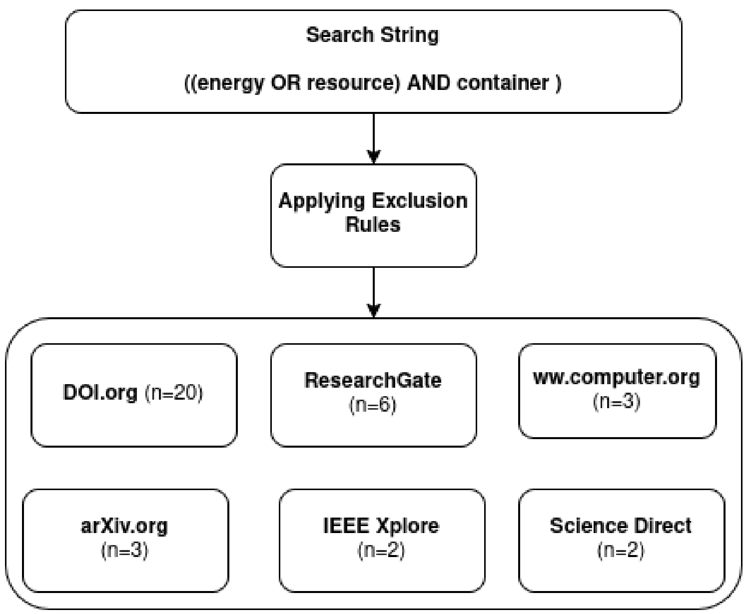

Fig. 2. Distribution of Exported Papers by Publisher

The majority of selected studies were published between 2017 and 2019 (Figure 3), reflecting a peak period of research interest.

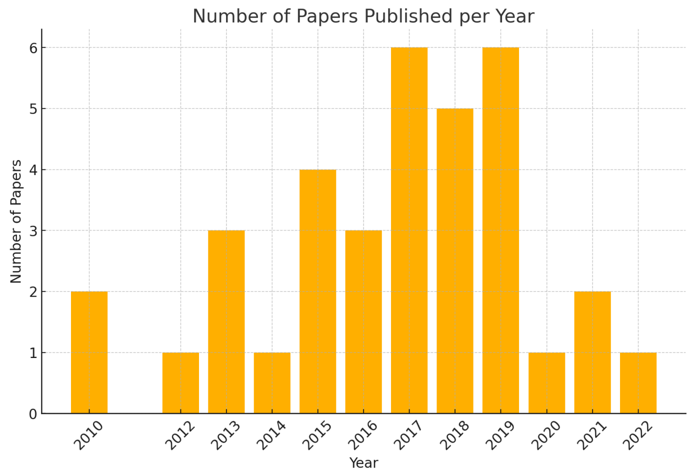

Fig. 3. Papers per Year

After paper selection, we defined specific classification metrics (Table 2) to analyze the technical dimensions of energy-aware scheduling approaches.

Classification Categories
Optimization Technique
Target Level
Scheduling Scope
Energy Strategy
Docker Awareness

Table 2. Paper Classification Metrics

The classification categories are defined as follows:

- **Scheduling Scope**
  - **Online**: Decisions at runtime.
  - **Offline**: Known workload beforehand.
  - **Predictive**: Forecast-based decisions.
  - **Reactive**: Triggered by SLA violations or thresholds.
  - **Concurrent**: Simultaneous multiple-task scheduling.
- **Optimization Technique**
  - **Greedy**: Stepwise local decisions.
  - **Model-Based**: Predictive analytical models.
  - **PSO**: Bio-inspired global optimization.
  - **Evolutionary**: Genetic/evolution-based methods.
  - **Meta-Heuristic**: High-level heuristic strategies.
  - **MCMF**: Flow-network optimization.

- **Energy Strategy**
  - **Consolidation**: Migrate to reduce active hosts.
  - **Brownout**: Temporarily disable app components.
  - **QoS Maintenance**: Energy-saving without SLA violations.
  - **Renewable-aware**: Optimize for renewable energy availability.
- **Docker Awareness**
  - **Pre-Docker**: Pre-containerization (VM-only).
  - **Early Docker**: Containers used like VMs.
  - **Mature Docker**: Container-specific optimizations.

We incorporated **Docker Awareness** to highlight the technological transition from virtual machines to containers, essential for understanding changes in scheduling complexity. **Energy Awareness** is included to represent approaches' capability to balance energy efficiency with QoS constraints, recognizing the different trade-offs between performance and sustainability.

Finally, Tables 3 and 4 summarize our classified selection, and Figure 4 visually correlates scheduling approaches with their publication years.

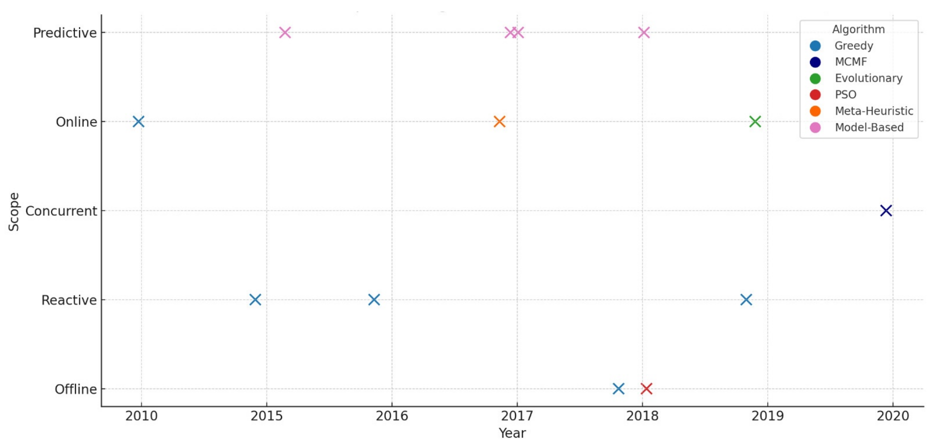

Fig. 4. Trend Analysis

Paper Ref.	Optimization Technique	Target Level	Scheduling Scope	Energy Strategy	Docker Awareness
Availability-Aware Scheduler [1] (2018)	Greedy	Container, VM	Offline	Availability	Mature Docker
Renewable-Aware Scheduling [12] (2019)	Greedy	Container, VM	Reactive	Consolidation	Mature Docker
Concurrent Container Scheduling [10] (2020)	MCMF	Container	Concurrent	Consolidation	Mature Docker
GP Hyper-Heuristic for Containers [17] (2019)	Evolutionary	Container, VM	Online	Consolidation	Mature Docker

Paper Ref.	**Optimization Technique**	**Target Level**	**Scheduling Scope**	**Energy Strategy**	**Docker Awareness**
PSO-Based Container Consolidation [16] (2018)	PSO	Container	Offline	Consolidation	Mature Docker
Energy-Efficient Container Framework [15] (2015)	Greedy	Container, VM	Reactive	Consolidation	Early Docker
VM Consolidation Strategy [6] (2017)	Meta-Heuristic	VM	Online	Consolidation	Mature Docker
Predictive Resource Provisioning [7] (2015)	Model-based	VM	Predictive	Consolidation	Early-Docker
Dynamic VM Reallocation [4] (2010)	Greedy	VM	Online	Consolidation	Pre-Docker
Autonomic Container Management [3] (2017)	Model-based	Container, VM	Predictive	QoS	Mature Docker
Brownout-Based Scheduling [18] (2016)	Greedy	Application Layer	Reactive	Brownout	Mature Docker
GPR + Convex VM Planning [5] (2017)	Model-Based	VM, PM	Predictive	Consolidation	Mature Docker
SLA-Aware Consolidation [13] (2018)	Model-Based	VM, PM	Predictive	Consolidation	Mature Docker

Table 3: Classification of Technical Papers on Energy-Aware Scheduling

Paper Ref.	**Type**	**Scope / Focus**
Survey on Energy-Aware Scheduling [8]	Survey	Overview of VM and container-level techniques
Global Energy Estimates [14]	Analysis	Environmental impact of data centers
Cloud Impact on Energy [9]	Report	Energy use in European data centers
IEA Report on Data Centers [11]	Report	Global energy trends in ICT and data centers
EU Data Center Energy Trends [2]	Analysis	EU-level trends in ICT energy consumption

Table 4. Survey and Analysis Papers

3 Trend reconstruction

3.1 Early approaches

Early efforts in energy-aware cloud infrastructure trace back to 2010, with Docker's release in 2013 marking a pivotal point for containerization. Even earlier, in 2006, electricity consumption by IT infrastructure in the US was estimated at $4.5 billion and projected to double by 2011 [4]. These cost concerns initiated research into energy-efficient resource management.

Initial studies, such as by Beloglazov and Buyya [4], applied bin packing heuristics with dynamic voltage and frequency scaling (DVFS) using the CloudSim simulator. Their methods achieved energy savings of up to 83% while maintaining SLA violation rates under 1.1%, laying the foundation for dynamic VM consolidation strategies.

By 2015, the research focus shifted toward predictive modeling to overcome the rigidity of static thresholding. Dabbagh et al. [7] introduced a Wiener filter-based workload predictor alongside Best Fit Decreasing allocation. This approach improved energy efficiency by up to 33% over heuristic methods.

In 2017, Bui et al. [5] advanced this with Gaussian Process Regression (GPR) for non-stationary workload prediction, optimized using the Fast Fourier Transform. Coupled with convex optimization for VM migration, this approach achieved 35% energy savings, with only a modest 15% latency increase.

In parallel, other strategies emerged: brownout-based scheduling [18], SLA-aware provisioning [13], and energy-visualization frameworks, all contributing to a broader, more nuanced approach to energy optimization [6, 8].

3.2 From Virtual Machines to Containers

As the field matured, attention shifted from VM-centric models toward container-based deployments. Containers offer lower overhead, faster startup times, and better resource packing, enabling finer-grained resource management compared to traditional VMs [1].

Early contributions in this space, such as those by Piraghaj et al. [15], explored container consolidation for energy savings. Later, more sophisticated methods emerged. Alahmad et al. [1] introduced availability-aware scheduling using modular watchdogs and Pearson correlation for anomaly detection. Hu et al. [10] proposed framing container scheduling as a minimum-cost flow problem, scalable to over 5000 nodes.

Tan et al. [17] combined evolved genetic heuristics with human-designed rules to solve two-level placement (container to VM, VM to host), accounting for reliability metrics like MTTF and MTTR. These hybrid systems improved both energy efficiency and fault tolerance.

Meanwhile, broader infrastructure-aware techniques appeared. Renewable-aware scheduling [12], optimization-based container consolidation [16], and DevOps-driven elastic container management [3] highlighted the expanding scope of energy-centric design.

3.3 Recent Shifts and Outlook

Beyond technical scheduling innovations, researchers increasingly examine broader systemic and policy-level trends. Studies like those by Avgerinou et al. [2] emphasize the macro-scale impact of data center energy use, while recent data from IEA and Masanet et al. [9, 11, 14] suggest a growing need for integrated, user-centric energy management systems.

This evolution—from basic heuristics and VM scheduling to hybrid AI-driven, reliability-aware container systems—reflects a maturing field. The latest focus centers on adaptability, sustainability, and predictive intelligence, paving the way for resilient, energy-optimized cloud ecosystems.

4 Method Analysis

4.1 Methodology

Over time, different researchers have utilized various testing environments and hardware configurations to evaluate their energy-aware scheduling algorithms. To examine the evolution of their experimental approaches, we have collected and structured data from the selected papers, focusing on the simulation environment, the number of CPU Cores, memory capacity, and power consumption of the experimentation hardware. Specifically, we define:

- **P idle (W)** as the power consumption at low CPU utilization (generally below 50% load).
- **P max (W)** as the power consumption at maximum CPU load.

In some studies, multiple hardware configurations were reported—either due to hardware availability or to assess performance on more powerful setups. In these cases, multiple entries are provided in the table. Note that when power consumption figures were not experimentally reported but instead obtained from hardware specification (by searching for the make and model), those values are marked with an asterisk (∗). Whenever certain data fields were omitted by the authors, we use "Nm" (not mentioned).

Paper Ref.	**Simulation Environment**	**CPU Cores**	**Memory (GB)**	**Power P idle (W)**	**Power P max (W)**
Availability-Aware Scheduler [1] (2018)	CloudSim	Nm	Nm	Nm	Nm
Renewable-Aware Scheduling [12] (2019)	Nm	4	64	100	150
Renewable-Aware Scheduling [12] (2019)	Nm	8	128	120	200
Renewable-Aware Scheduling [12] (2019)	Nm	16	256	150	250
Concurrent Container Scheduling [10] (2020) Homogeneous Cluster	ExoGENI	2	6	Nm	Nm
Concurrent Container Scheduling [10] (2020)	ExoGENI	1	3	Nm	Nm
Concurrent Container Scheduling [10] (2020)	ExoGENI	2	6	Nm	Nm
Concurrent Container Scheduling [10] (2020)	ExoGENI	4	12	Nm	Nm

Paper Ref.	**Simulation Environment**	**CPU Cores**	**Memory (GB)**	**Power P idle (W)**	**Power P max (W)**
GP Hyper-Heuristic for Containers [17] (2019)	Nm	Nm	0.8	Nm	Nm
PSO-Based Container Consolidation [16] (2018)	Nm	4	16	Nm	Nm
Energy-Efficient Container Framework [15]	CloudSim	4	64	86	117
Energy-Efficient Container Framework [15]	CloudSim	8	128	93	135
Energy-Efficient Container Framework [15]	CloudSim	16	256	66	247
VM Consolidation Strategy [6] (2017)	Nm	2	Nm	170	220
Predictive Resource Provisioning [7] (2015)	Nm	Nm	Nm	107	300.81
Dynamic VM Reallocation [4] (2010)	Nm	1	8	Nm	Nm
Autonomic Container Management [3] (2017)	LEGIS	Nm	16	Nm	Nm
Brownout-Based Scheduling [18]	CloudSim	2	4	156∗	247∗
GPR + Convex VM Planning [5] (2017)	Nm	4	12	225∗	375∗
SLA-Aware Consolidation [13] (2018)	CloudSim	2	4	102	117
SLA-Aware Consolidation [13] (2018)	CloudSim	2	4	116	135

Table 5: Hardware and Power Specifications

4.2 Plots and Visualization

We visualized the simulation environments used in each paper over time to observe how they have evolved (Figure 5). Each point corresponds to a paper, placed according to its publication year and the simulation platform it utilized (CloudSim, ExoGENI, LEGIS, or "Nm" if not mentioned) Each point denotes a paper, positioned by its publication year and the simulation platform it utilized.

Additionally, we plotted the relationship between the number of CPU cores and the *energy delta* (the difference $P_{\max} - P_{idle}$), shown in Figure 6. This second figure highlights how dynamic power consumption scales with increased core counts and is further distinguished by algorithm type (color) and simulation environment (marker shape).

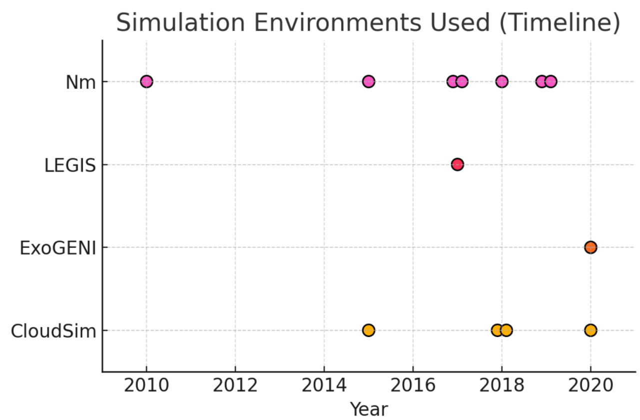

Fig. 5. Timeline of simulation environments used in each paper (2010–2020).

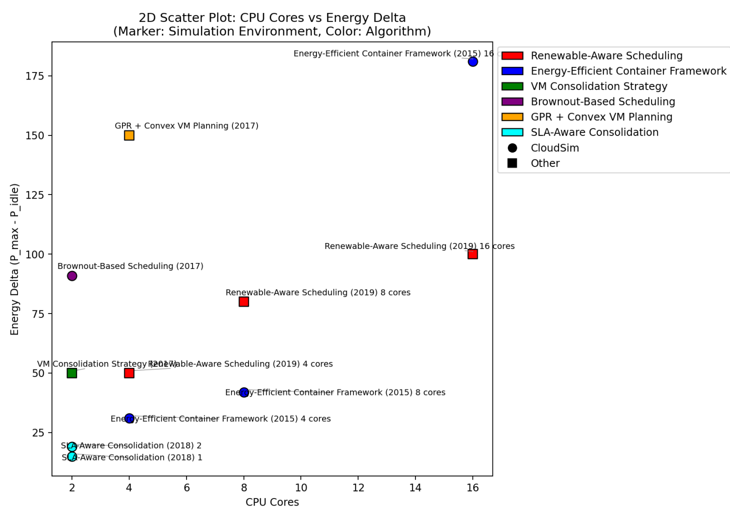

Fig. 6. 2D Scatter Plot: CPU cores vs. Energy Delta ($P_{\max} - P_{idle}$).

5 Conclusions

Energy consumption optimization in container-based cloud environments has evolved from early heuristic methods to sophisticated, predictive, and container-aware scheduling approaches. Early consolidation techniques demonstrated significant energy savings through aggressive host shutdown and DVFS, but their reliance on static thresholds often limited adaptability under dynamic workloads [6]. In contrast, predictive algorithms that incorporate workload forecasting and dynamic resource allocation have achieved further energy reductions (typically 30–35%) while balancing the trade-off between energy savings and service quality [5, 7].

The transition from VM-based to container-based solutions has enabled finer-grained resource management, with approaches such as flow-network optimization for container placement proving effective even in large-scale simulations [10]. Additionally, strategies like brownout scheduling [18] and meta-heuristic or AI-driven methods [16,17] have shown promise in achieving global optimization over complex infrastructures, albeit with increased computational overhead.

A key insight from the surveyed literature is that no single method outperforms all others under every scenario; instead, each approach excels in specific operational contexts. For instance, consolidation-focused strategies yield high energy savings but require careful management to avoid SLA violations, whereas QoS-aware methods maintain performance at the expense of some energy efficiency [13]. Moreover, the emerging integration of renewable energy awareness into scheduling algorithms [12] points toward a future where energy management is not only efficient but also environmentally sustainable.

Looking ahead, future research should aim to integrate these diverse techniques into hybrid frameworks that combine predictive scheduling, dynamic consolidation, and real-time QoS monitoring. This integration, alongside the continued evolution of container orchestration platforms, is essential for achieving the dual goals of operational efficiency and sustainable cloud computing.

References

1. Alahmad, Y., Daradkeh, T., Agarwal, A.: Availability-Aware Container Scheduler for Application Services in Cloud. https://doi.org/10.1109/PCCC.2018.8711295
2. Avgerinou, M., Bertoldi, P., Castellazzi, L.: Trends in data centre energy consumption under the european code of conduct for data centre energy efficiency **10**(10), 1470. https://doi.org/10.3390/en10101470, https://www.mdpi.com/1996-1073/10/10/1470
3. Barna, C., Khazaei, H., Fokaefs, M., Litoiu, M.: Delivering Elastic Containerized Cloud Applications to Enable DevOps. https://doi.org/10.1109/SEAMS.2017.12
4. Beloglazov, A., Buyya, R.: Energy efficient allocation of virtual machines in cloud data centers. In: 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing. pp. 577–578. IEEE. https://doi.org/10.1109/CCGRID.2010.45, http://ieeexplore.ieee.org/document/5493430/
5. Bui, D.M., Yoon, Y., Huh, E.N., Jun, S., Lee, S.: Energy efficiency for cloud computing system based on predictive optimization **102**, 103–114. https://doi.org/10.1016/j.jpdc.2016.11.011, https://linkinghub.elsevier.com/retrieve/pii/S0743731516301708
6. Carrega, A., Repetto, M.: Energy-aware consolidation scheme for data center cloud applications. In: 2017 29th International Teletraffic Congress (ITC 29). pp. 24–29. IEEE. https://doi.org/10.23919/ITC.2017.8065706, http://ieeexplore.ieee.org/document/8065706/
7. Dabbagh, M., Hamdaoui, B., Guizani, M., Rayes, A.: Energy-efficient resource allocation and provisioning framework for cloud data centers **12**(3),

377–391. https://doi.org/10.1109/TNSM.2015.2436408, https://ieeexplore.ieee.org/document/7111351/
8. Hameed, A., Khoshkbarforoushha, A., Ranjan, R., Jayaraman, P.P., Kolodziej, J., Balaji, P., Zeadally, S., Malluhi, Q.M., Tziritas, N., Vishnu, A., Khan, S.U., Zomaya, A.: A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems **98**(7), 751–774. https://doi.org/10.1007/s00607-014-0407-8, http://link.springer.com/10.1007/s00607-014-0407-8
9. Hintemann, R., Hinterholzer, S.: Cloud computing drives the growth of the data center industry and its energy consumption. Data centers (2022)
10. Hu, Y., Zhou, H., de Laat, C., Zhao, Z.: Concurrent container scheduling on heterogeneous clusters with multi-resource constraints **102**, 562–573. https://doi.org/10.1016/j.future.2019.08.025, https://www.sciencedirect.com/science/article/pii/S0167739X19309446
11. IEA: Data centres & networks. https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks, accessed: 2025-03-25
12. Kumar, N., Aujla, G.S., Garg, S., Kaur, K., Ranjan, R., Garg, S.K.: Renewable energy-based multi-indexed job classification and container management scheme **15**(5), 2947–2957. https://doi.org/10.1109/TII.2018.2800693, https://ieeexplore.ieee.org/document/8278252/
13. Li, L., Dong, J., Zuo, D., Liu, J.: SLA-aware and energy-efficient VM consolidation in cloud data centers using host states naive bayesian prediction model. pp. 80–87. IEEE Computer Society. https://doi.org/10.1109/BDCloud.2018.00025, https://www.computer.org/csdl/proceedings-article/ispa-iucc-bdcloud-socialcom-sustaincom/2018/114100a080/18AuNQj8lkA
14. Masanet, E., Shehabi, A., Lei, N., Smith, S., Koomey, J.: Recalibrating global data center energy-use estimates. Science **367**(6481), 984–986 (2020)
15. Piraghaj, S.F., Dastjerdi, A.V., Calheiros, R.N., Buyya, R.: A framework and algorithm for energy efficient container consolidation in cloud data centers. In: 2015 IEEE International Conference on Data Science and Data Intensive Systems. pp. 368–375. IEEE. https://doi.org/10.1109/DSDIS.2015.67, http://ieeexplore.ieee.org/document/7396528/
16. Shi, T., Ma, H., Chen, G.: Energy-Aware Container Consolidation Based on PSO in Cloud Data Centers. https://doi.org/10.1109/CEC.2018.8477708
17. Tan, B., Ma, H., Mei, Y.: A Hybrid Genetic Programming Hyper-Heuristic Approach for Online Two-level Resource Allocation in Container-based Clouds. https://doi.org/10.1109/CEC.2019.8790220
18. Xu, M., Dastjerdi, A.V., Buyya, R.: Energy efficient scheduling of cloud application components with brownout **1**(2), 40–53. https://doi.org/10.1109/TSUSC.2017.2661339, http://arxiv.org/abs/1608.02707