Energy-Aware Analytics in the Cloud

Carlos Ordonez
University of Houston
Houston, USA

Wojciech Macyna
Wroclaw University of Technology
Wroclaw, Poland

Ladjel Bellatreche
LIAS/ISAE-ENSMA
Poitiers, France

ABSTRACT

Big data is now mostly processed in the cloud and will keep growing, fed by databases and the Internet of Things (IoT: sensors, mobile devices, edge computing). On the other hand, AI is pushing computers and data analysis to limits we had not witnessed before. Analytics in the cloud is now a major fraction of energy consumption, among other less CPU-intensive tasks like web services. With this green computing motivation in mind, we present a survey of past research and a vision of big data analytics in the cloud. Energy consumption is difficult to minimize because it has conflicting correlated variables behind: high performance, money cost and pollution, We identify which software subsystems and hardware components have a higher impact on energy consumption, understanding how they can be tweaked or tuned to optimize energy consumption.

ACM Reference Format:
Carlos Ordonez, Wojciech Macyna, and Ladjel Bellatreche. 2024. Energy-Aware Analytics in the Cloud. In *International Workshop on Big Data in Emergent Distributed Environments (BiDEDE '24), June 9–15, 2024, Santiago, AA, Chile.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3663741.3664789

1 INTRODUCTION

Artificial Intelligence (AI) on Big Data is now the dominating trend to analyze the vast amounts of data generated by the digital ecosystem. Cloud computing has played a significant role in enabling rapid deployment and scalability of big data analytics, moving away from local (on-premise) servers. The Internet of Things (IoT) is a prominent data source of the digital economy, which allows collecting real-time data. The proliferation of smartphones, portable computers and edge computing devices has further accelerated the growth of the IoT. Data processing nowadays encompasses a wide gamut of applications including relational database management systems, NoSQL and data science (moving towards AI). Moreover, a significant portion of data processing is migrating to the Cloud. Schneider Electric estimates that the IT sector power demand will grow by 50 percent by 2030 [7], reaching 3,200TWh, equivalent to 5 percent Compound Annual Growth Rate (CAGR) over the next decade (Fig. 1). By 2040, projections indicate that the IT carbon footprint could reach 14%, with data centers contributing to almost half of this growth [7]. Given the significant energy requirements

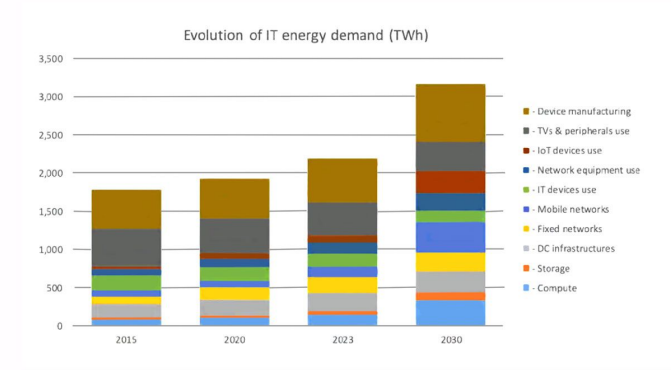

Figure 1: Breakdown of IT energy consumption.

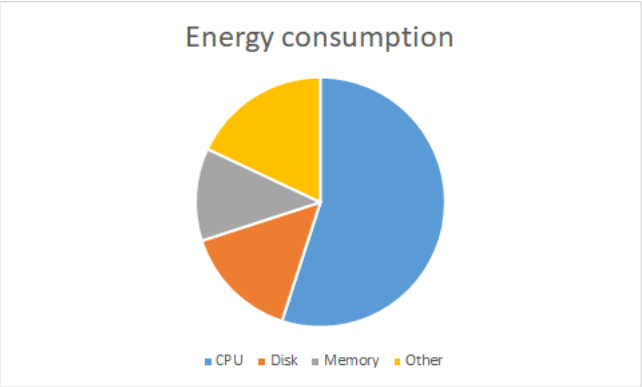

Figure 2: Energy consumption by hardware components.

of data centers that power cloud services, improving energy efficiency is crucial for sustainability and cost reduction. Based on energy consumed, AI is now contributing 1% to worldwide carbon emissions [14], which is less than pollution produced by factories or vehicles, but it will keep growing.

2 ENERGY OPTIMIZATION: HARDWARE

The statistics shown in Fig. 2 clearly indicate that the CPU consumes the largest portion of energy [13], followed by the storage device transitioning from HDDs to SSDs. The goal of this section is to highlight important energy reduction hardware techniques.

2.1 Dynamic Component Deactivation (DCD)

Techniques in this category aim at leveraging the workload variability by disabling relevant hardware components when they are idle. Setting up DCD techniques require prior workload knowledge,

BiDEDE '24, June 9–15, 2024, Santiago, AA, Chile

ACM ISBN 979-8-4007-0679-0/24/06…$15.00
https://doi.org/10.1145/3663741.3664789

to predict future workloads. This problem is more difficult in the cloud, where the hardware is shared by multiple Virtual Machines (to be discussed later).

2.1.1 Dynamic Performance Scaling (DynPS). DynPS can be seen as a focused response to DCD, More precisely, instead of complete deactivation, only the clock frequency of energy-hungry components, such as the CPU, can be decreased, with corresponding adjustments to the power supply, when resources are not fully utilized. Dynamic Voltage and Frequency Scaling (DVFS) is one of the most popular DynPS techniques, which allows a system to adaptively adjust the frequency and supply voltage to particular hardware components. This technique is widely used in ARM CPUs to increase battery duration, but it is not commonly used in the cloud, where powerful CISC (x86) CPUs need to deliver top performance, especially when AI models are computed. The application of the DVFS technique on a multi-core CPU is a complex task. It is often simplified by forcing each core on a package to operate at the same frequency and voltage. Having a system with only one global voltage for all cores (global DVFS) is energy-inefficient. To overcome this limitation, global DVFS and per-core DVFS architectures with multiple Voltage Frequency Islands (VFIs) have been proposed. In such platforms, the cores in an island share the same voltage and frequency, but different islands can be executed at various voltages and frequencies [15]. Advanced Configuration and Power Interface (ACPI), which is available on most operating systems, provides a standard interface for managing processor power states. To summarize, most of these proposed scheduling algorithms attempt to take advantage of the energy efficiency and time constraints of a real-time system in order to tailor the best reasonable compromise between electrical voltage and performance. The study conducted in [19] presents ongoing work in designing and implementing an energy-efficient DBMS (E^2DBMS) that enables significant energy conservation while maintaining a certain level of query processing performance. The tool achieves its goals via two strategies: (i) modifying CPU DVFS level through query optimizer based on a target specified by the DBA, and (ii) an energy-aware data placement technique by putting most I/O load into a subset of the physical disks, which allows disks entering low power/performance mode.

2.2 Accelerators

GPUs have evolved from their original purpose of fast processing of high resolution images and video to become essential components of modern AI infrastructure. Neural networks (deep, transformers), require many tensor (multidmensional arrays) computations, which are orders of magnitude more demanding than classical models (SVMs, decision trees, regression). GPUs provide extremely fast integer and floating point arithmetic for linear algebra computations, used to compute every ML model. But this comes at a price: GPUs consume more energy. In some cases a CPU-GPU architecture can provide better performance compared to executing all operations on a single device, especially in tasks like ETL (Extract, Transform, Load) processing and other I/O-intensive workloads. This can be achieved by harnessing the parallel processing capabilities of GPUs, offloading arithmetic operations to the GPU, allowing the CPU to focus on I/O aspects of the workload. The cloud offers a wide variety of server configurations with GPUs, in which the main consideration is cost, not energy. Currently, Large Language Models (LLMs) which involve huge tensors with billions of dimensions require servers with multiple GPUs.

Field Programmable Gate Arrays (FPGAs) are another accelerator choice. An FPGA is a semi-customized integrated circuit that can be programmed and configured for repetitive specific computations. CPU-FPGA architecture can be combined with resource provisioning and per-core CPU DVFS to further reduce energy usage [10]. However, FPGAs remain a less popular choice than GPUs in the cloud due to requiring advanced knowledge of computer architecture.

Tensor Processing Units (TPUs), invented by Google, are specialized hardware accelerators (ASICs) for neural network training workloads. TPUs are known for higher performance and energy efficiency compared to multicore CPUs. TPUs are designed for parallel processing of multidimensional arrays of real numbers. Since TPUs were designed specifically for fast matrix multiplications, they are generally faster than GPUs and consume less energy for the same workload, but they are more expensive. TPUs remain less popular than GPUs due to their higher cost, but they have the potential to surpass GPUs when competing AI libraries like PyTorch take full advantage of them.

2.3 Main Memory

Server energy consumption has been dominated by the CPU, followed my main memory (RAM). To reduce memory energy consumption, many techniques have been proposed. A certain number of techniques use the adaptive power saving or DVS offered by the modern multi-banked memory systems. Others reduce the power consumption by activating only some memory banks, leaving the other ones idle. Existing optimization techniques and algorithms focus on the opportunities to switch the entire memory or a part of it in low power mode, either during or at the time of the running process. In [12], the authors use rank aware memory allocation and rate-based data placement to deliberately skew memory access rates across available memory. This creates idleness on the least-loaded memory sections, thereby reducing overall memory power consumption. Some techniques exploit the CPU cache (e.g. L2 cache) to reduce energy consumed by RAM, by tuning core activity. It is important to highlight CPU caches are also power-hungry components in multicore CPUs.

In the case of neural networks, data movement up and down the memory hierarchy dominates energy consumption. The data movement can be reduced by controlling levels of local memory hierarchy considering different energy cost. Therefore, whenever data is transferred from a higher level in the memory hierarchy (e.g. registers) to a lower one (like L1 cache memory), it should be maximally reused to minimize the need for further access to the higher levels. Advanced memory technology can reduce the access energy for high density memories such as DRAMs. For instance, embedded DRAM (eDRAM) brings high density memory on-chip to avoid the high energy cost of switching off-chip capacitance. Moreover, eDRAM is 2.85x higher density than SRAM and 321x more energy efficient than DRAM [18].

Non-Volatile Memory (NVM) is the latest memory technology, bridging RAM and SSD [11], practically eliminating secondary

storage. Its main drawbacks are high energy consumption, shorter lifetime for write cycles and asymmetric I/O cost between read and write. When idle NVM draws minimal power to maintain data available, but it is not zero.

2.4 Secondary Storage

The storage system is an essential part of any system processing bug data. This is because of the growth in the size of data and the need to process and archive. The most energy-efficient storage hardware available today is the Solid-State Drives (SSDs). They use flash memory – a non-volatile memory having similar characteristics to electrically erasable programmable read-only memory (EEPROM). SSDs are much more power-efficient than hard disk drives due to their lack of moving parts. HDDs typically consume 6-15 Watts, whereas SSDs require 2-5 Watts (1/3). Servers equipped with HDDs have an average disk power consumption of 7%, while those with SSDs consume a mere 1%. SSDs dissipate less heat and, as a consequence, require less power for cooling. SSDs also require less power because most of the time they are in an idle state, whereas HDDs must continuously spin their disks for fast data access. In short, due to faster access and lower energy consumption SSDs are increasingly favored over HDDs in cloud data centers. It is anticipated that HDDs will eventually disappear as SSDs become cheaper.

3 ENERGY OPTIMIZATION: SOFTWARE

Energy efficiency in IT infrastructure is incomplete without energy-aware software. Software plays an important role in energy efficiency because it complements hardware. Here we provide a comprehensive survey, with an emphasis on database systems.

3.1 Virtual Machines and Containers

A common reason that motivates virtualization and containers is the low utilization of server components. A low utilization level (i.e., frequently idle) is inefficient due to wasting resources: infrastructure, maintenance, hardware, and power. Thus, a solution to optimize resource utilization is server consolidation by using virtualization, which enables running multiple independent virtual operating systems on a single physical computer. Virtualization is one of the most efficient methods for achieving energy efficiency in Cloud environments because a powerful physical server can support multiple virtual machines (VMs) with ample resources (CPU cores, RAM, storage) that will require minimal additional power but will use the same physical hardware, thereby reducing operating costs and power consumption aas well as simplifying Data Center Management. Containerization (e.g. Docker) is an alternative technique that enables application programs to run and be deployed on isolated virtual space, but the operating system kernel is shared among them. Sharing the operating system kernel in a container-based architecture is one of the key reasons why containers are more lightweight and faster to start on demand compared to VMs. Another benefit is that containers can be more easily orchestrated and scaled up or down to meet dynamic demands of modern applications and microservices architectures, making them a popular choice for cloud-native development and deployment. In the server consolidation method, several virtual machines and containers are packed in the minimum number of physical machines in order to turn off or switch the status of the idle hosts to sleep mode to minimize energy consumption. Container consolidation is more energy-efficient than VM consolidation [8]. Since the cloud is a shared resource it is becoming practice to set up a schedule where the cloud instance becomes available, giving the possibility to turn off idle machines.

Virtual machine migration, load balancing, and workload categorization are problem-solving techniques employed to reduce power consumption in the data centers. These methods involve migrating virtual machines when specific server thresholds are reached, distributing the workload evenly among various VMs, and categorizing workloads before assigning them to servers. To further enhance power management in data centers, machine learning algorithms are often applied on top of these approaches. Dynamic power management must be accomplished at the data center level. The objective is to allocate the minimum necessary physical resources to virtual machines while deactivating or putting unused resources into a sleep or hibernation state [13]. Live migration [6] moves a VM from one machine to another machine, scaling down or scaling up resources according to demand. Live migration considers energy, among other factors, load balancing and resource allocation.

In summary, VMs and containers decrease performance (to acceptable levels), but with significant energy and cost savings.

3.2 Energy-efficient Algorithms

Quantization: Reducing the precision of numerical representations (e.g., using 8-bit integers instead of 32-bit floating-point numbers) can significantly decrease computational demands and energy consumption while maintaining acceptable model performance.

Accelerating ML Algorithms: Developing machine learning algorithms that require fewer iterations or computations to reach a solution can achieve energy savings. For example, stochastic gradient descent (SGD) variants can be more efficient than older iterative methods or batch gradient descent (the default). Model pruning involves removing unnecessary parameters (or neuron/vertex connections). This optimization reduces computational complexity and energy consumption, while generally producing a minor impact on accuracy. During model training, which is the most CPU-intensive computation, dynamically adjusting the learning rate, batch size, and other hyperparameters based on energy consumption and system load to optimize efficiency. In contrast, inference (deploying a computed model on new data) has much lower energy cost.

3.3 Operating System

The Linux kernel plays a significant role in the new wave of embedded and mobile devices, in addition to cloud servers [2]. It leverages various power management features including hardware tuning tools like hdparm, swsusp, clock gating, voltage scaling, sleep mode activation, and memory cache deactivation. However, ongoing research aims to enhance the platform's functionality further. In [17], the authors explore the behavior of the task management subsystems (scheduler and load balancer) in the Linux kernel on multi-core Symmetric Multi-Processing (SMP) systems. It assesses their effectiveness at reducing energy consumption across different scenarios, such as idle and moderate load, and discusses techniques like timer

migration, task wakeup biasing, and related heuristics for energy reduction. Original power management from Linux is reproduced to Android. However, these solutions do not satisfy mobile devices or embedded systems. They must consider constraints like limited battery power capacity for instance.

3.4 Extended Cost Models: I/O and Energy

Here we describe energy savings in data systems, with a focus on database systems, following the system architecture introduced in [5]. Energy saving can be considered at both transactional and query levels. Database transactions, in particular, have a substantial impact on overall energy consumption. The energy usage correlates with the size of the data involved in the transaction. Larger data sizes require more resources, such as CPU, memory, and storage, leading to increased energy consumption. To mitigate this, techniques like batch inserts can significantly improve energy efficiency. Similar challenges arise during data exporting and loading processes, where energy-saving strategies can also be applied to reduce resource utilization and energy consumption. At the query level, most approaches are based on cost models for query processing [4]. In this way, a reasonable trade-off between performance and energy consumption can be estimated.

Following classical I/O cost models, cost models have been used in data processing systems for estimating energy consumed in query processing. In general, the energy consumption of query Q is the sum of energy used by hardware components: CPU, RAM, IT, and network (equation 1).

$$E^Q = E^Q_{CPU} + E^Q_{RAM} + E^Q_{IO} + E^Q_{NET}. \quad (1)$$

Regardless of system, Equation 1 is adapted to take into account data block sizes involved in energy consumption. The following linear combination is a typical example of cost model estimation in one local server.

$$E = E_{cpu} * N_{tuples} + E_{I/O} * N_{pages}, \quad (2)$$

where E_{cpu} and $E_{I/O}$ are the energy consumption coefficients of a record processed by the CPU (N_{tuples}), and the I/O cost coefficient of an I/O processed page (N_{pages}). In a cloud data center these equations are generalized to M machines (e.g. a cluster of uniform CPUs), disaggregated storage (e.g. Amazon S3) [20], high speed interconnection (e.g. InfiniBand) and networking hardware (e.g. Ethernet):

$$E_c = M * E_{CPU} + E_{storage} + E_{inteconnect} + E_{network}. \quad (3)$$

Notice there are subtle changes with respect to a local server: each machine does not have its own I/O cost since the cloud does not use a shared-nothing architecture and we are adding separate energy costs for interconnection (higher) and networking cards (lower). Another observation, is that we are bundling accelerator costs into CPU cost.

In the same spirit, other energy costs have been proposed. More concretely, the energy modeling of a query can be done at three levels: **(i)** query level, which considers the characteristics of the query itself, such as the number of I/O required to execute the query, **(ii)** pipeline level, which considers the characteristics of the

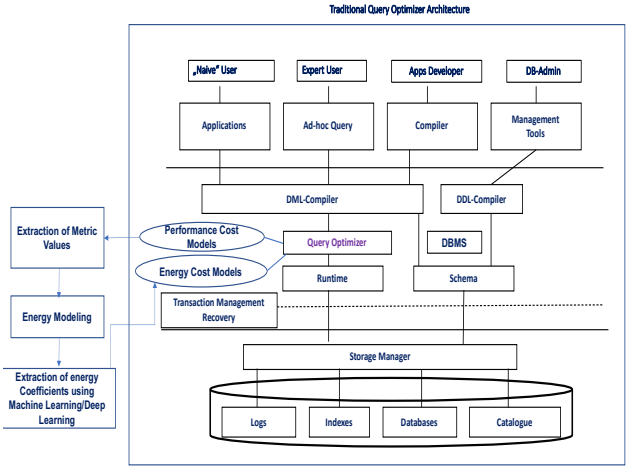

Figure 3: System architecture to develop energy cost models.

set of operators that run simultaneously and **(iii)** operator level, which considers the characteristics of the SQL operator individually. The energy consumption for a single query is not necessarily the same as that of a set of queries running at the same time in batch mode. Therefore, there are two query execution modes to consider in the design phase of an energy cost model: *isolated* and *concurrent* modes. When a database is deployed in distributed infrastructures, where its fragments are allocated in various nodes, the network plays a crucial role in increasing energy. This is due to data transfer when executing binary database operations like joins, where the data is not usually localized in the same node. In this case, the cost models can be easily enriched by an energy coefficient corresponding to the network components. By analyzing the different efforts in developing cost models related to energy savings, we realize that they are all defined on top of existing query processing approaches. As shown in Fig. 3, existing energy cost models reuse the models already present in the target big data system to build their models inspired by Equation 1.

After constructing the energy-efficient cost model, the next step is to identify the coefficients associated with each cost used by the target storage system (see equation 2). This identification is usually performed using AI-driven approaches. Most proposed cost models use traditional machine learning methods to extract the features of their cost models, with linear regression being the most popular. Other advanced AI techniques that describe energy behavior during query processing have to be explored to set the relevant values, and to dynamically calibrate parameters when the workload changes [1]. The diverse (non-uniform) nature of CPU nodes in the cloud makes ML models complex (to estimate the equation coefficients above).

Validating cost model accuracy is crucial to measure its effectiveness. The disparity between estimations given by cost models and real energy measurements can be calculated using AC power measurement tools (e.g. WattsUp Pro ES), which give am exact reference value.

Developing accurate energy cost models requires understanding the interactions between various system components, including

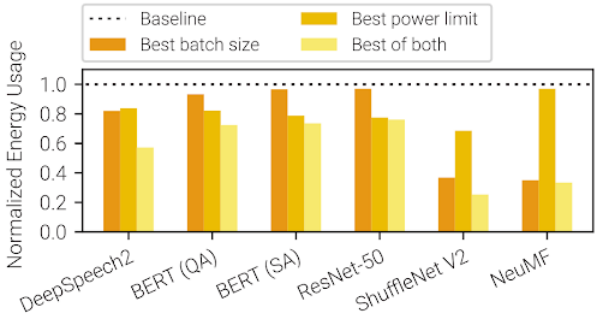

Figure 4: Energy improvement obtained by AI models.

hardware and software. Estimating I/O, CPU, RAM, and cache costs should be part of the cost model development rather than treating them as black boxes enriched with energy coefficients. Researchers need to consider selectivity factors of join predicates, intermediate result sizes, and algorithms used for basic data operations (e.g., hash join, sort-merge) when building energetic cost models. Extended cost models for predicting energy are used by various data systems to optimize database energy consumption [16]. The system introduced in [9] integrated an energy cost model into the query processing module of a DBMS. Rather than choosing plans with optimal performance, plans with acceptable performance degradation within a certain threshold are selected to save energy. Such approaches represent a proof that a trade-off between query performance and energy consumption is feasible.

3.5 AI Models to Predict Energy Consumption

Energy efficiency in machine learning refers to the practice of designing and implementing machine learning algorithms, models, and infrastructure with a focus on minimizing energy consumption, while maintaining computational quality of service (QoS). Developed at the University of Michigan, the open-source optimization framework *Zeus* studies deep learning models during training, pinpointing the best tradeoff between energy consumption and the speed of the training. Fig. 4 shows a variety of common deep learning models benefit from Zeus' ability to tune GPU power limits and the training batch size. When both parameters were tuned, the software achieved up to 75% energy reduction without changing any hardware components.

3.5.1 Model Architecture. Designing energy-efficient machine learning models involves choosing model architectures that strike a balance between accuracy and computational requirements. Smaller and more efficient models can achieve similar performance than larger ones, while consuming fewer resources.

4 OPTIMIZING ENERGY: ENVIRONMENT

The major contributor to the total energy usage in data centers is IT equipment, which consists of rack servers, storage devices, networking equipment and AC cooling systems. It is noteworthy AC cooling contributes one third of energy consumption. Nowadays, energy efficiency cooling techniques for data centers have become a major and attractive challenge. There are two main directions on power savings of cooling systems, one is to reduce the cooling production directly, the other is to reduce power consumption while at the same time maintaining a given cooling production profile. To optimize the cooling system the data-driven optimization approach can be used. It is based on the data and train models which take various system extrinsic and intrinsic factors into consideration, hence is highly adaptive to many circumstances like aging devices, deteriorating equipment conditions, and so on [21]. Cooling efficiency is also influenced by the type of computer system. Rack servers consume less energy to cool down due to their stacked configuration and efficient ventilation systems. Laptops and notebooks are designed to consume low power and dissipate heat, with small fans. But they may be less efficient than rack servers in cooling efficiency. ARM machines will probably dominae the market. Desktops, on the other hand, are the least efficient to cool because they are individually positioned, and air circulation is less effective.

5 RESEARCH ISSUES

5.1 Hardware

5.1.1 Hardware accelerators. Hardware accelerators such as GPUs, TPUs, FPGAs are extensively exploited for computationally intensive tasks, such as neural networks (AI) and numerical methods (HPC). But their energy consumption is high, especially with large GPUs. However, initial stages of a project or smaller problems can be solved with (more energy-efficient) multi-core CPUs. Thus we need new architectures, extended energy cost models. The growing trend towards using deep learning and GPUs in data processing systems poses new challenges for energy efficiency. GPUs are known to be energy-intensive, the number of data replicas and their size to maintain fault tolerance would increase CPU usage during data loading. Dynamic workload assignment to machines, as opposed to predefined configurations offered by cloud providers, can also have an impact on energy consumption. FPGAs accelerate specific computations, but could also save energy when repetitive operations can be offloaded from the CPU.

5.1.2 Dynamic voltage and frequency scaling. Dynamic Voltage and Frequency Scaling (DVFS) are well-known power management techniques in CISC/RISC CPUs. The energy impact of these hardware features in virtual machines and containers is not well understood. Moreover, non-uniform hardware (diverse CPUs, far memory, mixing SSD and HDD) brings up new challenges.

5.1.3 Storage Devices. New storage technologies such as SSDs have the potential to significantly decrease the energy consumption associated with processing big data. SSDs are much faster than HDDs and they consume much less energy. Nevertheless, SSDs have higher cost and shorter write life. It is necessary to extend and tune old I/O models, to save energy. Speed and energy tradeoffs between RAM and SSD need to be studied (SSD access speed is approaching RAM access speed).

5.1.4 Using machine learning models for tuning file access parameters. Machine learning predictive models have been used to optimize resource utlization in distributed processing in the cloud, in query processing and in AI itself. There are significant advances in learning cost model parameters for query processing via AI.

Such models should be extended to also reduce energy by tuning parameters, but providing acceptable performance.

5.1.5 Energy-aware edge computing. Edge devices feed the cloud. Energy-aware edge computing has been explored extensively, given the fact that devices tend to operate on batteries or small power supplies. But existing research often focuses on singular objectives like low latency, data privacy, or power saving. Opportunities exist for optimizing multiple objectives, such as energy efficiency and low latency simultaneously. Novel architectures and middleware have been proposed for interoperability [3], yet operating system level energy awareness in edge computing remains a challenge. Additionally, there is limited research on compiler-level optimization and managing heterogeneous hardware efficiently, indicating open areas for systematic energy reduction.

5.2 Software

5.2.1 Virtual Machines and Containers . Virtualization enables the efficient sharing of hardware among multiple virtual machines (VMs). Considering the trade-offs between adding more main memory and more virtual CPUs is essential for optimizing hardware usage, while achieving higher energy efficiency. By consolidating workloads onto fewer physical servers, virtualization reduces the overall energy consumption of data centers. However, optimizing workloads for energy instead of time performance, understanding energy consumption by cloud instance type and overall usage of virtual CPUs require further study.

Container technology (e.g. Docker) is on the rise and warrants further research into energy efficiency. Well-constructed containers enable hosts to maximize resource utilization, and isolated containers operate independently, enabling a single host to perform multiple functions. Future work should focus on developing efficient techniques for container/task placement on physical machines, taking into account CPU multicores, memory, storage, and network resources collectively.

5.2.2 Energy-efficient Analytic Algorithms. Numerical computations, and neural networks in particular, consume tons of energy. Neural networks have motivated using lower precision floating point arithmetic, smaller integers and quantization (binary coding) techniques. More numeric techniques to improve energy-efficiency with minor accuracy sacrifice are needed.

5.2.3 Extended Cost Models. We need more hybrid models combining energy and I/O cost. Further research is needed on non-linear models. Cost models should consider environment-level parameters such as hardware age, physical density (equipment per cubic feet) and room temperature hosting the data center hardware.

5.3 Socio-Political-Legal Aspects

Revising Economic Models in the Cloud: Current economic models used by data science in the cloud (SaaS) should be reviewed to include the energy dimension and adopt the "polluter pays" principle. This approach will encourage cloud providers and large corporations to be more environmentally responsible. Evidently, there are socio-political aspects beyond Computer Science.

Service Level Agreements (SLAs) in the cloud: As cloud computing continues to evolve, energy consumption will likely become another important requirement in Service Level Agreements (SLAs). In the past, was to have guaranteed performance under SLAs, but now energy efficiency will also be a critical factor.

REFERENCES

[1] Dana Van Aken, Andrew Pavlo, Geoffrey J. Gordon, and Bohan Zhang. 2017. Automatic Database Management System Tuning Through Large-scale Machine Learning. In *ACM SIGMOD*. 1009–1024.

[2] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. 2010. A view of cloud computing. *Commun. ACM* 53, 4 (2010), 50–58.

[3] V. Balasubramanian, Nikolaos Kouvelas, Kishor Chandra, R. Venkatesha Prasad, Artemios G. Voyiatzis, and William Liu. 2018. A unified architecture for integrating energy harvesting IoT devices with the Mobile Edge Cloud. In *4th IEEE World Forum on Internet of Things (WF-IoT)*. IEEE, 13–18. https://doi.org/10.1109/WF-IOT.2018.8355198

[4] Ladjel Bellatreche, Fouad Djellali, Wojciech Macyna, and Carlos Ordonez. 2023. Energy-Aware Query Processing: A Case Study on Join Reordering. In *IEEE International Conference on Big Data, BigData*. 3743–3752. https://doi.org/10.1109/BIGDATA59044.2023.10386332

[5] Simon Pierre Dembele, Ladjel Bellatreche, Carlos Ordonez, and Amine Roukh. 2020. Think big, start small: a good initiative to design green query optimizers. *Clust. Comput.* 23, 3 (2020), 2323–2345.

[6] Mohamed Esam Elsaid, Hazem M. Abbas, and Christoph Meinel. 2022. Virtual machines pre-copy live migration cost modeling and prediction: a survey. *Distributed Parallel Databases* 40, 2-3 (2022), 441–474.

[7] Charlotte Freitag, Mike Berners-Lee, Kelly Widdicks, Bran Knowles, Gordon Blair, and Adrian Friday. 2021. The climate impact of ICT: A review of estimates, trends and regulations. arXiv:2102.02622 [physics.soc-ph]

[8] Niloofar Gholipour, Ehsan Arianyan, and Rajkumar Buyya. 2020. A novel energy-aware resource management technique using joint VM and container consolidation approach for green computing in cloud data centers. *Simul. Model. Pract. Theory* 104 (2020), 102127. https://doi.org/10.1016/j.simpat.2020.102127

[9] Binglei Guo, Jiong Yu, Bin Liao, Dexian Yang, and Liang Lu. 2017. A green framework for DBMS based on energy-aware query optimization and energy-efficient query processing. *Journal of Network and Computer Applications* 84 (2017), 118–130.

[10] Michael Guilherme Jordan, Guilherme Korol, Tiago Knorst, Mateus Beck Rutzig, and Antonio Carlos Schneider Beck. 2023. Energy-aware fully-adaptive resource provisioning in collaborative CPU-FPGA cloud environments. *J. Parallel Distributed Comput.* 176 (2023), 55–69. https://doi.org/10.1016/j.jpdc.2023.02.009

[11] Saeed Kargar and Faisal Nawab. 2023. Challenges and future directions for energy, latency, and lifetime improvements in NVMs. *Distributed Parallel Databases* 41, 3 (2023), 163–189.

[12] Alexey Karyakin and Kenneth Salem. 2019. DimmStore: Memory Power Optimization for Database Systems. *Proc. VLDB Endow.* 12, 11 (2019), 1499–1512. https://doi.org/10.14778/3342263.33422629

[13] Avita Katal, Susheela Dahiya, and Tanupriya Choudhury. 2023. Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Computing* 26, 3 (2023), 1845–1875. https://doi.org/10.1007/s10586-022-03713-0

[14] Keith Kirkpatrick. 2023. The Carbon Footprint of Artificial Intelligence. *Commun. ACM* 66, 8 (2023), 17–19. https://doi.org/10.1145/3603746

[15] S. Pagani, A. Pathania, M. Shafique, J. Chen, and J. Henkel. 2017. Energy Efficiency for Clustered Heterogeneous Multicores. *IEEE Transactions on Parallel and Distributed Systems* 28, 5 (2017), 1315–1330.

[16] Amine Roukh, Ladjel Bellatreche, and Carlos Ordonez. 2016. EnerQuery: Energy-Aware Query Processing. In *ACM CIKM*. 2465–2468.

[17] Vaidyanathan Srinivasan, Gautham R Shenoy, Srivatsa Vaddagiri, and Dipankar Sarma. 2009. Energy-aware task and interrupt management in Linux. In *Ottawa Linux Symposium*.

[18] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. 2017. Efficient Processing of Deep Neural Networks: A Tutorial and Survey. *Proc. IEEE* 105, 12 (2017), 2295–2329. https://doi.org/10.1109/JPROC.2017.2761740

[19] Yi-Cheng Tu, Xiaorui Wang, Bo Zeng, and Zichen Xu. 2014. A system for energy-efficient data management. *ACM SIGMOD Record* 43, 1 (2014), 21–26.

[20] Jianguo Wang and Qizhen Zhang. 2023. Disaggregated Database Systems. In *Companion of SIGMOD Conference*. ACM, 37–44. https://doi.org/10.1145/3555041.3589403

[21] Yong Yu. 2020. AI Chiller: An Open IoT Cloud Based Machine Learning Framework for the Energy Saving of Building HVAC System via Big Data Analytics on the Fusion of BMS and Environmental Data. *CoRR* abs/2011.01047 (2020). arXiv:2011.01047 https://arxiv.org/abs/2011.01047