Optimizing Energy Consumed by Analytics in the Cloud

Carlos Ordonez
University of Houston
Houston, USA

Wojciech Macyna
Wroclaw University of Technology
Wroclaw, Poland

***Abstract*—Given the climate change crisis, there is a worldwide growing concern on energy production, energy consumption and pollution. Cloud computing represents a small fraction of global energy consumption, but trends indicate it will continue to grow, driven by Big Data and AI. AI analytics are pushing computing resources, especially CPUs and GPUs, to their limits. However, powerful CPUs and GPUs, consume tons of energy and require cooling appliances, which results into higher operating cost and as an indirect consequence, higher pollution and global warming. Based on these issues, we present a survey on measuring and reducing energy, especially when processing analytic workloads. We discuss tradeoffs between high performance (low latency to get results) and low energy (less power consumed over time). Our focus is on identifying modern hardware components which have a significant impact on energy consumption and then examining how software optimizations can manage hardware to reduce energy in a cloud data center. We conclude with a tentative research agenda, based on the state of the art of research at the intersection of big data analytics, high performance computing, electrical energy and cloud computing.**

I. Introduction

Big Data and Artificial Intelligence (AI) are major tasks in cloud computing. The Internet of Things (IoT) keeps Big Data growing. Edge computing is offloading some Big Data processing from the cloud, but most AI processing happens in the cloud. All these computing tasks need tons of energy. By 2040, projections indicate that the IT carbon footprint could reach 14%, with cloud data centers contributing 50% [10]. Given the significant energy requirements of cloud data centers, reducing energy demand is critical for sustainability. Based on electrical energy consumed, AI is now contributing a surprising 1% to worldwide carbon emissions [17], which is much less than the pollution from factories and vehicles, but it will keep growing as AI becomes more pervasive.

We argue energy is a hardware aspect, which needs to be measurable, tunable and tweakable by software. We believe developers and researchers need to become aware about how much energy is consumed in analytic workloads, especially AI with tons of linear algebra and numerical methods. We suggest potential solutions, considering data science projects developed and deployed in a cloud environment. We conclude the paper with a tentative, but ambitious, research agenda.

II. Background

A. Cloud Architecture

Cloud computing [3] is defined as a technology that offers software as a service (SaaS), guaranteeing high quality of service (QoS), under dynamic economic supply/demand changes (elasticity). The customer (user, company, organization) pays according to needed computing power, storage size, data retrieval speed, time availability and shared/exclusive access, under a Service-Level Agreement (SLA) guaranteed by the cloud provider. In general, energy is not a consideration for the user who looks at the cloud as an imaginary computer with ample (almost unlimited) resources.

From a hardware perspective, the cloud consists of many rack servers featuring multicore CPUs, large main memory (RAM), with/without GPU, interconnected with each other via a high speed hardware interconnection (e.g. Infiniband) or a fast LAN wire connection (Gigabit Ethernet). In general, rack servers do not have internal massive storage. Instead, they have access to Network Attached Storage (NAS), commonly known as disaggregated storage. In Section III, we explore these components at a fine granular level and at a macro cloud level, understanding performance and energy tradeoffs.

From a software perspective, the operating system, virtualization, containerization, microservices, scheduling, and orchestration manage all hardware components, giving the user the impression of working with a local physical server.

B. Power and Energy

Our discussion focuses on electrical energy. We start by defining electrical power as

$$P = V \cdot I, \tag{1}$$

where V (voltage) is measured in Volts (V), I (current intensity) is measured in Amperes (A) and power P is measured in Watts (W). Therefore, Watts are intuitively understood as Watts=Volts $\times$ Amperes. Example: a computer in the US working at 120V, with a 3A power supply can draw up to 360 Watts at full capacity.

Based on the previous succinct definitions, energy is defined as power consumed over a period of time, going from seconds to hours. In Physics energy is defined, in a general manner, as power consumed over time: $P \cdot t$, where t is time elapsed in seconds. In practice, for billing and provisioning purposes,

Watts are measured over hours (not seconds), resulting in Watts/hour (Wh) to measure electrical energy consumption. In this article, we use the energy equation below, where electrical energy E is measured in Wh:

$$E = P \cdot t \tag{2}$$

When analyzing energy trends it is common to measure energy in kilowatts/hour (kWh) and megawatts/hour (MWh). Therefore, we will use the three measurement units (Wh, kWh, MWh), depending on the size and scope of the underlying components: Wh for specific hardware components, kWh for many servers, interconnected devices and AC cooling units and finally MWh for cloud data centers.

Hardware-based energy measurement techniques directly monitor physical components like CPUs, GPUs, memory, and storage in data centers or on-premises setups. Key methods include power meters, which provide real-time consumption data at the rack, server, or processor level, and intelligent power distribution units (PDUs), which offer detailed device-level usage data by measuring voltage, current, and power factor. Some modern CPUs and GPUs also have on-chip sensors, reporting energy and temperature data via interfaces like Intel's RAPL or NVIDIA's NVML. Additionally, external monitoring tools, such as smart meters and energy analyzers, deliver precise measurements for individual devices or entire cloud regions, making them valuable in hybrid and experimental environments.

C. Analytic Tasks Classification

In the past, analytics were mostly exploratory (cube queries, business intelligence), statistical (histograms, plots, statistical tests, simple predictive models) and data mining (machine learning pattern discovery on large data sets). Today, the main analytic task is training and deploying neural networks, which are commonly known by the popular keyword "AI". However, database technology is still used to feed clean, integrated, data to neural networks, statistics are still needed to understand probabilistic distribution, linear vs non-linear behavior and to compute preliminary predictive models, and data mining techniques are being revisited and extended to make neural networks scalable. However, the input data files tend to be much larger for neural networks (mixing text, images and databases) and the computation tends to involve many more arithmetic and floating point operations (linear algebra, numerical methods). In this survey, we aim to quantify what fraction and amount of energy is used by each analytic technique.

III. Optimizing Energy on Modern Hardware

The goal of this section is to highlight key hardware mechanisms for reducing energy consumption. We will discuss various concepts related to energy efficiency and identify the cloud hardware components where significant energy reduction can be achieved.

The variety of cloud infrastructures and workflow types makes it impossible to define a unified method for saving energy. Different setups and workloads require tailored approaches to optimize energy consumption based on specific needs and hardware resource configurations.

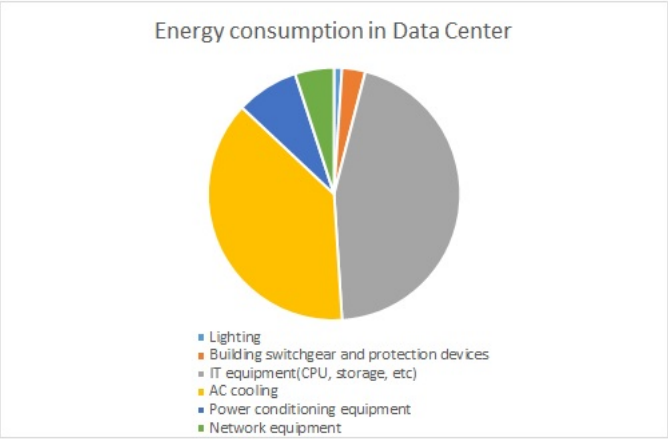

Fig. 1. Breakdown of energy consumption by hardware component.

A. Power and Cooling Components

The number of servers in a data center varies significantly based on its size and purpose. Small data centers may house hundreds of servers, while enterprise data centers typically host between 500 and 3,000 servers. Hyperscale data centers, such as those operated by Google, Amazon, or Microsoft, can accommodate anywhere from 100,000 to 400,000 servers and even more, depending on the facility's scale and operational needs. It is estimated that the total power consumption of Google's server data centers ranges from 500 to 700 megawatts.

The major contributor to the total energy usage in data centers is IT equipment, which consists of rack servers, storage devices, networking equipment, but also AC cooling systems. Cooling contributes a whopping one-third of this energy use [13], a fact that is generally overlooked or dismissed. Therefore, the system deployment environment (building, geographical location) is critical to the optimal functioning of these components. Nowadays, energy efficiency cooling techniques for the cloud have become a central problem. There are two main directions for power savings of cooling systems, one is to reduce the cooling air production directly down to some minimum threshold, and the other one is to reduce power consumption with alternative cooling mechanisms, while at the same time maintaining a given cooling production profile. Cloud providers like Google, Amazon, and Microsoft are adopting more efficient cooling methods to reduce energy usage. Free cooling uses natural air or water to cool data centers, reducing the need for mechanical air conditioning. On the other hand, liquid cooling, such as direct-to-chip systems, circulates coolant around components and can cut power consumption by up to 40%. Immersion cooling submerges servers in non-conductive liquid, offering even more energy savings

[33]. AI-optimized cooling dynamically adjusts cooling configurations in real-time. 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 [32].

B. CPU Energy-Performance Optimization: DVFS

Dynamic Voltage and Frequency Scaling (DVFS) can significantly reduce energy consumption in CPUs by dynamically adjusting their voltage and frequency based on workload requirements. Energy savings from DVFS in the cloud can vary depending on the workload and system architecture but typically range from 20% to 50% in general computing environments [27], [14], [20]. The savings can be optimized further by integrating DVFS with workload management and scheduling strategies [6]. However, the exact rates depend on system configurations, the type of tasks being performed, and the level of scaling applied [1].

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 [23]. By lowering the voltage and frequency of less critical or idle areas, energy consumption is significantly reduced without sacrificing the performance of active sections. VFI is particularly effective in complex systems like multi-core processors and cloud infrastructures, where workload variability is common.

C. Accelerators

Hardware accelerators play a key role in reducing processing time and improving energy efficiency across various computing environments by offloading specific tasks from general-purpose processors to specialized hardware optimized for those tasks. This enables faster computation with reduced energy consumption. Two popular types of hardware accelerators in the cloud are GPUs and FPGAs. GPUs are particularly effective when used in conjunction with CPUs.

1) GPU: GPUs today repesent the most important accelerator. 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 (multidimensional 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 [30], used to compute every ML model. But this comes at a price: GPUs consume more energy.

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. Different GPUs are designed for specific workloads, which can be broadly categorized into graphics, High-Performance Computing (HPC), and deep learning (DL). Graphics workloads perform best on GPUs optimized for texture cores and Graphics-DDR memory, like the NVIDIA A10 and RTX A5000. In contrast, HPC and DL workloads benefit from GPUs with tensor cores and High-Bandwidth Memory (HBM), such as the NVIDIA A100 and T4. Both types of GPUs are widely used in the clouds.

High-performance GPUs like NVIDIA GeForce or AMD Radeon models typically consume between 15 to 30 watts when idle. Under moderate loads, such as gaming or graphics rendering, these GPUs can draw between 150 to 300 watts. However, during tasks like deep learning, AI training, or scientific simulations, high-end GPUs such as the NVIDIA A100 or AMD Instinct MI100 can consume up to 400-500 watts or more. On the other hand, low-power CPUs, such as those designed for mobile or energy-efficient applications, typically consume between 10 to 35 watts under load. In contrast, standard desktop CPUs generally range from 50 to 125 watts, depending on the specific model and workload requirements. Meanwhile, server or high-performance CPUs, which are tailored for data centers or compute-intensive tasks, can consume between 100 to 300 watts or more, particularly during peak performance conditions. While GPUs can consume up to 400-500 watts under heavy workloads, their ability to complete tasks faster than CPUs often results in overall energy savings for high-performance tasks, especially in cloud infrastructures.

Compute-intensive tasks, such as AI model training or gaming, significantly increase power usage due to the parallel processing capabilities of GPUs, whereas idle or low-utilization tasks use less power but still more than CPUs. Clock speed and voltage adjustments, through mechanisms like DVFS, help manage energy consumption by reducing power when lower performance is sufficient [19]. Modern GPU architectures, such as NVIDIA's Ampere and AMD's RDNA are designed for greater energy efficiency, while specialized AI-focused GPUs further optimize power use for specific tasks. Effective thermal management is essential to prevent overheating, which can lead to higher energy consumption [22]. Finally, GPUs adjust automatically their power consumption based on workload intensity through different power states, with minimal energy use in idle states and higher consumption during demanding applications. Energy efficiency in GPUs is often measured by the performance per watt metric, indicating how much computational work can be done for each watt of power consumed. While newer GPU models generally increase both peak performance and power usage, architectural improvements and advanced power management techniques have made modern GPUs more energy-efficient on a per-task basis compared to older versions. This allows newer GPUs

to deliver higher performance while consuming proportionally less energy for equivalent tasks, optimizing both power use and computational output.

Measuring GPU energy consumption is crucial for optimizing performance and energy efficiency, particularly in data centers and high-performance computing. Methods include using external power meters or in-line monitors to measure total power usage, software tools like NVIDIA SMI and AMD Radeon Software for real-time monitoring, and GPU performance tools such as GPU-Z and MSI Afterburner. Many modern GPUs also feature integrated power management, while profiling tools like CUDA Profiler and TensorFlow Profiler can assess energy efficiency during specific tasks. Energy consumption can be calculated over time by recording power usage at intervals, and thermal sensors can provide insights into the impact of temperature on energy consumption. Combining these methods offers a comprehensive understanding of GPU power usage, facilitating optimization strategies.

2) CPU-GPU: 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.

Intel (CISC/x86) and ARM (RISC) architectures are both integrated with GPUs. Table I highlights the key differences between these architectures. In the cloud, ARM-based processors (like AWS Graviton) are gaining traction due to their energy efficiency and lower cost, and GPU acceleration is increasingly becoming a focus in ARM-based cloud computing environments for tasks such as AI inference and edge computing.

3) Field Programmable Gate Arrays (FPGAs): 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. FPGAs are especially valuable for tasks requiring real-time data processing and adaptable hardware configurations, enhancing flexibility and scalability in cloud infrastructure. In the last few years, FPGAs have been used in the cloud for diverse applications (see [5]). Common cloud FPGA use cases include: (a) Customer applications, where users can develop, simulate, and scale their custom FPGA logic for tasks like genomics, financial analytics, or video processing; (b) Application as a Service (AaaS), where the cloud provider develops FPGA designs and exposes only necessary APIs, offering high performance but limited customer control; and (c) Provider applications, where cloud providers use FPGAs to accelerate internal workloads freeing up CPU resources for customer use.

FPGAs within a node in the cloud can be (a) not connected to any other significant device (i.e., be a Disaggregated resource) or connected to one or more devices; (b) connected to CPUs, e.g., through PCIe; (c) connected to other FPGAs, e.g., through a PCIe switch and/or using direct and programmable interconnects; (d) connected to GPUs, e.g., through a PCIe switch; (e) connected to ASICs, e.g., through multiple potential forms of connectivity depending on the ASIC; (f) connected to storage devices through the device-specific interface, e.g., SPI for flash and DDRfor SDRAM. Table II presents various cloud providers along with their intra-node connectivity types and associated use case categories. For example, Alibaba Cloud offers two possibilities: customers can either manage the FPGA logic themselves, or the provider can use FPGAs to accelerate customer tasks. In this cloud environment, the FPGA can be connected to a CPU, other FPGAs, and storage, enabling flexible and efficient offloading and acceleration of workloads.

D. Main Memory

To reduce memory energy consumption, various techniques have been proposed. Some techniques utilize Dynamic Voltage Scaling (DVS, analog to DVFS) to adjust the voltage supply in modern multi-banked memory systems. Other methods focus on reducing power consumption by activating only specific memory banks, allowing the rest to remain idle. Additionally, optimization algorithms target opportunities to switch either the entire memory or portions of it into low-power modes, either during or immediately after processes are running, enhancing overall energy efficiency. In [15], 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. It is important to note that CPU caches are also power-hungry components in multicore CPUs. Some techniques exploit the CPU cache (e.g. L2 cache) to reduce energy consumed by RAM, by tuning core activity.

In neural networks, data movement across the memory hierarchy, particularly between higher (e.g., registers) and lower levels (e.g., L1 cache), significantly drives energy consumption. To minimize this, it is crucial to control memory access by maximizing data reuse at lower levels. When data is transferred from a higher to a lower level in the hierarchy, it should be reused as much as possible to reduce the frequency of future transfers and avoid costly energy operations. This approach reduces the reliance on fetching data repeatedly from energy-intensive upper memory levels, optimizing both performance and energy efficiency in neural network processing. Advanced memory technologies can significantly reduce access energy in high-density memories like DRAMs. One example is embedded DRAM (eDRAM), which integrates high-density memory directly onto the chip, avoiding the high energy costs associated with switching off-chip capacitance. This on-chip integration reduces the need for energy-intensive data transfers between the CPU and external memory. Additionally, eDRAM offers 2.85 times higher density than SRAM and is 321 times more energy-efficient than standard DRAM, making it an attractive option for energy-constrained applications such as mobile devices, AI, and cloud computing environments [29].

TABLE I
INTEL (CISC/X86) AND ARM (RISC) COMPARISON.

Feature	Intel x86 (CISC)	ARM (RISC)
Design Philosophy	Complex instruction set; more operations per instruction	Simple instruction set; focus on efficiency
Power Consumption	Higher power consumption, especially under load	Lower power consumption, highly efficient
Performance	Higher raw performance, suitable for intensive workloads	Optimized for performance per watt, mobile and low-power environments
GPU Integration	Integrated (Intel Iris) or paired with discrete GPUs	Integrated GPUs (Mali, Adreno) on SoCs
Energy Efficiency	Lower efficiency, higher performance	High efficiency, ideal for mobile/embedded uses
Best Use Cases	High-performance computing, gaming, content creation	Mobile, cloud computing, IoT, AI inference

TABLE II
CLASSIFICATION OF FPGA CLOUD ARCHITECTURES.

Cloud System	Intra-node connectivity	Use Case
Alibaba	FPGA,CPU,Storage	Customer,Provider
Baidu	CPU,Storage	Customer,Provider
Microsoft Catapult	CPU,ASIC,Storage	AaaS,Provider
Amazon AWSF1	FPGA,CPU,Storage	Customer
Huawei	FPGA,CPU,Storage	Customer,Provider
Nimbix	CPU,Storage	AaaS
Tencent	CPU,Storage	Customer

While many energy-saving techniques can be applied to individual servers in the cloud, significant energy savings can be achieved by cloud providers adopting modern, energy-efficient memory banks across all cloud components. When these improvements are scaled across the entire data center, they enhance overall energy efficiency and contribute to more sustainable cloud operations. By selecting energy-optimized hardware for memory-intensive operations, cloud providers can lower both operational costs and their environmental impact.

E. Storage: From Hard Disk to NVM

The storage system is a crucial component of any cloud infrastructure. Currently, two types of storage disks dominate the market: traditional hard disk drives (HDDs) and solid-state drives (SSDs). Among these, SSDs are recognized as the most energy-efficient storage hardware available today. They utilize flash memory, which is a non-volatile memory technology with characteristics similar to electrically erasable programmable read-only memory (EEPROM).

We can distinguish two types of transfer protocols used in SSDs: SATA (Serial Advanced Technology Attachment) and NVMe (Non-Volatile Memory Express). Older SSDs rely on the SATA protocol, which provides a maximum data transfer speed of six gigabits per second (Gbps). Although this is slower than more recent interfaces, it is still significantly faster than traditional hard disk drives (HDDs). In contrast, NVMe SSDs can achieve transfer speeds of up to 20 Gbps by utilizing the Peripheral Component Interconnect Express (PCIe) bus. NVMe SSDs typically connect directly to a computer's motherboard using the M.2 form factor, which is more power-efficient, compact, and faster than the commonly used 2.5-inch SSDs. M.2 drives do not require cables and, despite their small size, can store up to eight terabytes (TB) of data. They are compatible with any motherboard that has an M.2 slot, and when using the NVMe interface, M.2 NVMe SSDs offer some of the fastest data transfer speeds available today. SSDs are significantly more power-efficient than hard disk drives (HDDs) due to their lack of moving parts (see Table III. Additionally, 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. This efficiency in power consumption, combined with faster data access speeds, makes SSDs a preferred choice for cloud environments seeking to optimize energy use while maintaining performance.

F. Networking

Energy consumption in the cloud is significantly dependent on the types of connections used within the cloud. Network energy efficiency in the cloud minimizes the power consumption of networking infrastructure components, maintaining QoS. These connections not only impact the overall performance, but also play a key role in determining how efficiently energy is used during peak and idle times. In a typical cloud computing environment, multiple layers of networking and connectivity are involved, each of which consumes varying amounts of power.

Data centers heavily depend on Ethernet-based connections for communication among servers, storage, and networking devices. High-speed Ethernet switches and routers (e.g.,

TABLE III
POWER CONSUMPTION RANGE.

Disk type	Idle state	Reading data	Writing data
SATA SSD	0,25W to 2W	4W to 8W	5W to 9W
NVMe SSD	0.50W to 3W	2W to 8W	3W to 10W
3.5-inch HDD	5W to 7W	9W to 15W	9W to 15W
2.5-inch HDD	2W to 5W	5W to 7W	5W to 7W

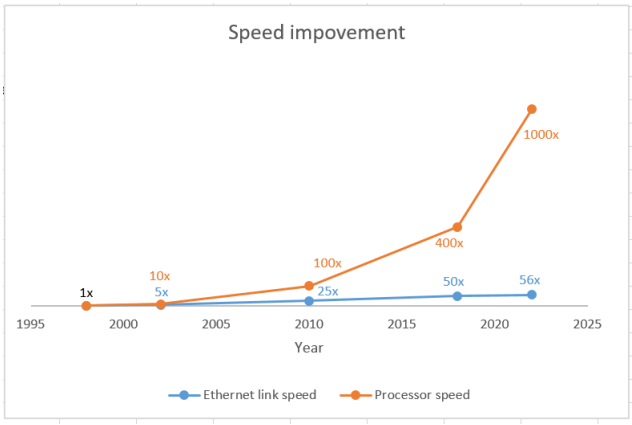

Fig. 2. Ethernet network vs. CPU speed improvement.

10/40/100 Gbps) consume considerable energy, proportional to the number of ports, with power being drawn by each port even when idle. An approximate power usage is 15W for one port, with total power consumption for a typical switch ranging from 300W to 600W. To mitigate this power demand, technologies like Energy-Efficient Ethernet (EEE) have been developed, which can reduce energy consumption by placing inactive network links into low-power states [24]. Alternatively, interconnects like InfiniBand, widely used in high-performance computing (HPC) and cloud environments, offer low latency and data rates from 2.5 Gbps to 200 Gbps. Research has shown that InfiniBand can achieve up to 30-40% lower power consumption per gigabit compared to Ethernet, making it a more energy-efficient solution in many high-throughput scenarios [25].

Some cloud data centers rely on optical fiber networks [18], [8]. Optical fiber is used for high-speed, long-distance data transmission within and between data centers. Although fiber optic provides faster data transmission (up to 100 Gbps) with lower latency and can be more energy-efficient per bit than copper cables, the intermediate infrastructure to manage and power fiber optic transceivers and signal amplifiers still requires substantial energy.

Cloud networks can implement sleep modes for network switches, routers, and other devices. These components can enter low-power states when network demand is low, reducing the power draw of idle components. Many high-performance cloud data centers also integrate power scaling mechanisms to adapt power usage based on dynamic real-time network demands.

Figure 2 shows the increasing gap between Ethernet link speed improvements and processor speed advancements over the past 25 years. As this gap widens, techniques such as disaggregated storage have emerged to help address the disparity. Disaggregated storage in the cloud [21] refers to a storage architecture where compute and storage resources are separated or ”disaggregated” and managed independently, unlike traditional architectures where storage is often tightly coupled with compute nodes. This approach allows cloud service providers to allocate, scale, and optimize storage resources independently of compute, leading to increased flexibility, resource utilization efficiency, and cost savings. In disaggregated storage, compute nodes (servers running applications) and storage nodes (where data is stored) are decoupled. Compute resources access storage over the network, typically using high-performance optical fiber networks. Disaggregated storage can lead to better energy efficiency. Resources are allocated based on actual usage, which prevents underutilized hardware from drawing unnecessary power.

G. Solutions to Reduce Energy in the Cloud

We now provide recommendations to help save energy in cloud computing environments. Most of our recommendations are general, but we identify a few which are specific to AI workloads. Given the multitude of factors, constantly evolving due to hardware innovations, influencing energy consumption, it is not possible to identify a single best recommendation.

Dynamic component deactivation is a key feature of cloud computing, allowing systems to efficiently adapt to changing workloads, reduce operational costs, and enhance overall resource management. Cloud platforms offer auto-scaling and dynamic deactivation capabilities that optimize resource usage and reduce costs. Auto-scaling allows components like virtual machines, containers, or serverless functions to be automatically activated or deactivated based on real-time demand, ensuring efficient scaling. Dynamic deactivation suspends unused components, minimizing consumption of compute, memory, and storage resources, which is cost-effective in pay-as-you-go cloud models like AWS or Azure. Additionally, this deactivation reduces energy consumption in the cloud, enhancing energy efficiency and supporting environmentally sustainable cloud computing practices.

Dynamic component deactivation is an effective method for energy saving in the cloud, with varying benefits depending on the use case. In e-commerce, it allows deactivating components during non-peak hours and reactivating them during high-traffic periods like Black Friday. For development

and testing environments, it reduces energy consumption and costs by deactivating unused resources. In AI/ML workflows, resources are activated only when needed for tasks like training or inference, avoiding unnecessary energy use and expenses when models are idle. This approach enhances flexibility, optimizes operational costs, and improves cloud resource efficiency.

To enhance energy efficiency through Dynamic Voltage and Frequency Scaling (DVFS) in cloud infrastructures, providers should regularly monitor workloads and dynamically adjust CPU/GPU frequencies based on real-time demand. Prioritizing low-impact tasks for DVFS can help save energy without degrading overall performance. The integration of advanced scheduling algorithms prevents over-provisioning, while aligning DVFS strategies with Service Level Agreements (SLAs) ensures stability.

GPUs like the NVIDIA A100 and Tesla T4 are ideal for energy efficiency, featuring Dynamic Voltage and Frequency Scaling (DVFS) for adaptive power management. Similarly, the AMD Instinct MI100 excels in high-performance workloads while emphasizing energy conservation, particularly in AI and deep learning applications. By selecting GPUs that support DVFS and leveraging advanced workload management and real-time monitoring tools, cloud providers can scale performance dynamically based on demand, achieving notable energy savings without compromising performance.

Integrating FPGA (Field Programmable Gate Array) technology is highly recommended to enhance performance and energy efficiency. FPGAs offer customizable hardware acceleration, ideal for workloads like AI, machine learning, and data processing. They can be reprogrammed to optimize specific tasks, reducing latency and improving throughput. Additionally, FPGAs consume less power than traditional CPUs and GPUs, making them a more energy-efficient solution in cloud environments.

For cloud providers, NVMe SSDs are an excellent choice due to their superior speed, low latency, and scalability. NVMe technology is ideal for data centers, high-performance computing, AI/ML workloads, and applications that require fast data access and high IOPS (Input/Output Operations Per Second). Providers should consider SSDs like the Intel Optane or Samsung PM983 series, which offer the benefits of NVMe for improving data transfer speeds and reducing power consumption.

Cloud providers should adopt optical fiber networks and interconnects such as InfiniBand due to their superior bandwidth, minimal latency, and high reliability. Fiber optics efficiently transmit large volumes of data, making them ideal for supporting the demands of AI, machine learning, and big data applications in data centers. Additionally, these technologies enhance overall network performance and scalability, which is crucial for modern cloud computing environments. Moreover, by combining sharing high-volume fiber optic channels instead of adding new fiber optic cable, leveraging low energy Ethernet on low demand servers, and letting users turn on dynamic voltage and frequency scaling, cloud providers can further reduce the overall energy consumption of networking infrastructure, which in turn can contribute to overall cloud data center energy efficiency and sustainability.

IV. Optimizing Energy with Software

A. Cloud Computing: Virtual Machines and Containers

Having low server utilization (i.e., server frequently idle) is inefficient because it wastes resources: data center infrastructure, hardware, and power. Exclusive access is even more wasteful. Therefore, low utilization and sharing resources motivates virtualization. Virtualization allows running several independent virtual operating systems on a single physical computer. A single physical server can run multiple virtual machines (VMs) sharing ample hardware resources (CPU cores, RAM, storage), requiring minimal additional power, resulting in lower energy consumption and data center operating costs. Containers (e.g. Docker) represent a newer, lightweight, virtualization technique that enables application programs to run on isolated virtual space, but sharing the operating system kernel. In general, containers have a faster startup time than VMs. In addition, containers have more flexibility to be orchestrated and scaled, up and down, to meet dynamic demand of analytic workloads and microservices.

When servers are consolidated, several virtual machines and containers are packed into a small number of physical machines in order to turn off or switch the status of the idle hosts to sleep mode to minimize energy consumption. It has been shown that container consolidation saves more energy than VM consolidation [11]. Another technique involves scheduling cloud instances, allowing to turn off idle machines. Some further techniques used in cloud data centers include virtual machine migration, load balancing, and workload categorization. Virtual machines are migrated when preconfigured thresholds are reached, distributing the workload evenly among various VMs, and classifying workloads according to their demand, before a server is assigned to them. Power in data centers is further reduced using machine learning algorithms, a solution explained later. The objective of dynamic power management is to allocate the minimum physical resources to VMs, but deactivating unused resources or setting them into a sleep state [16]. Live migration [9], a dynamic technique, moves a VM from one physical server to another one, scaling down or scaling up hardware resources according to demand. Live migration can consider energy as a factor to assign resources. In short, virtualization and related techniques decrease performance (down to acceptable levels), but yielding significant energy savings.

B. Efficient Analytic Algorithms

In general, analytic algorithms are programmed and optimized to reduce computation time at all cost, using all available computer resources. Fast machine learning algorithms, that require less iterations or fewer math computations, can achieve significant energy savings. Stochastic gradient descent (SGD), the dominating objective function optimization method in AI, is more energy-efficient than older iterative methods or

batch gradient descent (the default). A related technique during neural network training is model pruning, which removes unnecessary neural network parameters (or neuron/vertex connections). This optimization reduces computation time and energy consumption, with minor accuracy decrease. 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 in the cloud, but it is important when the neural network is deployed to make predictions on the edge (a device). A last technique worth mentioning, is quantization, which reduces numerical precision (e.g., using 16-bit floats instead of 64-bit double-precision numbers), but maintaining acceptable model accuracy.

C. Operating System

The Linux kernel plays a significant role in the new wave of embedded and mobile devices, in addition to cloud servers [3]. 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 [28], 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.

D. Extended Cost Models Combining I/O and Energy

Here we describe energy savings in data systems, following classical cost models from database systems in [7]. Energy cost can be considered for both transactional and query workloads. Database transactions have a substantial impact on overall energy consumption because they are CPU bound.

In general, the energy consumption of analytical job J is the sum of energy used by each hardware component: CPU, RAM, storage, and network (equation 3), where J is a collection of CPU and I/O operations (vector/matrix multiplications, file scans, file merge, file filtering/search).

$$E(J) = E_{CPU}(J) + E_{RAM}(J) + E_{IO}(J) + E_{NET}(J). \quad (3)$$

In a cloud data center the equation above is generalized to M machines (e.g. a cluster of uniform CPUs), disaggregated storage (e.g. Amazon S3) [31], high speed interconnection (e.g. InfiniBand) and networking hardware (e.g. Ethernet card, fiber optic, Ethernet switches):

$$E_c = M * E_{CPU} + E_{storage} + E_{inteconnect} + E_{network}. \quad (4)$$

There are important changes with respect to a local server: each machine does not have separate 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). Moreover, we are bundling the costs of accelerators (GPU, FPGA) into the CPU costs.

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. 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 common. Other AI techniques that predict energy behavior during query processing have to be explored to set the relevant values, and to dynamically calibrate parameters when the workload changes [2]. The non-uniform nature of CPU nodes in the cloud makes ML models more difficult to fit (i.e., to estimate the equation coefficients above).

It is necessary to validate cost model accuracy. The difference between estimations given by cost models and real energy measurements can be computed with AC power measurement devices, which provide a reference value.

Extended cost models for predicting energy are used by various data systems to optimize database energy consumption [26]. The system introduced in [12] 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.

V. Research Issues

We conclude the paper by providing a tentative research agenda aiming to co-design and co-optimize hardware and software. However, we do not provide neither theory nor experiments because this paper is a survey to encourage research on energy. We identify features which are likely to impact energy and then discuss how these features can help reducing energy in a cloud environment. We conclude this section with issues beyond computer science.

A. Hardware

1) CPU Clock speed: As introduced above, Dynamic Voltage and Frequency Scaling (DVFS) are established power management techniques in both CISC and RISC CPUs. The impact on energy of these hardware features in a cloud environment running on virtual machines and containers needs further research. In addition, hardware with non-uniform specs (CPUs with different speed, far/near memory, mixed storage with SSD and HDD) make energy optimization more difficult.

2) Hardware accelerators: Hardware accelerators, including GPUs, TPUs, FPGAs are exploited for numerically intensive tasks, such as neural networks (AI), numerical methods (HPC) and gaming. But their energy consumption can be very high, especially with powerful GPUs featuring 1000s of cores. We argue that the initial stages of a data science project or small AI problems can be solved with multi-core CPUs, compromising performance, but saving tons of energy. Therefore, there is a need for new hardware architectures and more general energy cost models. The current trend using GPUs in AI poses new challenges for energy efficiency. GPUs are energy-hungry, because the number of data replicas and their capability to support fault tolerance increases CPU usage during data loading. Assigning workload dynamically to physical servers, instead of predefined instances (configurations) offered by cloud providers, can also have a signifcant impact on energy management. FPGAs are used to accelerate specific computations, but they can potentially save energy when repetitive, but expensive, computations can be transferred from the CPU to the FPGA.

3) Secondary Storage Devices: Secondary 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).

4) Using machine learning models for tuning file access parameters: ML models have been used to optimize resource allocation in the cloud, in query processing and in ML computation itself. There is significant research on learning parameters of I/O cost models for query processing. Such ML models must be extended to reduce energy, but providing an acceptable performance reduction.

5) Energy-aware edge computing: The cloud is fed with a lot of data coming from edge devices. Optimizing energy on edge devices has received significant attention, because edge devices work with batteries or little power supplies. But most research has focused on one objective: either low latency, or data privacy, or power saving. Hence we believe multi-objetive optimization is needed, to reduce energy and latency simultaneously. On the other hand, there is work on middleware architectures [4] to improve device interoperability, but tweaking the OS to save energy is still an open problem. At a lower level, compiler-level code optimizations are needed to save energy, especially with multicore CPUs with complex instruction sets.

6) Quantum computing: Quantum computing is a future alternative to save energy to solve difficult problems, at the price of a small accuracy sacrifice.

B. Optimizing Energy with Software Controlling Hardware

1) Virtualization: Physical recources (hardware) are shared among multiple virtual machines (VMs). Analyzing trade-offs increasing RAM and virtual CPUs (VCPUs) is paramount to optimize hardware usage, reducing energy overall. When multiple workloads are consolidated onto fewer physical servers, virtualization reduces global energy consumption in data centers. This problem requires further study, at the cloud instance level. On the other hand, containers (lightweight virtualization such as Docker) is popular in AI analytics. Carefully configured containers enable host machines to reach optimal resource utilization. Future research is needed to decide container and task placement on physical machines, considering CPU multicores, large memory, secondary storage, and network resources working in sinergy.

2) Saving Energy in Analytic Algorithms: Training a neural network consumes tons of energy due to heavy linear algebra computing tensor equations. As we mentioned above, large neural networks have pushed innovation with lower precision floating point arithmetic, smaller integers and quantization (binary coding) techniques, which, suprisingly, still yield accurate parameters, allowing learning larger neural networks. The energy angle deserves more attention.

3) Extended Cost Models missing Energy and I/O: I/O cost models are fundamental in query processing, but they are insufficient to reduce energy. General hybrid models, mixing energy and I/O cost are needed. From an ML side, non-linear models deserve more attention since in they have been shown to be more accurate than linear models. Moreover, extended cost models environment-level parameters should also be considered, such as hardware age, IT equipment per cubic feet, external temperature, data center air volume.

4) Interaction between Hardware and Software on AI Workloads: The interaction among hardware components, software AI libraries and the cloud infrastructure is hard to understand. Can we reuse partial computations in past neural network propagation iterations to save not only time, but also energy? Can smaller neural networks be deployed (inference) without GPUs (i.e. only with CPUs)? Is it better to have more memory in a GPU vs more RAM in CPU, given the fact that GPUs are more power hungry? To compute an LLM, architecture is better? a multi-core CPU vs distributed computing with a cluster of machines? Again, as motivated above, do lower precision floating point numbers in a larger neural network (compared to a neural network with double precision) result in lower power consumption?

C. External Factors beyond Hardware and Software

Here we summarize aspects beyond computer science: societal, economic, political and legal. A viable solution cannot ignore them.

New Economic Models: The ”polluter pays” principle should be enforced making cloud providers and users aware of energy and potential pollution caused by cloud applications. We believe cloud providers and large corporations can be be more environmentally responsible if people learn energy and pollution implications.

Service Level Agreements (SLAs): Energy consumption is likely to be another item in Service Level Agreements

(SLAs). In the past, The number one requirements in the cloud have been guaranteed performance and availability, (at an acceptable cost), but energy is now a second consideration.

A viable, long-term, solution requires a collaborative effort involving AI analytic developers and AI users becoming aware about energy usage and deciding an acceptable energy/performance level, but also cloud service providers exposing energy measurements by component, and enabling fine-grained cloud management settings and controls to reduce energy consumption. Finally, energy companies should be transparent about energy pricing, energy sources (renewable vs fossils) and long-term energy trends they observe on the IT industry.

ACKNOWLEDGMENTS

We thank Ladjel Bellatreche who explained to us how energy is measured and optimized in large-scale parallel database systems and for emphasizing the importance of extending database I/O cost models with an energy dimension.

REFERENCES

[1] Hossein Ahmadvand, Fouzhan Foroutan, and Mahmood Fathy. DV-DVFS: merging data variety and DVFS technique to manage the energy consumption of big data processing. *J. Big Data*, 8(1):45, 2021.

[2] Dana Van Aken, Andrew Pavlo, Geoffrey J. Gordon, and Bohan Zhang. Automatic database management system tuning through large-scale machine learning. In *ACM SIGMOD*, pages 1009–1024, 2017.

[3] 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. A view of cloud computing. *Commun. ACM*, 53(4):50–58, 2010.

[4] V. Balasubramanian, Nikolaos Kouvelas, Kishor Chandra, R. Venkatesha Prasad, Artemios G. Voyiatzis, and William Liu. 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)*, pages 13–18. IEEE, 2018.

[5] Christophe Bobda, Joel Mandebi Mbongue, Paul Chow, Mohammad Ewais, Naif Tarafdar, Juan Camilo Vega, Ken Eguro, Dirk Koch, Suranga Handagala, Miriam Leeser, et al. The future of fpga acceleration in datacenters and the cloud. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 15(3):1–42, 2022.

[6] Anita Choudhary, Mahesh Chandra Govil, Girdhari Singh, Lalit Kumar Awasthi, and Emmanuel S. Pilli. Energy-aware scientific workflow scheduling in cloud environment. *Clust. Comput.*, 25(6):3845–3874, 2022.

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

[8] Chris Develder, Marc De Leenheer, Bart Dhoedt, Mario Pickavet, Didier Colle, Filip De Turck, and Piet Demeester. Optical networks for grid and cloud computing applications. *Proceedings of the IEEE*, 100(5):1149–1167, 2012.

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

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

[11] Niloofar Gholipour, Ehsan Arianyan, and Rajkumar Buyya. 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*, 104:102127, 2020.

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

[13] M. Iyengar, R. Schmidt, and J. Caricari. Reducing energy usage in data centers through control of room air conditioning units. In *12th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, pages 1–11, 2010.

[14] Amir Javadpour, Arun Kumar Sangaiah, Pedro Pinto, Forough Ja'fari, Weizhe Zhang, Ali Majed Hossein Abadi, and Hamidreza Ahmadi. An energy-optimized embedded load balancing using DVFS computing in cloud data centers. *Comput. Commun.*, 197:255–266, 2023.

[15] Alexey Karyakin and Kenneth Salem. DimmStore: Memory power optimization for database systems. *Proc. VLDB Endow.*, 12(11):1499–1512, 2019.

[16] Avita Katal, Susheela Dahiya, and Tanupriya Choudhury. Energy efficiency in cloud computing data centers: a survey on software technologies. *Cluster Computing*, 26(3):1845–1875, 2023.

[17] Keith Kirkpatrick. The carbon footprint of Artificial Intelligence. *Commun. ACM*, 66(8):17–19, 2023.

[18] Mirosław Klinkowski and Krzysztof Walkowiak. On the advantages of elastic optical networks for provisioning of cloud computing traffic. *IEEE Network*, 27(6):44–51, 2013.

[19] Karlo Kraljic, Daniel Kerger, and Martin Schulz. Energy efficient frequency scaling on gpus in heterogeneous hpc systems. In *International Conference on Architecture of Computing Systems*, pages 3–16. Springer, 2022.

[20] Jiechao Liang, Weiwei Lin, Yangguang Xu, Yubin Liu, Ruichao Mo, and Xiaoxuan Luo. Energy-aware parameter tuning for mixed workloads in cloud server. *Clust. Comput.*, 27(4):4805–4821, 2024.

[21] Rui Lin, Yuxin Cheng, Marilet De Andrade, Lena Wosinska, and Jiajia Chen. Disaggregated data centers: Challenges and trade-offs. *IEEE Communications Magazine*, 58(2):20–26, 2020.

[22] Icess Nisce, Xunfei Jiang, and Sai Pilla Vishnu. Machine learning based thermal prediction for energy-efficient cloud computing. In *2023 IEEE 20th Consumer Communications & Networking Conference (CCNC)*, pages 624–627. IEEE, 2023.

[23] S. Pagani, A. Pathania, M. Shafique, J. Chen, and J. Henkel. Energy efficiency for clustered heterogeneous multicores. *IEEE Transactions on Parallel and Distributed Systems*, 28(5):1315–1330, 2017.

[24] Pedro Reviriego, Ken Christensen, Juan Rabanillo, and Juan Antonio Maestro. An initial evaluation of energy efficient ethernet. *IEEE Communications Letters*, 15(5):578–580, 2011.

[25] Lorenzo Rosa, Luca Foschini, and Antonio Corradi. Empowering cloud computing with network acceleration: A survey. *IEEE Communications Surveys and Tutorials*, pages 1–1, 2024.

[26] Amine Roukh, Ladjel Bellatreche, and Carlos Ordonez. Enerquery: Energy-aware query processing. In *ACM CIKM*, pages 2465–2468, 2016.

[27] Muhammad Sohaib Ajmal, Zeshan Iqbal, Farrukh Zeeshan Khan, Muhammad Bilal, and Raja Majid Mehmood. Cost-based energy efficient scheduling technique for dynamic voltage and frequency scaling system in cloud computing. *Sustainable Energy Technologies and Assessments*, 45:101210, 2021.

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

[29] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *Proc. IEEE*, 105(12):2295–2329, 2017.

[30] V. Volkov and J.W. Demmel. Benchmarking gpus to tune dense linear algebra. In *Proc. of IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–11, 2008.

[31] Jianguo Wang and Qizhen Zhang. Disaggregated database systems. In *Companion of SIGMOD Conference*, pages 37–44. ACM, 2023.

[32] Yong Yu. 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.

[33] Qingxia Zhang, Zihao Meng, Xianwen Hong, Yuhao Zhan, Jia Liu, Jiabao Dong, Tian Bai, Junyu Niu, and M Jamal Deen. A survey on data center cooling systems: Technology, power consumption modeling and control strategy optimization. *Journal of Systems Architecture*, 119:102253, 2021.