Politecnico di Torino

Master's degree in Engineering and Management of Sustainability and Technology
A.y. 2023/2024
Degree Session April 2024

Eco-Efficient Cloud Manufacturing:

A Paradigm for Energy Consumption Optimization

Supervisor:
Alessandro Simeone

Co-supervisor:
Paolo Claudio Priarone

Candidate:
Maria Melone
Student ID 296788

Ackowledgments

Vorrei dedicare questo spazio a chi, con dedizione e pazienza, ha contribuito alla realizzazione di questo elaborato.
Questo percorso ha portato con sé tante emozioni, crescita personale e professionale.
Ogni persona che ne ha fatto parte ha avuto un ruolo importante che ha caratterizzato questo viaggio.
Con il presente lavoro si chiude un capitolo importante della vita e vorrei ringraziare alcune persone che in particolare hanno contribuito.

Un immenso ringraziamento è volto al mio relatore Prof. Simeone Alessandro, per avermi guidato e supportato nella realizzazione di questo progetto, fornendomi chiarimenti e suggerimenti ogni qualvolta mi trovassi in difficoltà, e per essersi dimostrato un riferimento dal punto di vista professionale ed educativo.

Un ringraziamento è volto al correlatore Paolo Claudio Priarone per aver permesso la realizzazione di questo progetto.

Un ringraziamento speciale è volto all'azienda TCS per avermi offerto la possibilità di intraprendere questo percorso lavorativo e di formazione all'estero, e in particolare un ringraziamento al manager e a tutto il team di sostenibilità, per aver condiviso questa esperienza, la quale è stata parte importante della mia crescita.

Un ringraziamento speciale va ai miei genitori, Caterina ed Enrico,
che mi hanno sempre sostenuto e supportato, e che con me hanno condiviso le mie gioie e hanno dato amore e conforto nei momenti di debolezza.

Un ringraziamento speciale è volto a mio fratello Antonio, che con la sua esperienza e condivisione mi ha insegnato a scoprire nuovi orizzonti e a cercare nuove opportunità.

Un ringraziamento speciale va al mio fidanzato Michele,
che con il suo affetto e la sua personalità mi ha donato forza, coraggio e equilibrio,
facendomi sentire sempre al sicuro.
Grazie per avermi insegnato tanto e per esserci sempre stato.

Un ringraziamento importante è volto ai miei amici,
che sono una parte preziosa della mia vita e
con cui ho condiviso momenti di felicità che mi hanno riempito il cuore.

Abstract

The exponential growth of data center operations has raised significant concerns regarding their substantial energy consumption and its environmental impact. Cloud usage is increasing more and more from companies and individuals. The rising use of cloud manufacturing, innovative way of conducting operations to which companies are joining, presents concerns about its environmental impact.

In this context, this thesis work aims at developing a methodological framework to boost energy consumption optimization within cloud manufacturing infrastructures. The procedure starts with a comprehensive energy diagnostic, utilizing energy measurement methods to monitor power consumption and link the latter to different metrics measurements such as CPU usage, RAM statistics, network traffic, and more, to pinpoint energy inefficiencies and hotspots within cloud manufacturing systems. The framework continues with energy allocation on different server tasks and reach the root cause analysis passing through the problem characterization. Then, the framework reviews the diverse techniques to be used to improve energy consumption by increasing system efficiency. These corrective actions are shown to improve cause inefficiencies which are distinct depending on where the cause is allocated in the cloud manufacturing system. The heart of the framework lies in its dual capability to not only detect but also revise inefficiencies by employing mostly computer-science based techniques. Through a careful review of system logs and a critical examination of energy consumption trends, the thesis shows how effective the framework is, enabling potential long-term improvements in cloud manufacturing. This work demonstrates how management engineering principles and technology advancements can be combined to improve productivity and sustainability in the context of digital manufacturing.
Two hypothetical case scenarios are presented to show the applicability of the framework.

Index

1. Introduction 8
1.1. Justification of Research 8
1.2. Research Questions and Objectives 9
1.3. Scope and Boundaries 10
2. Literature Review 11
2.1. Methodology 11
3. Cloud Manufacturing 13
3.1. Cloud Computing 13
3.2 Virtualization 16
3.3 Containerization 16
3.4 Cloud Manufacturing 18
3.4.1. Cloud Manufacturing Architecture 19
4. Framework and Methodology 24
4.1. Energy Measurement Methods 26
4.2. Energy Inefficiency and Hotspot Identification 31
4.3. Energy Inefficiency Allocation Activities 36
4.4. Inefficiency Characterization 42
4.4.1. Metrics definition and quantification 45
4.5. Root Cause Analysis 46
4.6 Energy Mitigation Strategies 59
4.6.1. Energy Efficiency Improvement Technological Solutions 60
4.6.2. Green Cloud Computing 75
4.6.2.1. Energy Management Techniques 77
4.7. Energy Savings Quantification 87
5. Case Studies 89
5.1. First Case Study: overloaded server 89
5.2. Case Study: Idle server 96
6. Conclusions 102
References 103

1. Introduction

This section will present the motivations of this thesis work. Indeed, it is possible to understand and discover the reasons behind the thesis work. It is divided in three sections covering the following topic: Justification of research, that represents the reason why the work had been done; Research Questions and Objectives, in which is possible to understand which questions guide the elaborate work; Scope and Boundaries in which limitations of the work are discussed.

1.1. Justification of Research

Cloud manufacturing an innovative paradigm that incorporates cutting-edge cloud computing technologies into conventional manufacturing processes, which has emerged as a response to the growth of the digital economy. The integration of this industry has led to unprecedented levels of efficiency and flexibility. However, there is a serious environmental risk associated with the high energy consumption of cloud-based services, particularly in data centers. With the increasing global demand for energy and the growing importance of sustainability, the situation becomes more complicated due to the increasing prevalence of cloud-computing based operations between consumers and businesses. Indeed, the construction of data centers alongside ever-higher power needs is being driven by the rising demand from consumers for pay-as-you-go resources.

These facilities require a substantial amount of electricity to run several types of equipment, such as cooling fans, monitors, and other peripheral devices. This is way data centers are consuming more and more energy. Global data center power consumption increased from 70,000 million kWh to 330,000 million kWh between 2000 and 2007. Projections indicate that data centers only will increase their consumption from 200 TWh in 2016 to 2967 TWh in 2030[1]. The significant energy usage of data centers poses a significant challenge for cloud computing, since the emission of harmful gases like CO2 exacerbates environmental concerns [2].

Another study says that the amount of energy used for computing increased by 56% between 2005 and 2010, making up 1.1% to 1.5% of the world's total energy consumption and 2% of CO2 emissions. According to a more recent analysis, by 2020 the information and communication technology (ICT) industry, which includes computers, phones, and data centers, may account for as much as 4% of global greenhouse gas emissions, with predictions of even higher emissions in the next years. According to this study [3], the amount that ICT contributes to greenhouse gas emissions will significantly rise it is predicted that by 2040, this contribution will have increased to over 14% and between 2020 and 2025, it may double to 8% [3].

According to [4], the global trend in employing Internet networks, data centers and cloud computing was subject to a rash increase. From the following table (*Table 1*), it is possible to see the quantification of people using the internet, the amount of Internet usage, the quantity of workloads done in data center (total), the energy associated with it and the energy allocated to data transfer.

Table describing the global trend increase regarding the usage of Internet and Data center, with related energy consumption (cumulative) according to the source [4]

Table 1.Data and energy utilization statistics, adapted from [4]

	2015	**2022**	**Increase (%)**
Internet Users	3 B	5.3 B	78 %
Internet Traffic	0.6 ZB	4.4 ZB	600 %
Data Center Workload	180 M	800M	340 %
Data Center Energy Use (without crypto)	4 TWh	100-150 TWh	2300-3500 %
Data Transmission Network Energy Use	220 TWh	260-360 TWh	18-64%

The demand for sustainable solutions increases because of this surge in demand, which in turn causes data centers to consume more energy. This thesis work is motivated by the goal of bridging the gap between technological advancement and responsible environmental behavior. According to this goal, this work offers a novel framework for optimizing energy use in cloud manufacturing and acknowledges the complicated problems brought about by the widespread adoption of cloud computing. The topic addressed aims to cover a large portion of the environmental issues related to cloud manufacturing by identifying inefficiencies and implementing techniques to mitigate them, paving the way for a more sustainable digital and manufacturing future.

1.2. Research Questions and Objectives

The landscape of modern manufacturing is increasingly embracing the integration of cloud technologies, giving rise to a new era of efficiency, flexibility, and interconnection. However, this transformation has its challenges, particularly concerning the sustainability and energy consumption of cloud manufacturing systems. The emerging interest in this field stems from the critical need to address the environmental impact of company's activities and the growing demand for energy-efficient cloud manufacturing processes. In order to identify the causes of excessive energy consumption and develop strategies for its optimization, this research will explore the fundamentals of energy utilization in cloud manufacturing environments.

The methods for monitoring and calculating energy usage in cloud manufacturing environments form the basis of this study. Tracking energy use is essential for locating inefficiencies and potential improvement areas. In order to provide useful insights into energy management, this research aims to investigate the technologies and approaches that can be used to implement real-time energy measurement.

It is essential to understand the reasons why cloud manufacturing processes use a lot of energy. This entails breaking down many aspects of manufacturing processes, from the usage of equipment to the processing requirements of cloud services. The investigation seeks to identify the primary factors influencing energy consumption.

This research aims to understand the primary causes of energy inefficiencies within cloud manufacturing systems. By identifying them, the research aims to highlight areas where energy usage can be minimized without compromising on productivity or quality.

After acquiring a solid understanding of energy behavior in cloud manufacturing, the study goes on to investigate various approaches for energy optimization that can be implemented with success. The topic focuses on possible solutions that can result in energy savings by solving the primary issue or implementing corrective actions.

To summarize what was just mentioned, the following central questions serve as the basis for this research:

- How can the amount of energy used in cloud manufacturing be tracked and measured?
- What are the main reasons why cloud manufacturing uses so much energy?
- What are the primary causes of energy inefficiencies in cloud manufacturing?
- Which energy optimization strategies can be applied in a cloud manufacturing setting with success?

The goals of this thesis work are the following corresponding to previous queries:

- Create a thorough framework for cloud manufacturing energy diagnostics and measurement.
- Determine and describe the system's hotspots and energy inefficiencies.
- Make root cause analysis.
- Implement corrective measures to increase energy efficiency.

1.3. Scope and Boundaries

The scope of this thesis is confined to the development and application of an energy optimization framework tailored for cloud manufacturing systems. The research will focus on the systematic identification of energy inefficiencies, the allocation of energy resources, and the implementation of energy-saving measures. The study's limitations are acknowledged in terms of the generalizability of the proposed solutions across different cloud manufacturing platforms and the variability of manufacturing processes.
This thesis only addresses the creation and implementation of an energy optimization framework designed specifically for cloud manufacturing systems. The distribution of energy resources, the application of energy-saving techniques, and the methodical identification of energy inefficiencies will be the main areas of study. Regarding the applicability of the suggested solutions across various cloud manufacturing platforms and the variability of manufacturing processes, the study's limitations are recognized.

2. Literature Review

Cloud computing has become a common way of storing, processing, and distributing large quantities of data worldwide. As cloud services expand, so does the energy consumption of data centers, which has raised significant concerns regarding their environmental impact and sustainability. This literature review examines current research related to energy consumption in cloud computing and cloud manufacturing, trying to discover the reasons and inefficiencies that lead to high energy usage. The goal of this thesis work is to construct a framework that can identify root causes and suggest potential corrective actions for energy-efficient and reduced-energy use in cloud services.
In order to do this, deep research on themes like cloud computing, cloud manufacturing, ICT energy consumption have been done. Different studies have been conducted recently on these topics, mostly from the urgent need to find sustainable technologies and practices. Indeed, many academic papers found regarded algorithms to efficiently manage and monitor energy and workloads in computing systems.
At the beginning of the research work, a book on Green IT “Sustainable IT Playbook for Technology Leaders” of Niklas Sundberg was red in order to get introduced to the general topic of sustainable computing.

2.1. Methodology

The methodology adopted is presented in the following flowchart *in Figure 1,* which describes the approach that has been used to conduct the literature review for the thesis work regarding energy consumption in cloud. The procedure is as follows:

- Start: Initiate the literature review process.

- Selection of databases: Choose several scholarly databases to search for relevant literature. This may include databases like ScienceDirect, ResearchGate, IEEE, SpringerLink, Google Scholar, and using a general search with Google Browser for any additional materials not found in academic databases.

- Selection of keywords: Identify and select key terms that are pertinent to the research topic. Examples given are "Cloud Computing," "Cloud Manufacturing," "Energy Efficiency," "ICT Energy Consumption," and "Energy Optimization."

- Title screening: Review the titles of the papers retrieved using the selected keywords. If a title is not related to the research topic, discard the paper.

- Abstract and conclusion screening: For titles that are related, proceed to read the abstract and the conclusion of the paper. This step helps in determining the relevance of the paper's content to the research topic.

- Relevance assessment: Decide if the text is relevant to your research topic. If it isn't, the paper should be discarded.

- Publication date check: Verify whether the paper was published in or after the year 2014 to ensure the currency of the research.

- Analysis: If the paper is relevant and meets the date criteria, proceed to analyze the scientific paper thoroughly.

- Categorization: After analysis, categorize the paper according to how it will fit within the broader context of your literature review.

- Knowledge extraction: Extract knowledge, data, findings, and insights from the paper to be included in your literature review.

- End: Conclude the literature review process once sufficient information has been gathered and categorized.

This approach ensures that the literature review is comprehensive, up-to-date, and relevant to the research topic. It is an efficient way to screen through large volumes of literature papers and select the most useful ones for this work studies. The methodology is shown in the following flowchart in *Figure 1.*

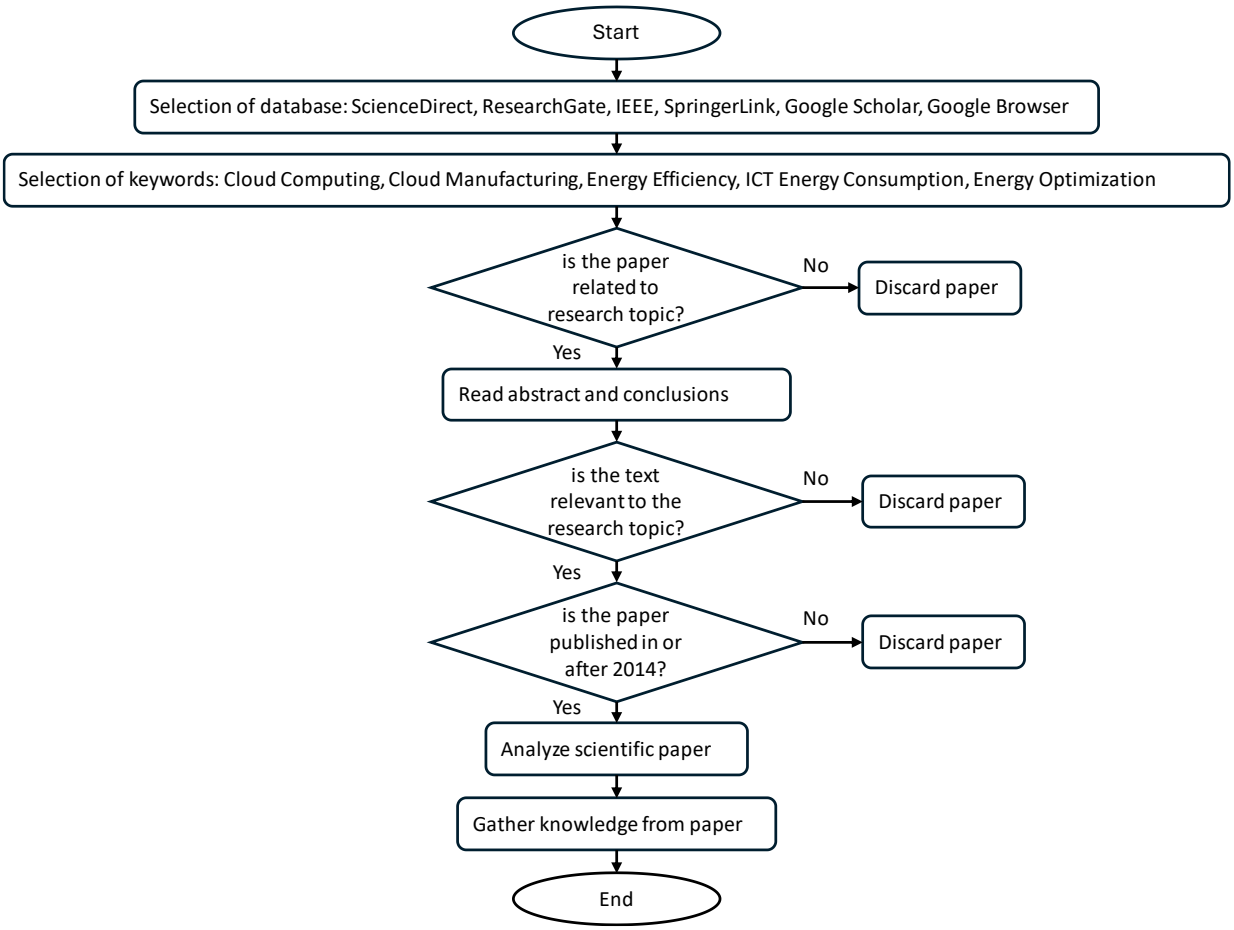


Figure 1. Flowchart representation of methodology adopted for literature review.

3. Cloud Manufacturing

Cloud manufacturing is the new innovative paradigm to hold production process. It provides wider and on-demand services thanks to the employment of cloud computing resources and connectivity. Cloud computing technologies allow for a scalable and flexible way of working and conducting operations, in a way never seen before. In the following paragraphs there is an explanation and description of how cloud manufacturing is built and how it is able to provide operational efficiency.

3.1. Cloud Computing

Cloud computing is a technology providing on-demand services by offering the availability of computing resources over the Internet. It is a convenient support for companies so that they don't have to manage infrastructure themselves but only pay for the employed service [5].
Cloud computing is a widely used, resource-rich, adaptable, and flexible technological model that offers users instantaneous, dependable, and flexible computing environments. It uses massive systems and storage capacities spread across multiple international locations to host cloud applications. The distributed architecture of cloud computing, which is essentially an evolution of parallel, utility, cluster, and grid computing, depends on a network of independent resources that are dispersed throughout various locations. Cloud computing is an approach that offers widespread, practical, on-demand network usage of a pool of collective and configurable computing assets (such as network storage, network connectivity, servers, services, and apps) that can be rapidly supplied and distributed with little effort from interaction with the service provider (definition deducted from National Institute of Standards and Technology) [6]. *Figure 2* shows the different types of resources that can be connected in cloud computing system to enhance services.

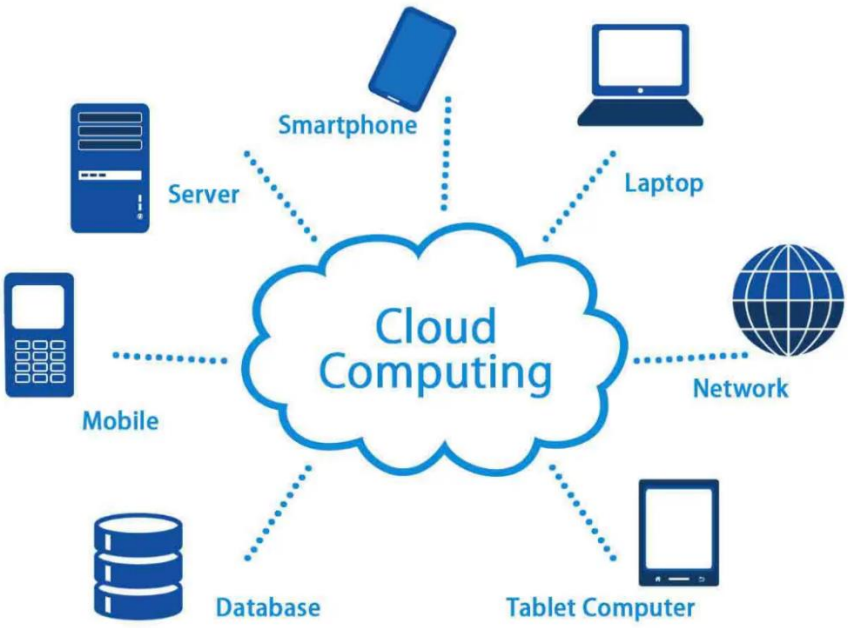


Figure 2. Cloud Computing Connectivity Devices. adapted from [7]

Cloud computing can be employed in different models:

1. Software-as-a-Service (SaaS): this is a form of cloud computing where vendors offer software and resources online, making them available through internet-based services rather than through direct installation on individual computers [8].

2. Platform-as-a-Service (PaaS): this service model facilitates the sharing of a computing platform or environment, enabling the deployment of software and applications. Users access this shared platform via the Internet [8].

3. Infrastructure-as-a-Service (IaaS): this approach involves distributing computing infrastructure, such as servers, storage, and networking hardware, to users over the Internet, allowing them to access and manage physical computing resources remotely [8].

The user has different capability to control their data depending on which model adopts. Major control is possible with Infrastructure as a service, then it reduces accordingly to the type of service, so in the platform as a service there is a medium control over its data, while in software as a service the control is very low, as shown in *Figure 3*.

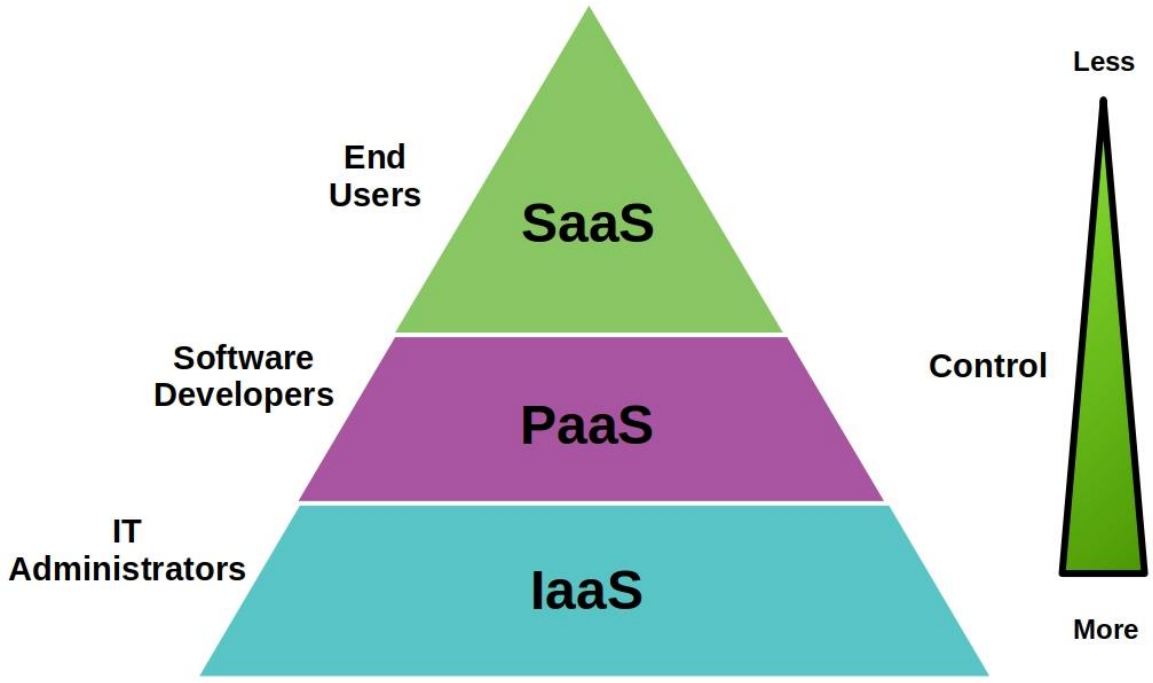


Figure 3 Cloud Computing Service Models, adapted from the source [9]

The National Institute of Standards and Technology (NIST) identifies five key attributes that define cloud computing. These include [10]:

1) The capacity for users to autonomously allocate resources as required, eliminating the need for direct communication with the service provider (on-demand self-service).

2) The ability to use a variety of devices to access services via the internet (broad network access).

3) The capability to utilize a multi-tenant model to aggregate the provider's computing resources to serve several clients, with various virtual and physical resources being dynamically allocated and reassigned in response to customer demand (resource pooling).

4) The ability of resources to quickly expand or contract in response to demand, enabling systems to scale up or down (rapid elasticity or expansion).

5) The capacity of keeping records of each customer's resource usage for reporting, monitoring, and control, which offers transparency to both the service supplier and the users of the service they are consuming. (measured service) [10].

Cloud computing can be deployed in different ways, in particular there are four models employed to enable the usage of this resource, as shown in *Figure 4*: public clouds, private clouds, community clouds, and hybrid clouds. The first one is a computing model that enables the general public to access for free available resources through the internet, as provided by the service host [8]; the second one is a computing model that creates a private and secure cloud environment exclusively for the use of a specific client like a company or its affiliates [8] [6]; the third one provides collaborative resources aimed at groups of people with similar interests [6]; while the fourth one combines elements of both private and public environments to perform separate functions within an organization [8] [6].

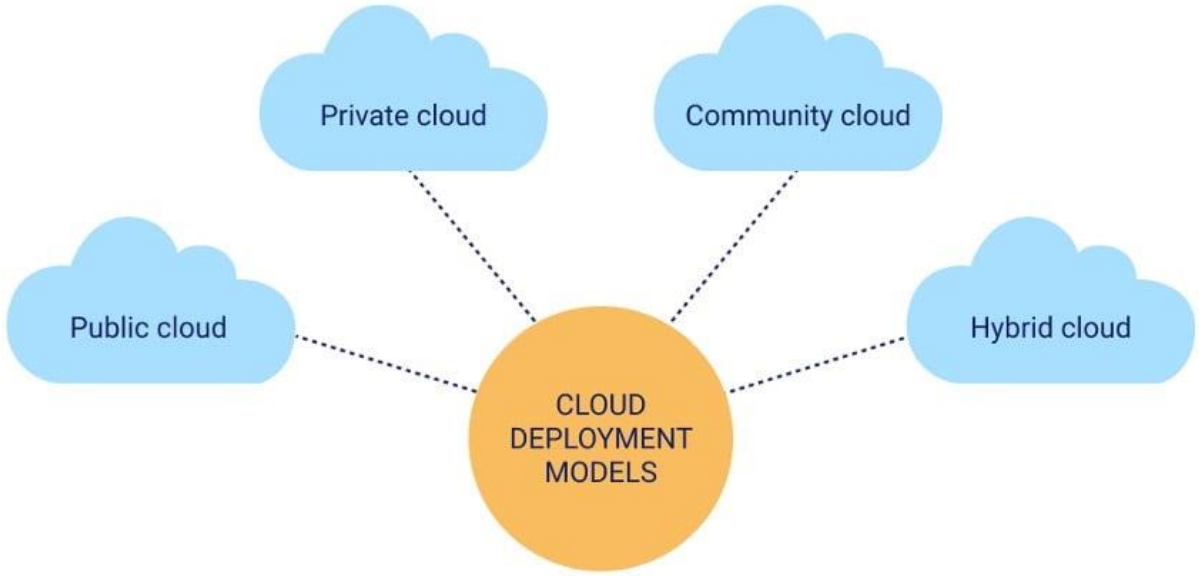


Figure 4. Cloud Computing Deployment Models, adapted from source [11]

3.2 Virtualization

Cloud computing enables an innovative and efficient feature called virtualization. This technology is able to create a virtual version of physical resources, such as hardware system. These virtual resources may include, but are not limited to, operating systems, storage capacity, main memory, processing power, and internet bandwidth. Usually, this technology is used to create virtual machines (VMs), which simulate computer systems and provide a range of resources, including storage, processing power, networking, and platforms. When a job is sent to the cloud for processing or any other purpose, one or more virtual machines (VMs) located at Cloud Service Providers (CSPs) facilities carry it out. Consequently, each task that is sent to the cloud is handled by one or more virtual machines (VMs). Many virtual machines (VMs), each operating as a distinct logical entity, are housed on a single server, which is called the host. There are many hosts (servers) in a data center, and one service provider can own different data centers [7]. *Figure 5* shows an illustration of how the virtualization works by creating different virtual machines on a physical server.

Figure 5. Illustration of how virtualization works, adapted from[12]

3.3 Containerization

Another way to create a virtualized entity is permitted by another feature in cloud systems called containerization. Containerization represents a lightweight version of the previous virtualization solution, that enhances the distribution and operation of application services across various platforms such as edge, fog, cloud, and IoT environments. It is distinguished by its ability to create isolated operating system environments within a single host, enabling multiple workloads to share the kernel efficiently. Kernel is the core part of operating system which control and manage the hardware part and assign them to different running applications. This approach significantly reduces administrative expenses since only one OS needs to be monitored for updates for safety problems, among other things. Containerization offers several advantages, including efficiency in storage and resource use, performance efficiency, cost-

effectiveness, portability (ability to move applications from one computing environment to another), energy efficiency (this topic will be explored in the "Green Cloud Computing" section), and quick startup times compared to traditional machine (VM) environments virtual. By allowing for a higher density of applications per physical server and reducing the need for extensive hardware, containerization contributes to more energy-efficient and cost-effective data center operations [1].

According to [1], the several differences between virtualization and containerization technologies are shown in *Table 2*.

Table 2. Virtualization vs Containerization, adapted from [1]

Parameter	**Virtualization**	**Containerization**
Guest (OS)	Multiple different OS by Hypervisor on the same server	Containers share the same OS and its Kernel
Performance	Small overhead	Almost no overhead
Security	Based on hypervisor implementation	Built-in security characteristics
Isolation	Virtual machines are separated to other VM and OS host; information cannot be shared	Containers are separated but sharing of data is possible
Start-up time	Higher than container	Lower than VM
Storage requirement	VMs take more storage	Containers need less storage

In the following picture (*Figure 6*), there is a schematic illustration to visualize the mentioned difference between physical server, virtual machine, and container architecture.

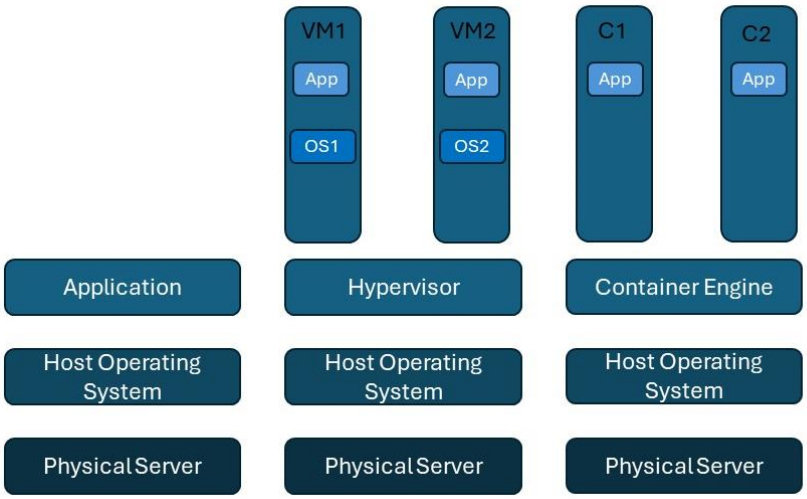


Figure 6. Architectural differences between physical server, virtual server, containerized server, adapted from[13]

3.4 Cloud Manufacturing

Cloud manufacturing is defined as a new paradigm for manufacturing that combines cloud computing technology and advanced manufacturing processes. With this novel approach, traditional manufacturing resources and capabilities are intended to be transformed into highly adaptable, scalable, and service-oriented cloud-accessible entities. Cloud manufacturing offers a range of manufacturing services that are able to be dynamically customized to meet shifting customer and market demands by utilizing the vast features of the cloud. In fact, producers and customers can connect via a cloud-based platform, making it simpler and more direct to provide manufacturing services.
The cloud manufacturing sector provides important advantages that are essential to this innovative manufacturing paradigm. Cloud manufacturing can be adopted by all type of enterprises, from small to large, and this will change the type of application model of cloud manufacturing. Moreover, there is also a distinction between cloud services adopted by Industry 4.0/smart manufacturing only to some extent, while Cloud Manufacturing is fully reliant on cloud computing support [14].
In addition, cloud manufacturing integrates the Internet of Things (IoT) to allow for remote communication with manufacturing assets. Cyber-Physical Systems (CPS) are essential in such circumstances because they allow physical assets to be created as digital twins for online management. This type of virtualization enables the planning of maintenance and repair tasks, proactive malfunction detection, and continuous asset condition monitoring. These kinds of skills contribute to the building of a large knowledge base that assembles perspectives from numerous companies in various industries. This database supports enhanced procedural analysis, enhancements, and corrections, which promotes an environment of continuous learning and optimization [15].
At its core, cloud manufacturing aims to offer a variety of production assets as services (XaaS), enabling clients and resource suppliers to efficiently connect and fulfill manufacturing demands. This model enables scalable production without huge capital expenditures because businesses can adapt their cloud-based capabilities without needing to make substantial financial investments in additional machinery, extra transportation, or larger storage facilities. Because of this, CM provides a "pay-per-use" method of accessing manufacturing resources, eliminating the overheads connected with traditional manufacturing methods, which typically require complex contracts for outsourcing and subcontracting between companies [15].

The relationship between cloud manufacturing models (SaaS, PaaS, IaaS) and the allocation of physical, platform, and software resources in manufacturing is described as follows and it is illustrated in *Figure 7* [16]:

- Software as a Service (SaaS): SaaS applications operate on cloud platforms and infrastructure and are primarily used by end users. Systems such as CAD (Computer-Aided Design), CRM (Customer Relationship Management), MES (Manufacturing Execution Systems), ERP (Enterprise Resource Planning), and SCADA (Supervisory Control and Data Acquisition) are examples of these. Because these software solutions are distributed online, users can access effective tools without having to install them locally.

- Platform on Demand (PaaS): Providing a set of higher-level services that abstract a large portion of the complexity involved in handling the infrastructure itself, because PaaS are

supported by IaaS. This covers a range of platforms for operation and maintenance, testing, and development of products. Without having to worry about the infrastructure, developers can create or modify applications by building upon the structure that PaaS offers.

- Infrastructure as a Service (IaaS): this is the foundational layer, and it is made up of physical components like servers, data centers, networks, production lines, sensors, and storage. Within the context of cloud manufacturing, these resources are virtualized through algorithms and offered as a resource pool that can be distributed to different users and applications as needed. IaaS provides the raw hardware of computers for various applications, which support the previously mentioned platforms.

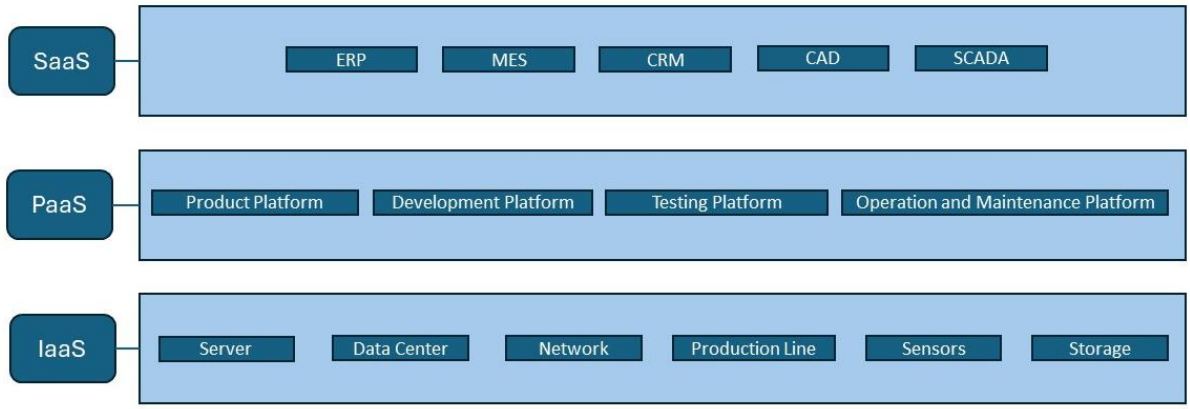


Figure 7. Cloud Manufacturing Distribution Models associated to physical, platform and software resources, adapted from [16]

3.4.1. Cloud Manufacturing Architecture

In order to provide a better understanding of how the cloud manufacturing works, based on previously mentioned features and technologies distribution, it is possible to explain and add more information to give a broader perspective on infrastructure. According to the source [17] several technologies are necessary to enhance the industrial manufacturing processes, focusing on the interplay between Service-Oriented Architecture (SOA), the Internet of Things (IoT), and Cloud Computing. The way these components are connected, and their function is explained below and represented in *Figure 8* [17]:

- Cyber-physical Systems (CPS): Cyber-physical systems architecture which is the technological foundation of the plant. These systems combine network and computational operations with physical operations. Because CPS are connected to IoT devices, it is possible to monitor and control physical operations in the production environment in real time.

- Internet of Things (IoT): IoT technology is one of the main elements connecting cyber-physical systems and the cloud. It makes it possible for devices to gather and share data, which makes it easier to connect cyber-physical and cloud systems. The architecture's

objective of enhancing data processing, data analysis, and remote management capabilities is supported by this integration.

- The Cloud: the cloud is the fundamental framework that places and oversees a variety of services from databases to messaging, from business management software like ERP to industrial control systems like SCADA and MES, and to analytical tools like business intelligence. Its role in the system is to provide an area that can expand and change to accommodate the storing, processing, and analysis of data [17].

- Clients: clients are devices that employ cloud services for human-machine interaction, and as such they are a part of the broad architecture. These clients allow users to communicate with the system, track procedures, and take reasonable choices based on real-time data. These clients can be desktop computers, tablets, or mobile devices [17].

The architecture's design allows for seamless interaction among these components, ensuring efficient data flow from Cyber-physical systems through IoT to the Cloud and finally to the end-users via clients. This interconnected setup enhances the agility, stability, and interoperability of industrial systems, making it adaptable to the diverse needs of manufacturing environments [17].

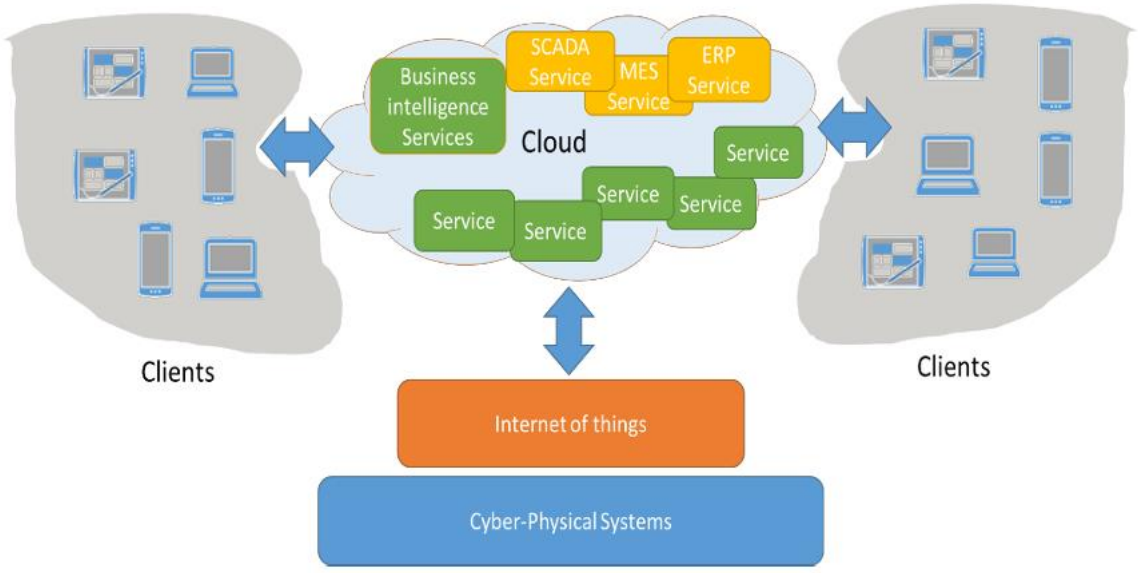


Figure 8. Cloud Manufacturing Architecture, adapted from[17]

There is also another type of system infrastructure in cloud manufacturing (*Figure 9*) which includes edge-fog layers, according to the source [16]. It consists of several key layers, each playing a distinct role in the data processing pipeline from IoT devices to cloud servers. The architecture in this case is built with same previous components plus fog and edge elements [16]:

- Cyber-Physical Systems Layer: as already mentioned previously, these are Intelligent systems that merge computational and physical processes. These systems can monitor and regulate the production floor's physical processes in manufacturing thanks to information feedback loops that let computations influence physical processes and vice versa [16].

- IoT Layer: The industrial environment is embedded with Internet of Things (IoT) devices, which comprise a range of sensors, actuators, and mobile devices like tablets and phones. They oversee collecting information from the manufacturing facility regarding the performance of machines, the state of the environment, the state of production, and other things [16].

- Edge Computing Layer: this layer consists of servers that are located is closer to the data source. These servers have near-instantaneous processing speeds for the data gathered by IoT devices. Because of their close proximity to the data sources, reactions and measures can be taken quickly, which is crucial for manufacturing processes that depend on deadlines. In order to enable interaction with the devices and possibly integrate with other systems like the GPS (Global Positioning System) for services based on location, these edge servers have connections to base stations [16].

- Fog Computing Layer: Fog Computing acts as an intermediate layer between the edge servers and the cloud data centers. It is a distributed computing infrastructure that provides data, compute, storage, and application services closer to the client or near-user edge devices. Fog computing consists of distributed servers which allow for more scalable and context-aware data processing. It uses fog nodes, which can handle more data and perform more complex processing than edge servers [16].

- Cloud Server Layer: Finally, at the top of this architecture, there is the centralized cloud servers. These powerful servers perform the most resource-intensive computations and store vast amounts of data. They are responsible for long-term data analytics, overseeing the entire manufacturing process, and making strategic decisions based on the aggregated data received from the fog and edge layers [16].

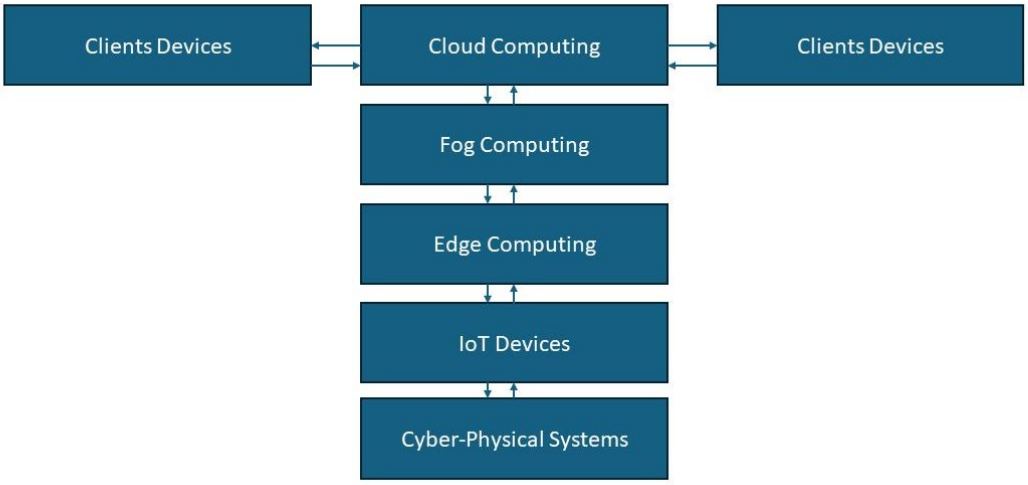


Figure 9. Cloud Manufacturing Architecture with Edge-Fog layers, adapted from [16]

In order to better understand and define the different layers activity regarding data workload, a description is made for this aim. So, as previously mentioned, data flow starts at the IoT devices, thanks to their ability to gather data from manufacturing machines and floor, and then it moves through the edge computing layer for initial processing. After, data passes to the fog

computing layer for more complex tasks and local storage, and finally to the cloud layer for global analytics and decision-making. After processing, commands or insights can be sent back down through the layers to affect changes in real-time operations.

Data-driven manufacturing represents a significant shift in how the industry approaches production, moving away from traditional, model-based methods to a more dynamic, data-centric model. This transformation is primarily given by the exponential increase in data generated through various manufacturing processes and the parallel development of sophisticated technologies for data collection, processing, and analysis [15]. The information system process is possible thanks to several technologies embedded in cloud manufacturing. As mentioned previously, cloud technology is an important part of this flow, but cloud manufacturing technology is granted by other technologies such as big data analytics, Internet of Things (IoT), and machine-to-machine (M2M) communication, used to enhance data transfer and communication speeds. Big data analytics makes it feasible to track and assess each stage of the production process, enhancing the analysis of additive manufacturing (AM) procedures and facilitating the integration of AM with supply chains, for instance. The information exchange between devices is improved by M2M communication. Thanks to these innovative technologies, cloud manufacturing is able to adapt and modify systems. Indeed, the main goal of CM is to make sure that computing services are consistently available, scalable, and dependable in a spread industrial environment [14].

The following scheme (*Figure 10* [18]) shows the different layers and steps required by the informative process.

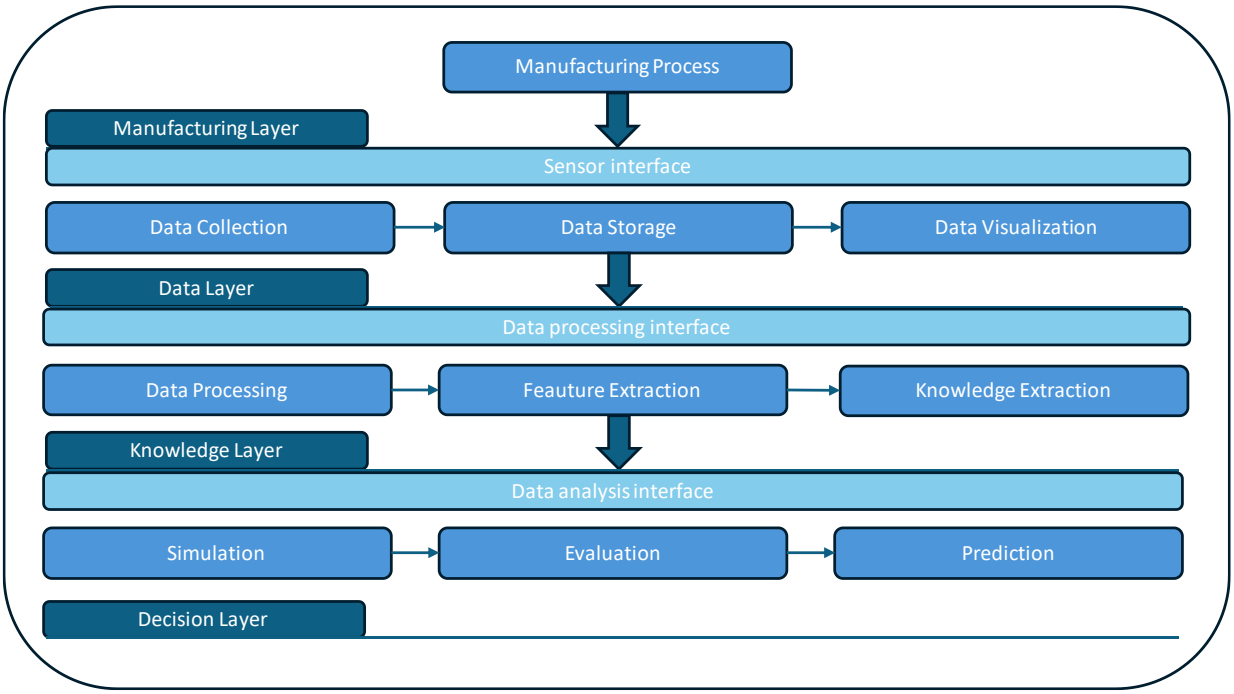


Figure 10. Data flow from the manufacturing layer to the decision layer, adapted from [18]

In the layered architecture of data-driven manufacturing, the process begins at the manufacturing layer where various sensors and data acquisition systems are employed. These systems are fundamental in capturing a wide array of data points, ranging from the functionality of the machines to the ambient environmental conditions, and even to the intrinsic properties of the materials being handled. This layer's crucial role is to provide a consistent and detailed data flow, which encapsulates aspects of the production process [18].

Moving on to the data layer, the massive amount of data is first stored before going through a thorough refining process. In order to prepare for the subsequent, more in-depth analytical procedures, this step entails clearing out any anomalous data or unnecessary information. The system uses a combination of advanced machine learning algorithms and statistical models to analyze this carefully selected dataset and extract meaningful information, such as trends in machine performance, process flow inefficiencies, and product quality assurance. At this point, identifying patterns, predicting possible problems and developing suggestions to adjust the manufacturing trajectory are the main objectives [18].

Machine learning and artificial intelligence are prominent in the knowledge layer, which is a crucial component of the data-driven paradigm. In this case, data analysis moves beyond machine learning and into the domain of predictive intelligence, where systems continuously learn from past data patterns. In addition to being skilled at predicting future events like equipment failures or breaches in product integrity, this automatic learning environment additionally has the ability to recommend proactive interventions [18].

Lastly, the system's ability to make decisions in real time is highlighted at the decision layer. Equipped with real-time, continuous information from the manufacturing processes, the system can make quick adjustments to maintain optimal efficiency. Real-time adjustments to machinery parameters, precise resource reallocation, or revisions to production schedules to increase productivity, reduce waste, and ensure consistent product quality are all motivated by strategic analysis and insights obtained from the framework's layers. The ultimate achievement of the data-driven manufacturing is this real-time decision-making ability, which guarantees a flexible, clever, and extremely effective production environment [18].

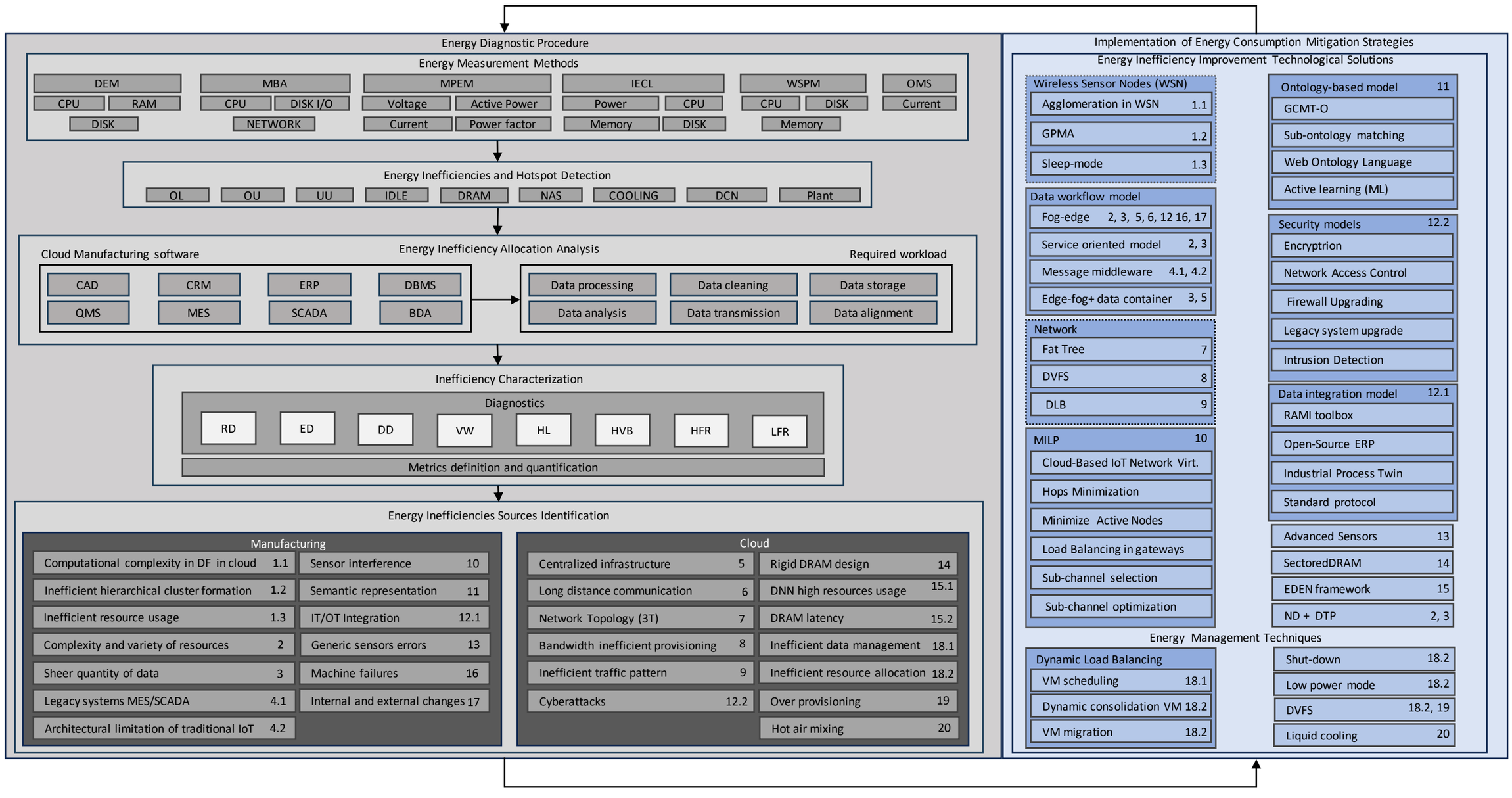
After the comprehensive knowledge of how cloud manufacturing system works and what is the pipeline data has to make in this infrastructure, it is possible to study and analyze the environmental impact of these activities in detailed.
In line with the research objectives illustrated in Section 2, in order to fill the literature gap illustrated in this section, this thesis proposes a methodological framework aimed at finding the roots of inefficiencies which increase the environmental impact of the system and optimizing them through the use of specific solutions, which will be detailed in Section 4.

4. Framework and Methodology

The procedure starts with monitoring energy consumption values directly through the use of sensors. If the energy usage is different from the one expected, or the energy trend increases, or if the consumption is too high, it is necessary to discover the reasons behind this behavior and act accordingly.
The importance of this action not only addresses the issue of related money expense, helping the firm to have lower utility costs and higher profit, but mostly to reach sustainable goals and lower the burdens that the environment is facing, which could add also an advantage for the company in terms of image and visibility.

The energy monitoring happens both at manufacturing floor and at cloud manufacturing servers.
Regarding the latter, it is useful to understand how the energy consumption is associated to servers' behavior. One way to accomplish this relation is to also monitor servers' fundamental parameters during the same time period and check for correlations.
During this monitoring phase, there are several methods, shown in the box of *Figure 11*, that can link the energy consumption to the values of indirect parameters of servers such as CPU (Central Processing Unit), RAM (Random Access Memory), and disk Input/Output.
The second step is to find what is the hotspot by analyzing the previous data. There are some possible and not exclusive inefficiency states leading to high energy consumption: server overload (OL), server overutilized (OU), server underutilized (UU) and idle server, the number of active servers (NAS), or the hotspot can be associated to the DRAM itself, the Data Center Network or the manufacturing plant (machines).
Once the hotspot is identified, the next step is to make an energy consumption allocation with the corresponding activity that is loading to better understand where the faulty reason arises.
After all, the inefficiency characterization takes place. In this phase the aim is to visualize and quantify the inefficiency associated with the energy consumption. Depending on the case, it is possible to monitor or make an analysis of production data in order to check these inefficiencies. In the box "Diagnostic" the following inefficiencies are exhibited: redundancy data, dirty data, variability of workload, high latency, high bandwidth usage and high frequency of requests. Also in this case, it is possible that more than one is present.
When this is done, the next step is to make a root cause analysis with the aim to understand what the primary cause of inefficiency is to then be able to solve it. These findings are divided into two box, trying to underling where at the infrastructure level the issue was born.
After the root issue has been identified, it is possible to implement one or more solutions specifically for that cause.
When the corrective action has been applied, it is important to go back on the first step to monitor the energy consumption, check and quantify the improvement.
Figure 11 in the next page illustrates all the methodological steps of the framework, providing a global overview of the procedure for energy consumption optimization in cloud manufacturing.

Energy Consumption Optimization Framework



KEY:
DEM=Distibuted Energy Metric; **MBA**=Measurement based Approach; **MPEM**=Multi-Port Hardware Energy Meter; **IECL**=Intelligent Energy Consumption Model; **WSPM**=adaptive workload-aware power consumption measuring method; **OMS**=online monitoring system; **OL**=server overload **OU**=server overutilization **UU**=server underutilization **IDLE**=inactive server **DRAM**=Dynamic Random Access Memory **NAS**=number of active servers ; **DCN**=Data Center Network **CAD**=Computer-Aided Design **CRM**=Customer Relationship Management **ERP**=Enterprise Resource Planning **DBMS**=Database Management System **QMS**=Quality Management System **MES**=Manufacturing Execution Systems **SCADA**=Supervisory Control and Data Acquisition **BDA**=Big Data Analytics **RD**=redundant data **ED**=excessive data **DD**=dirty data **VW**=workload variability **HL**= High latency **HVB**=High variability bandwidth **HFR**=high frequency of requests **LFR**=Low Frequency of requests **DF**= Data fusion **IoT**=Internet of Things **IT**=Information Technology **OT**=Operational Technology **3T**=3 Tier **DNN**=Deep Neural Network **GPMA**=Generalized Particle Model Algorithm **DVFS**=Dynamic Voltage and Frequency Scaling **DLB**=dynamic load balancing **MILP**=Mixed Integer Linear Programming **VM**=virtual machine **GA**=genetic algorithm **GCMT-O**=General Cloud Manufacturing Task Ontology **ML**=machine learning **RAMI**=Reference Architecture Model Industrie **EDEN**=Energy-Efficient High Performance DNN using approximate DRAM **ND**=Neighborhood Density **DTP**=Decision Tree Progression

Location
Model

Figure 11. Energy Consumption Optimization Framework flowchart.

4.1. Energy Measurement Methods

The first step proposed by the framework is the energy measurement. This phase is extremely important as it is the starting point that at the end will be used as a comparison.
As Peter Drucker's statement says, "You can't manage what you can't measure". This sentence implies that improvement is achievable only by having measurements to do comparison afterwards. This is why the energy optimization procedure starts with the monitoring phase. Indeed, this axiom guides the development of this framework, which is intended to improve the operational effectiveness and sustainability of cloud manufacturing environments.
During this phase, it is feasible to overview the power state and check if the expected value of consumption is overcome, or if there is an increasing and unexpected trend. If this is the case, following the procedure is a must to understand the power behavior faults and to apply the corresponding solution. In order to do this, it's important to create a relationship between energy usage and servers' behavior. This means to find a link for which is possible to associate the high energy consumption to some specific values of servers' parameters.
The first step to do is to track energy behavior thanks to dedicated sensors that are able to measure it.
Also, it is essential to adopt an accurate and reliable mechanism for measuring energy utilization in order to comprehend and control the intricate dynamics of cloud manufacturing environments. Simultaneously, it is useful to monitor operational server parameters and then it is feasible to make a relation between the two.
The framework establishes a strong basis for comprehensive analysis and optimization by capturing the core range of parameters, such as CPU utilization, memory utilization, disk I/O operations, and network traffic.
During the research done, it was possible to put in evidence some of the several energy measurement techniques that are in line with the previously mentioned.
The first method proposed by the framework when it comes to tackling the challenges of tracking and measuring energy use in diverse cloud computing environments is the Distributed Energy Meter (DEM) approach. Cloud computing environments provide particular challenges for energy management because of the wide range of hardware, software, and workload configurations they exhibit. The DEM system is designed to work with this variety, offering precise evaluations of energy consumption by taking into account the unique qualities of different server architectures, operating systems, and application requirements [19].
Within heterogeneous cloud infrastructures, hardware varies in terms of CPU architecture, memory capacity, and storage technologies. These differences also change over time as new machines are added to the mix alongside older models. There are many different versions of Linux and Windows new technology in use, and each one has its own resource management and monitoring features. Operating systems also differ greatly from one another. Workloads also cover a wide range of activities, from I/O-intensive projects like web services and databases to compute-intensive tasks like scientific simulations [19].
The DEM system has the ability to estimate the energy consumption of CPUs and cache, memory, and disk operations through the use of a multi-component power consumption model. With its ability to adjust to the unique setups of a system, this model provides accurate estimates of energy consumption on various platforms [19].
This system is capable of estimating the energy consumption of these components by creating the mathematical relationship between their resource usage levels and the resource power consumption. This means that the DEM system monitors the usage levels of these components

to provide a more comprehensive understanding of how resource utilization impacts energy consumption [19].
The dynamic CPU power consumption model of the DEM, which adapts to the unique architecture and operating state of the machine, and its innovative disk power consumption strategy that distinguishes between sequential and random I/O operations are crucial to its efficacy. To appropriately account for the energy impact of various workload types, this kind of differentiation is essential.
The architecture of the DEM system uses a specific type of configuration, in which there are “slave” nodes responsible for monitoring energy consumption metrics from specific cloud servers in real time. The data is then combined by the master node to provide an overall picture of the cluster's energy usage. The communication strategy of the system is particularly creative; it combines event-triggered and periodic data transmission to improve the timeliness and accuracy of the energy data that is gathered.
So, to sum briefly, DEM system monitors the cloud server energy consumption directly using slaves nodes, then send data to master nodes that aggregate it and then establish the mathematical model to associate energy consumption to specific hardware components based on its utilization level taken from standard performance counters [19].
Extensive testing has demonstrated that the DEM system performs noticeably better than current energy consumption models, particularly when it comes to precisely estimating the power consumption of disks in intricate cloud environments. This superiority highlights both the system's sophisticated architecture and its potential to support cloud computing practices that are more energy-efficient [19].

Another methodology to monitor energy consumption is defined as a “measurement-based approach” [20] and the study was made on a Linux-based server environments for the flexibility offered. The aim of this method to determine energy consumption by understanding how the energy and power consumed by a server's CPU, disk, and network interface change under different configurations [20]. This approach starts with a collection of server operational data with high accuracy. Its main goal is to manipulate CPU frequencies by using kernel modules, benchmarks, and specialized commands. This method employs the use of the metric "Active CPU Cycles per Second" (ACPS) to measure CPU load. This is more accurate than traditional CPU load percentages, which can vary depending on frequency fluctuations. Examining CPU power usage, the method makes use of a script which is intended to simulate different CPU core loads. The method enables a profound measurement of power consumption across various configurations by methodically adjusting CPU frequency (speed in executing data processing) and load. This method, which begins at the lowest CPU frequency and increases it gradually, carefully plots the energy behavior of the CPU and offers insightful information about its patterns of power consumption.
This method is further enhanced by measuring disk power consumption, which is done with scripts that make it easier to manipulate data block sizes and processing volume precisely, resulting in the possibility to measure power consumption specifically for both reading and writing operations. By isolating the specific energy impact of disk operations, this method improves the overall comprehension of the server's power consumption. Network power consumption is examined using client-server scripts, which look at how much energy the server uses when sending and receiving data. By varying system frequency and packet sizes and transmission rates, this technique calculates the network interface card's (NIC) percentage of the server's overall power consumption. This methodology's central idea is a comprehensive perspective meant to capture the complex dynamics of power consumption among different

server components. A thorough analysis is carried out by calculating the server's total power consumption and subtracting the baseline power, which represents the energy used while the server is not in use. This analysis shows how different configurations and operational parameters affect the amount of energy used by the server [20]. To sum up, it is used a Voltech PM1000+ power analyzer, that is connected to the server and its power supply, to keep track on the server's power usage. This analyzer updates every second and tracks the total power used by the server in real time. Additionally, there is the monitoring of how much of the server's hardware, that are the network, disk, and CPU, are in use at any given moment. Firstly, it is determined the energy contribution from the CPU and then evaluate how this figure deviates from a predetermined baseline in order to analyze the distinct contributions of each part to the server's overall energy usage [20].

One research [21] shows a novel approach to monitor energy usage in cloud servers consisting in the employment of the “multi-port hardware energy meter” [21] system. This system provides a comprehensive solution for the careful monitoring of power consumption because of its capability at collecting precise real-time data on the energy usage from numerous connected devices across data centers. The system collects real-time data on voltage, current, active power (real power consumed by the servers to perform tasks and operate normally), and power factor (dimensionless number ranging from -1 to 1 that describes how effectively the electrical power is being converted into useful work output) [21]. This allows for detailed monitoring of the energy consumption behavior of each device, facilitating the identification of trends, peaks, and potential inefficiencies. The precision of the system is derived from the utilization of current sensors, an electronic circuit for data interpretation, and a Raspberry Pi 3 for data processing and communication. The system has been carefully calibrated to ensure maximum accuracy in the operating range of typical data center equipment.
The Raspberry Pi 3 (*Figure 12*) is the brain that conducts the operations, coordinating data processing and sensor-to-printed circuit board communication from its command center. It is used the Serial Peripheral Interface (SPI) as a method for exchanging data among various devices, utilizing a master-slave setup where the master device directs the slaves, and the slave devices follow the master's commands. It works to enable a communication system which can send and receive data simultaneously, offering fast data transfer speeds [22].

Figure 12. Raspberry Pi 3, single-board computer, adapted from [23]

So, after the data collection, there is the transmission to a central server using API (Application Programming Interface). This enables the storage and further analysis of energy consumption patterns, which is crucial for making data-driven decisions aimed at improving energy efficiency. In the paper it is shown the importance of making benchmarking test to validate the method, and this is done with the benchmarking platform Phoronix suite (Figure 13) [21]. An example of how this platform can screen is presented in the following image.

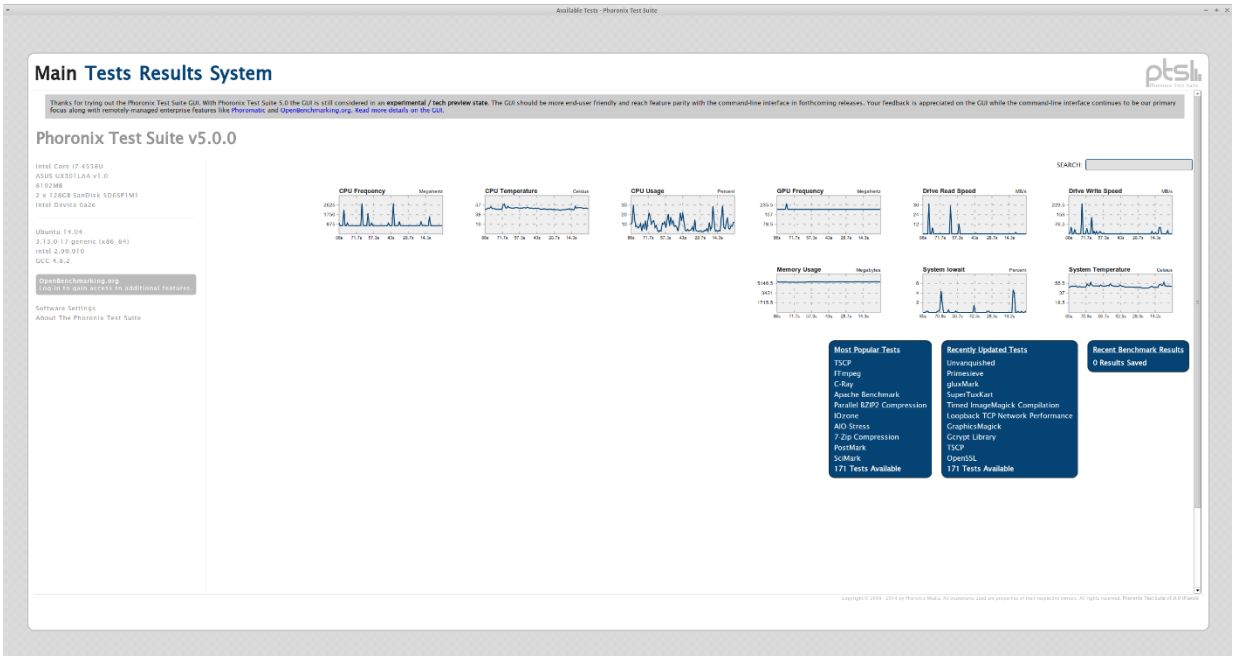


Figure 13. Example of capability of Phoronix benchmarking platform, adapted from [24]

Another innovative model defined as "An Intelligent Energy Consumption Model" [25] is used to monitor and predict energy consumption in cloud manufacturing; this approach not only makes it possible to track energy usage effectively in the present, but also establishes the foundation for predictive analytics, which can predict patterns of energy consumption in the future. The process starts with a preliminary configuration in which power meters (*Figure 14*) are connected to servers. This crucial stage guarantees the gathering of precise and comprehensive data regarding these servers' power usage. One important component of this setup is the use of energy sensors, which have a direct link to each server. These sensors were chosen because of their non-intrusive measurement capabilities, which ensure that the server's power supply integrity is maintained, and, as a result, the accuracy of the data gathered [25].
For real-time monitoring, the system uses software programs like Ganglia and Zabbix. These programs can provide a perspective on energy usage across the server infrastructure by combining software monitoring tools with hardware sensors. The proposed approach uses two machine learning algorithms, Support Vector Machine (SVM) and Random Forest (RF), to select features after gathering data on energy consumption and the parameters of operation. The model is then further refined by applying Grid Search (GS) optimization techniques. A predictive model that can predict future energy consumption based on historical data and operational parameters is the result of this analysis [25].
The significance of this predictive capability lies in its capacity to forecast patterns in energy usage, which can lead to the development of more effective and efficient energy management tactics. It is important to emphasize that this methodology's predictive component can be extremely useful even when power meters are not present anymore. Organizations can forecast energy consumption patterns without ongoing direct measurement by depending on the

established model. This feature represents a substantial development in the fields of cloud manufacturing and server control of energy by creating new opportunities for energy optimization and efficiency enhancements. Although the approach includes a predictive component, it is important to note that this section does not address its application for future energy optimization processes. The present section emphasizes the method's fundamental focus on monitoring and initial prediction instead of comprehensive optimization strategies because the investigation of these possibilities is outside the scope of this section [25].

Another approach to track energy utilization in server environments proposed in the framework is defined as “adaptive workload-aware power consumption measuring method” [26]. The WSPM method focuses on accurately modeling and measuring power consumption based on the heterogeneity of workload types, such as CPU-intensive, I/O-intensive, memory-intensive, and mixed workloads. This process is implemented in multiple critical steps, starting with the gathering server performance metrics, such as CPU, memory, and disk I/O utilization.
The method starts with constructing separate power consumption models for different types of workloads. By distinguishing workload types, this technique can apply the most suitable power model to achieve accurate power consumption estimation. In order to do this, the resource data collection is done through system performance counters, afterwards data is cleaned and normalized to form datasets for offline and real-time analysis. After clusterization and classification, based on their results, the method assignes the upcoming workload type and selects the most appropriate power consumption model for measurement.
The “Watts up? PRO”[26] external electric meter is used by the methodology to measure power consumption by recording the server's actual power usage. This direct measurement serves as a benchmark for evaluating the accuracy of the predictive models and is essential for validating the energy consumption models created during the research.
According to the study [26], tests and comparisons have been done to validate this methodology. The study aims to improve energy usage monitoring in cloud server settings by combining advanced data collection and analysis techniques with direct power measurements [26].

Another methodology to monitor energy consumption in cloud manufacturing is to check on energy consumed at the manufacturing layer. In order to do this, the starting point is the monitoring step but referring to production plant [27]. Indeed, according to the source, the deployment of an online monitoring system is critical for real-time analysis of energy flows within the plant. This system allows for the tracking of energy consumption across different departments, enabling a plant-level vision of energy use. Data-driven approach in manufacturing is important for energy monitoring, because through the collection and analysis of data from the monitoring system, the plant can identify hot-spots of energy consumption and inefficiency. This leads to a better understanding of where specific measures can be implemented to improve energy efficiency and reduce the carbon footprint [28]. Indeed, by creating a reference model of the plant's energy consumption, based on the collected data and integrated knowledge of the processes, the plant can more effectively monitor and improve its energy efficiency [28].

All the techniques presented in this work used to measure power consumption related to different physical components are illustrated in the first step of the framework as shown in *Figure 14*.

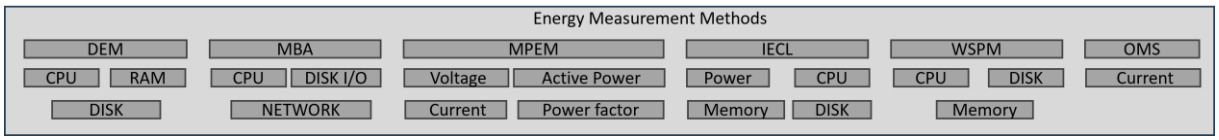


Figure 14. Energy Measurement Methods

Moreover, there is a general indicator of how well a data center uses energy is the Power Usage Effectiveness (PUE), which measures the energy efficiency of data centers by understanding how much power loss in non-IT activity is. Such model is depicted in Eq. 1

$$PUE = \frac{E_f}{E_{IT}} \tag{1}$$

Where:
$\frac{E_f}{E_{IT}}$ is the ratio of the facility's overall energy consumption to the energy consumption of the IT equipment housed within the data center.
The energy used by the whole data center, that is the numerator of the ratio, includes all functional demands like IT hardware, lighting, cooling, and power conditioning systems, can be reported by the utility meter. The electrical power consumed by servers, storage devices, network equipment, and other associated parts, such as switches, monitors, desktops, and laptops, is all included in the energy use of IT equipment (denominator of the ratio) [6].
An optimal value of PUE is 1, which means it is 100% efficient, that is impossible to accomplish. In 2020 the average PUE was 1.59, while in 2021 it was 1.57 [1]. According to [29] and [8], the common PUE value for data centers is around 1.8, which is high and shows a requirement for improvement.

Based on the insights garnered from prior methodologies, it has become feasible to track patterns of energy consumption, gather these data, and concurrently evaluate indirect indicators of hardware status. This approach enables us to decipher the connection between elevated energy usage and the condition of server components or providing useful information to identify the manufacturing department, providing a comprehensive understanding of how cloud manufacturing resources influence power consumption.

4.2. Energy Inefficiency and Hotspot Identification

The second part of the framework consists of identifying what are the energy consumption inefficiencies. According to a comprehensive study [30], in which is analyzed the energy consumption patterns within IT environments, it has been discovered that the distribution of energy usage is significantly prone towards specific components of the IT infrastructure. Remarkably, IT equipment alone accounts for a substantial 50% of the total energy consumption, which includes servers, storage devices, and network equipment, highlighting the intensive energy demands of maintaining and processing digital information.
Cooling systems, designed to mitigate the heat generated by these devices, constitute 40% of the energy expenditure. This underscores the critical role of cooling in ensuring the optimal performance and longevity of IT equipment, even if with a high energy cost.
The remaining 10% of energy consumption is attributed to other infrastructure components. This encompasses a variety of elements, including but not limited to, lighting, power

distribution systems, and the underlying building infrastructure supporting IT operations. This distribution of energy consumption shown in *Figure 15* underscores the importance of focusing on energy efficiency across all facets of IT infrastructure, from the equipment to the cooling systems and beyond, to mitigate the environmental impact and operational costs associated with technological advancements [30].

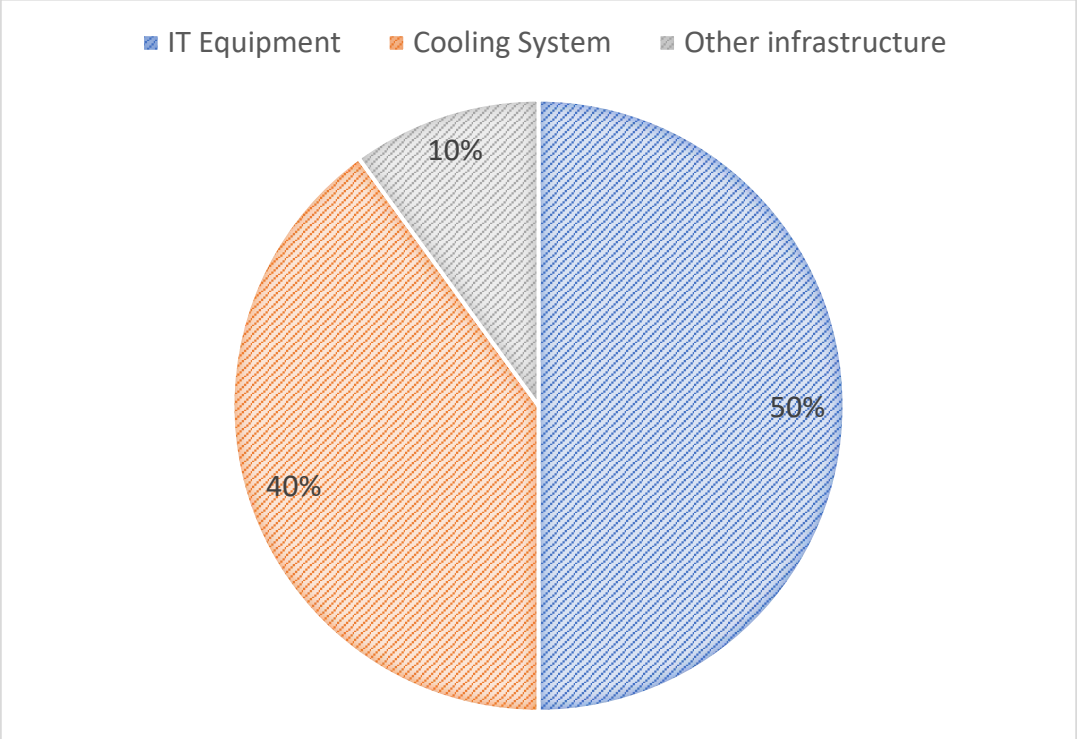


Figure 15. Energy consumption distribution in a cloud environment, adapted from [30]

Regarding IT equipment, servers represent the main object of interest. Indeed, several modes of servers are inefficient in terms of energy consumption.
The framework step highlighted in *Figure 16* shows different hotspots which affect the inefficient energy consumption of cloud manufacturing: server overload, server overused, server underused, idle server, DRAM (Dynamic Random Access Memory) hardware, the number of active servers, the cooling system, the data center network and the manufacturing plant (machines). These hotspots are going to be explained in this section and they are presented in the following picture (*Figure* 14).

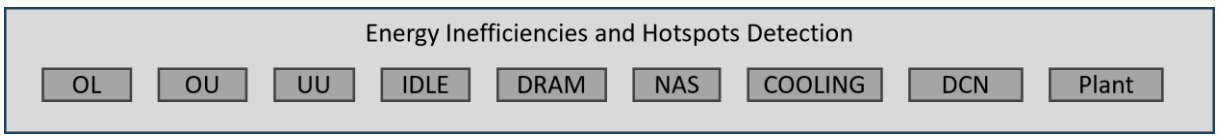


Figure 16. Energy Inefficiencies and Hotspot Detection phase of the framework

The first mentioned in the framework is server overload which occurs when a server's utilization exceeds a certain threshold, which is inefficient for energy use. This is because an overloaded server consumes more power without a corresponding increase in productive work, leading to an imbalance between energy input and output, which means the energy consumed by the server (input) does not translate into a proportional amount of computational work or services delivered (output). Essentially, the server uses more energy than necessary for the work it performs, which is inefficient. The specific threshold for inefficiency varies depending on several

factors, including the server's design, the workload type, and the data center's operational parameters. Typically, it is defined by the point at which additional energy consumption does not result in significant performance gains, indicating that the server is operating beyond its optimal capacity for energy-efficient performance. This threshold is often determined through monitoring and analysis within the specific operational context of a data center. Moreover, the extra power consumption generates more heat, necessitating increased cooling efforts, which further escalates energy consumption [10].
A methodology to detect server overload is given by the Gradient Descent Regression (Gdr) model [31], which adjusts the CPU use level based on past utilization data. By examining historical CPU activity, this method continuously changes the threshold to determine when the machine is overloaded. When the CPU utilization surpasses this dynamically adjusted threshold, it is said to be experiencing server overload. This means that the server is working above its maximum capacity and is not able to effectively manage more work.
In general, a server overload happens when a server's resources run out and it is unable to process new requests. There are several reasons why server overload could occur. Operations frequently use excessive bandwidth, and at other times the system overuses RAM or operates out of CPU power [32]. From this, it is possible to say that a high level of operational data like CPU or RAM usage over time, shows the presence of server overload.

According to source [33], the overuse of one or more essential resources, such as the CPU, disk, or network interface, causes servers to become overloaded. When one or more resources are used excessively, it is known as overutilization and results in low server throughput and longer client response times. Overload refers to the situation where the number of requests exceeds the capacity of the system, resulting in increased response time or service denial, while overutilization (in *Figure 16* denoted as “OU”) occurs when one or more resources are used excessively, leading to low server throughput and increased response times experienced by the clients. In summary, overload refers to the situation while overutilization refers to the cause [33]. It is important to remember that a server hosts different virtual machines, which act as different server and each of them has its own utilization. So, the utilization coming from these affect the physical server utilization state [34].

Another possible situation, opposite to the previous is given by the idle server (fourth box of *Figure 15*).
As it is well known, data centers need to deploy multiple servers in order to handle the surge in cloud subscription requests. However, many of these servers are idle during regular working hours. Indeed, idle resources account for almost 60% to 70% of the total energy consumption [35][7]. A server is idle when it does not receive any workload for a while, but still consumes power since it is turned on and it is waiting for a future request.

Moreover, it has been noted that servers' average utilization rates rarely go above 50%, and occasionally they may even fall as low as 20% [35]. According to [10], usually, the data center utilization rate stays between 10% and 50%. This common scenario also presents a wastage of electrical power because servers are in a state of underload (in *Figure 16* denoted as “UU”), meaning that their capacity and computational resources are not fully adopted, and they can potentially accommodate more workloads. According to [10], several researchers studied how to detect an overloaded or underloaded server. For instance, Beloglazov and Buyya (2010a) tell to Identify these possible states of servers by setting upper and lower utilization thresholds; Beloglazov and Buyya (2010b) then suggest to use of statistical analysis of past virtual machine

utilization data and using it for the auto-adjustment of utilization thresholds, useful to detect underloaded servers. The same researchers in 2012 proposed a comparative analysis of relative host utilization to detect underloaded hosts within a data center.
Horri et al. (2014) suggest making a control on CPU utilization and the number of virtual machines on the host to find underloaded hosts; Lin et al. (2011) suggest calculating a lower threshold by considering the common and maximum workloads of virtual machines (VMs) in order to identify servers that are underloaded. Yang et al. (2014) identify underloaded servers and assess the operational state of physical machines using load ratios, notably the total used resource weight ratio.

According to [34], it is possible to assign six different thresholds to spot the server state, which divide resource utilization in five zones: very low, low, moderate, high, very high. These zones have been assigned the corresponding ranges of utilization: from 0 to 30%, from 30% to 50%, from 50% to 70%, from 70% to DY, and from DY to 100%. DY can be a utilization percentage of 90. In the following picture (*Figure 17*) there is the illustration of what just mentioned.

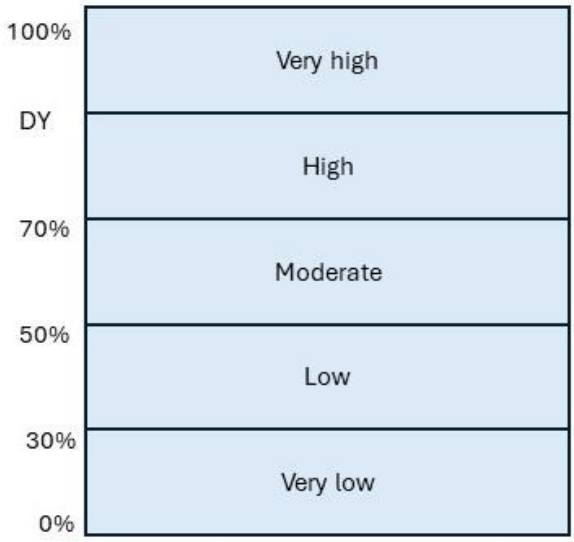


Figure 17. Allocation ranges of server utilization, deducted from the source [34]

According to this division of utilization and from previous analysis, it is possible to say that an idle server has a utilization rate lower than 10%, an underloaded server from 10 to 50%, an optimal utilization rate from 50% to 70%, an overutilized server from 70% to 90%, and an overloaded server from 90 % to 100%.
The simulation of this dynamic algorithm application has been demonstrated in the two fictitious use cases presented at the end of the framework explanation in section 5.

Another common hotspot is related to the DRAM (Dynamic Random Access Memory), which is a type of memory used in computers and other devices for storing data. DRAM is used in computing devices because of its high-speed ability, but it is known that it is an energy-intensive component. DRAM's architecture and design are the main causes of its energy inefficiency (this topic will be addressed later in section “Root Cause Analysis”). The density and cost-optimization of traditional DRAM architecture result in some innate inefficiencies, especially when it comes to the use of energy for access to information and retention [36].

If servers are seen to be in optimal condition and the high value of energy consumption is related to the DRAM, then the issue can be the DRAM itself.

Also, the number of active servers can create an energy intensive utilization [6].
Another source of energy consumption can be found in the data center network or in the manufacturing floor, such as from IoT devices or production machines.

Overall, the current stage of this procedure involves a thorough analysis meant to clarify the connection between operational metrics and patterns of energy consumption built in previous data monitoring acquisition phase in order to find what is the hotspot. This analysis carefully examines the operational data mentioned previously in the first step, such as CPU, RAM, disk, and network utilization, to localize and spot, looking at the correspondent energy consumption values, where this consumption comes from. The idea is that by figuring out these correlations, it is feasible to identify the specific system elements and conditions from which energy waste arrives, because they account for a disproportionate amount of energy consumption compared to the required and/or expected.

Drawing from the discussion done until now, for the sake of simplicity, I suggest utilizing a comparative approach to examine CPU utilization across time in order to evaluate the current state of server utilization. It is possible to classify the server's operational state, whether it is underutilized, overutilized, overloaded, or idle by defining specific ranges of CPU usage.
A CPU usage of less than 10% it may indicate an idle server, while from 10 can be a sign of underutilization and a sign that the server is not being used to its maximum capacity. An ideal usage range would be between 20% and 70%, which would show good server activity without taxing the system too much. On the other hand, CPU utilization above 70% may indicate overuse, which could result in decreased performance and a higher chance of hardware failure.
If instead the CPU value is often around 0 %, the hotspot is the idle server, while if it is often around 100%, the detection refers to an overload server.
According to source [1], it is possible to associate energy consumption levels of different servers to the corresponding server utilization level.

Under the case of optimal CPU usage, If the energy consumption is high, it is possible to allocate the issue to the DRAM. Because this type of memory is often used in computing system, it is possible that an higher energy cost is due to the DRAM itself.

In parallel, it’s possible to assess the overall effectiveness of our server deployment by looking at the number of active servers over time and their corresponding utilization rates. A pattern of underutilization that is consistent across several servers may indicate that the infrastructure is overbuilt for the workload at hand, offering a chance to lower the number of active servers during regular business hours in order to save money and energy.

Furthermore, it is possible to calculate the PUE value after the energy monitoring and quantify how much well our data center performs in terms of power consumption. Suppose there is an high PUE, for example around 1.8. In that case, it is possible to understand that much of the energy consumed does not come from IT operations data workload and transmission), but is mostly due to supporting structures of data centers like lighting and cooling. So, from the research shown in *Figure 15* [30], it is most probable that the power consumption comes from the cooling activity. It is possible to confirm this hypothesis by monitoring the servers'

temperature over time in order to have information about how well the cooling system is working and checking the server states to understand the cooling behavior. Temperature spikes on the server, especially after extended periods of high use, could indicate that the cooling system is having difficulty dissipating the excess heat produced. In order to prevent overheating, this calls for an examination of the server's operational state to find any areas that could use optimization. On the other hand, if servers consistently experience cooling problems even when they are idle or operating at a normal load, this indicates a need for corrective actions to implement a better and more efficient cooling mechanism.
It is important to hold an energy audit in which there is the identification of the energy consumption hotspots within the manufacturing plant by analyzing the energy consumption trend. This step involves the collection and analysis of data regarding the overall plant consumption, departmental consumption, and the efficiency of various areas within the plant. This step helps in understanding the plant's energy requirements and creating a map of energy efficiency across different sections [27]. According to the case provided in [27], the monitoring system is designed in a way to acquire data every 10 minutes throughout the day. This data is then stored in a database, allowing for detailed tracking and analysis of energy consumption patterns. The system collects 14 distinct signals corresponding to various departments and systems within the three main buildings, ensuring a comprehensive overview of the plant's energy usage. From this analysis it is possible to spot the intensive energy machinery.

4.3. Energy Inefficiency Allocation Activities

This part of the procedure consists of understanding how energy is allocated to distinct activities.
The core of this part is to understand the type of workload, the activity that is running during high and inefficient energy consumption. So, based on previous data gathering and relationship between energy and resource usage, now it is added the connection with the program and activity performed. By using some monitoring tools, it is feasible to observe the historical process information related to CPU, disk usage, and other critical system resources and link them on the programs/activity executed. By analyzing this data with the energy related data previously gathered, it becomes possible to discern which specific activities are responsible for higher consumption levels. These tools, designed to track and record system performance over time, provide useful insights into the patterns of resource utilization. By means of thorough analysis, it is possible to determine not just periods of high usage but also to associate these events with particular actions or tasks that are being carried out on the system during those periods. With this method, resource consumption can be understood at a finer level, allowing for focused optimizations to reduce excessive energy use. Through the identification of the most resource-demanding activities, strategies can be put in place to improve system performance overall, cut down on operating expenses, and possibly prolong hardware component lifespans by preventing needless strain.
One example of the software tool is Kubernetes, which can identify different information like the activity (PID), number of tasks executing, number of tasks waiting, etc [37]. This tool screen is shown in *Figure 18* with the purpose of having an understanding of how it works.

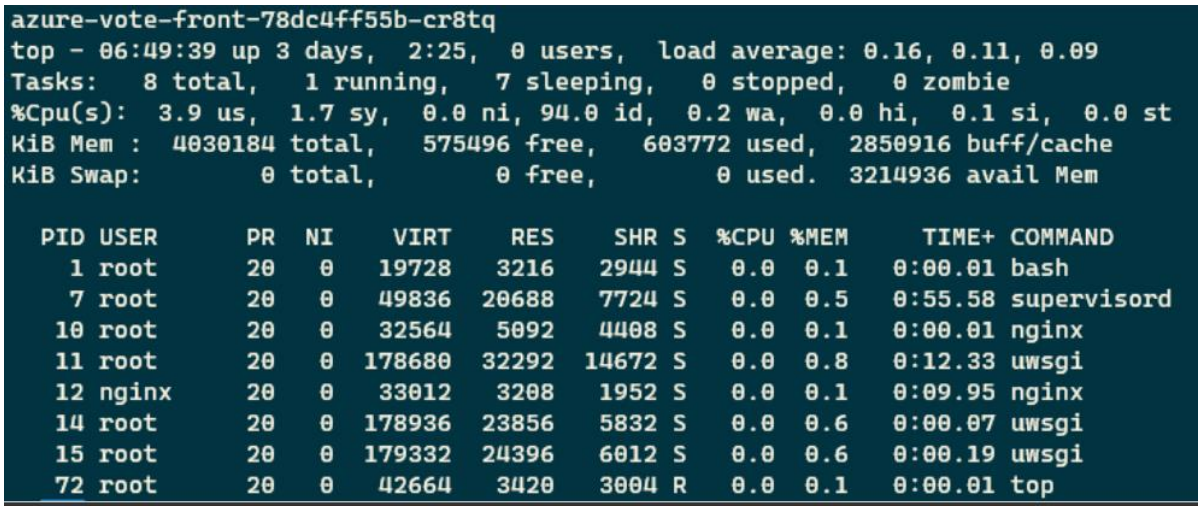


Figure 18. Illustration of Kubernetes functioning, adopted from the source [37]

So, for instance, it is possible to merge the knowledge from the Kubernetes tool and the knowledge from the methodology regarding the energy model previously shown in section 4.1, which is the "Adaptive Workload-Aware Power Consumption Measuring Method" [26](WSPM) for servers in cloud data centers. According to the systematic approach to associate energy consumption with server workloads provided by this approach, it is possible to establish a relationship between energy consumption, resource usage, type of workload and the activity and program used.

Regarding the DRAM, it is possible to identify which activities consume energy by using both hardware and software performance analysis tools. These tools include hardware performance monitors, specialized profiling instruments, and sophisticated modeling methods. Specifically, according to [38], different application tools like Intel Performance Counter Monitor (PCM), AMD uProf, and other tailored performance profiling tools are needed to pinpoint where energy inefficiency occurs. Using these types software, it is possible to allocate energy inefficiency to activities by collecting past data and analyze them, in this way is possible to understand in which moment a task was executing and linking it with the energy trend.
According to [36], another method to identify inefficiencies in DRAM is done through the use of a tool called DRAMPower, which can examine DRAM energy consumption patterns by taking into account different operational states (such as active, precharge, idle and refresh), as well as the changes between them. Some actions to identify inefficiencies consist of:

- Tracking Memory Access Patterns: This involves keeping an eye on when, how, and how often activations, precharges, and accesses to particular rows and columns occur in the memory[38].

- Analyzing Energy Consumption: Determine the amount of energy used for various operations and states by using the traced memory access patterns. This analysis aids in determining instances in which excessive energy consumption occurs, such as when keeping unoccupied cells or during idle times.

- Comparing Usage to Capacity: determine which portion of the DRAM's capacity is really being used by applications (to check for underprovisioning)[38]

- Analyzing Refresh Strategies: Looking for possible over-refreshing, which wastes energy, by evaluating the frequency and distribution of refresh operations[38].

DRAMPower overall provides the data needed to understand where and how energy is consumed within DRAM, which can be used to infer the presence of inefficiencies. It's a powerful tool to identify potential areas for energy efficiency improvements.

In general, using this type application monitoring tools and profiling tools suppose that with a profiling tool it is possible to understand more in detail which program is running and which hardware software is affected, linking this information to the energy consumption.
In the *Figure 19*, it is shown an overview of different software and programs used in cloud manufacturing and which type of data activities is required.

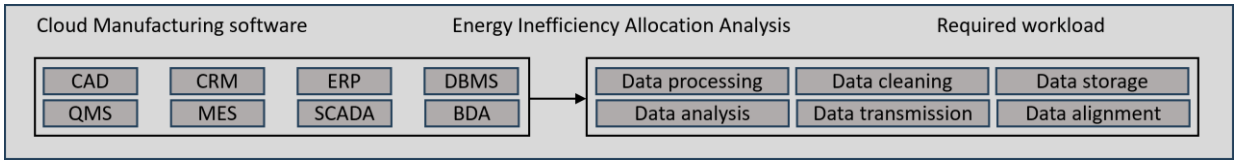


Figure 19. Energy Inefficiency Allocation Analysis phase of the Framework

According to [39], cloud manufacturing software run on the cloud (SaaS), offering new possibilities for value-added integration in the direction of co-creation and collaboration centered on platforms capable of synchronized and convergent operation and also reducing overhead costs [40].
Design professionals support their work with the use of computer-aided design software to document and design real-world objects. Computer Aided Design programs are used by businesses in the engineering, architectural, surveying, and construction sectors to represent models, planning, and design details of physical assets. CAD software comes in two popular general-purpose varieties: AutoCAD and MicroStation. ArcGIS Pro supports data from these types of programs [41]
Customer Relationship Management systems are needed to fill client data from a several sources, such as the business website, phone conversations, live chat, direct mail, promotional materials, and social media posts. By giving employees thorough knowledge of a customer's personal information, past purchases, preferences, etc, these platforms enable more individualized and knowledgeable customer care [40][42].
Enterprise Resource Planning system are employed to optimize fundamental organizational operations, such as supply chain management, manufacturing, finance, and human resources. Fundamentally, ERP makes it easier for these important processes to be organized and integrated seamlessly. Artificial intelligence (AI) and machine learning (ML) are two innovative technologies used in modern ERP solutions that are deployed via the cloud to improve automation, efficiency, and insight throughout the entire business. Additionally, by connecting an organization's internal activities with its global network of business contacts, these systems enhance collaboration and offer the adaptability required to thrive in the contemporary marketplace [43].
A Quality Management System (QMS) is an organized system used by businesses to supervise and carry out the processes and guidelines required to comply with quality standards and accomplish organizational goals. Companies can efficiently coordinate and manage their operations to meet their unique quality benchmarks by implementing a custom QMS, which guarantees uniformity and compliance throughout the activities [44].

A manufacturing execution system (MES) is a digital tool used to monitor and control the production process on the factory floor. It serves as an essential channel between the actual manufacturing processes and enterprise planning systems like ERP. Its primary responsibility is to continuously track and document the transformation of raw materials into finished goods, using information from machinery, sensors, and human input to provide precise and timely upgrades on production state [45].
Supervisory Control and Data Acquisition (SCADA) are industrial software designed to control and manage equipment. By integrating computer technology, SCADA allows operators to view, analyze, and control real-time operational data. This is useful to control equipment and timely provide alarms if there are any potentially dangerous circumstances [46].
A Database Management System (DBMS) is a type of specialized system software designed for database creation and administration. Users can create, secure, retrieve, edit, and remove data from a database with this system. Acting as an intermediary between the database and the users or applications that access it, the DBMS is essential to guaranteeing that data is consistently accessible and organized in an orderly manner. The DBMS includes a number of fundamental features in its structure, such as data organization, data access, modification, and secure locking, and database schema definition, which establishes the logical structure of the database [47].
This large data sets are consolidated into a structured repository on a big data platform, which located in the cloud. In order to do this software and hardware are used. A data analytics platformis an integrated collection of tools and services made specifically for the analysis of complex, dynamic, and vast amounts of data. It allows data to be recovered, combined, interacted with, explored, and visualized from a variety of sources that a company may own. A comprehensive platform for data analysis integrates multiple tools, each with distinct functionalities, such as content and natural language analysis, location intelligence, data visualization, and predictive analytics. The main goal of this platform is to convert all data types into insights that can be used into practice and produce measurable business outcomes [48].
All these activities imply the use and transfer of huge quantity of data in order to improve decision-making across multiple domains and hierarchical levels within a system, so it results necessary to gather big data and have the capacity to make analytics on them (Big Data Analytics) [48]. The gathering of information in manufacturing environment starts from sensors which then they route raw data. This is the first step in the general workflow within a decision-making structure, as shown in Figure 20 [49], and it is essential for the analytic purposes. Data acquisition in manufacturing is a very important step that allows for the collection and analysis of data throughout the lifecycle of a product. This process begins, as just mentioned, with gathering of raw data, which include various types of information such as temperature, vibration levels, and operational metrics from manufacturing machines like Computerized Numerical Control (CNC) machines, among others [50].

Initially, data needs to be cleaned to remove any inaccuracies, such as null values or outliers, which may deviate the analysis [50], in the scheme represented in *Figure 20* this step is called data validation. The next step is the preprocessing in which operation like data filtering is done to eliminate any irrelevant, unclear, or inaccurate information. After this preliminary screening, the data is processed to make sure it meets the specifications of computerized models and algorithms. To aid in choices, this prepared data is then examined using mathematical models and algorithms.
The representation of the big data analytics workflow is described below In *Figure 20* [49].

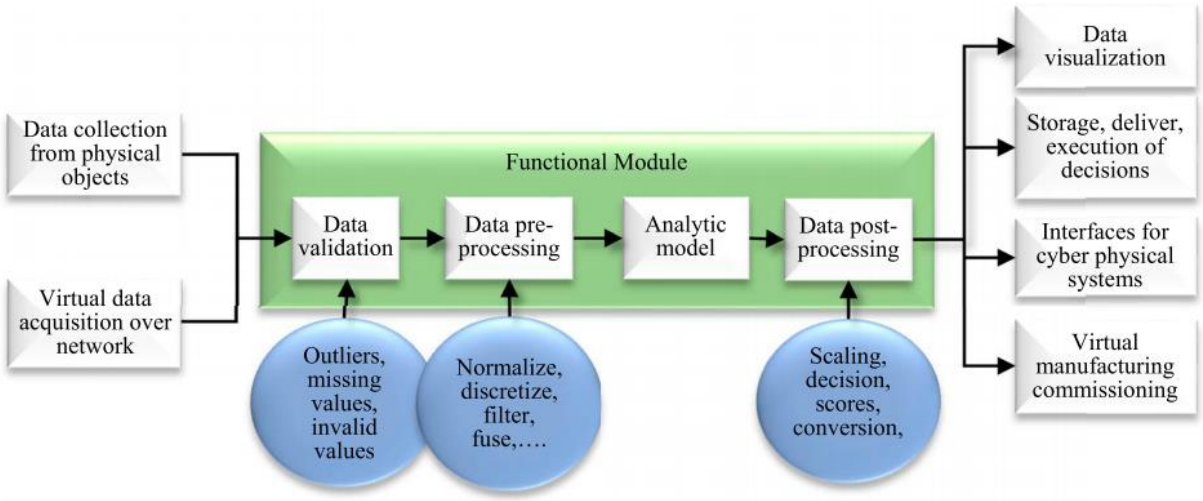


Figure 20. Big data analysis workflow, adopted from [49]

The first step involves employing technologies to facilitate the efficient and accurate collection of data. While manual data collection methods are still in use, they tend to produce data that can be inaccurate or delayed, leading to ineffective decision-making. To overcome these challenges, Internet of Things (IoT) technologies are increasingly being integrated into the manufacturing environment. These technologies enable the seamless gathering of data through smart devices and sensors embedded within the machinery and across the manufacturing floor [50].

For instance, in smart factories, Wireless Sensor Networks (WSNs) are used to collect, record, and keep track data in real time. Due to their affordability in executing industrial processes, technologies like Bluetooth, RFID, and ZigBee are common options for real-time data acquisition. By controlling machine status, Data Acquisition Devices (DAQs) provided with sensors such as transformers for current and cameras deliver non-intrusive methods of data collection. The manufacturing floor can be connected with these DAQs, and the data gathered can be organized and transmitted for analysis through a central gateway [50].

Moreover, data flow and management architectures are crucial for handling the large volumes and velocity of data generated. These architectures often utilize industrial communication protocols like OPC Unified Architecture (OPC-UA) and MQTT for data collection and transmission [50]. They ensure that data is collected efficiently, structured for analysis, and stored securely, often in flexible databases like NoSQL to accommodate the heterogeneity of the data.

Because real-world data is typically noisy, incomplete, and inconsistent, careful elaboration is necessary to improve the quality of modeling results [51].

One of the problems with raw data is given by missing values in input variables, which result from sensor readings or manufacturing process inspection items, as we will discuss in section 4.4.

Owing to the nature of production data, errors can occur, resulting in large volumes of missing data from a variety of sources, including sampling inspection, machine breakdowns, sensor malfunctions, and maintenance procedures. Since missing data frequently complicates the modeling process, data mining models must address missing values. When dealing with high amount of missing values, a bivariate approach is advised because it enables the exclusion of a small number of cases for efficient modeling and visualization [51].

Another crucial concern is outliers which are occurrences that may come from human error, contamination, process abnormalities, and sensor errors are some of the causes [51]. Certain anomalies could point to product malfunctions, requiring a review by domain experts to identify the problems and their fixes. In order to distinguish outliers from process drifts, which are alterations in data properties over time brought on by elements consisting of seasonality or specification changes, in the document [35] proposes a criteria-based method for defining outliers. It is possible to have outliers associated with failures, these may provide important information about the underlying causes, while those unrelated to failures could be eliminated [35].

Also, as mentioned previously, there may be data shifts related to the manufacturing process, which can be allocated to process drifts and seasonality. According to [51], in order to account for these changes in the characteristics of the data, is possible to create new variables that are informed by domain expertise and calibrated against these shifts. In this way, it's feasible for the modeling process to take into consideration how manufacturing data changes over time, guaranteeing that analyses stay accurate and consistent [51].
In simple words, process drift is a situation where the properties of production data change over time due to modifications in the manufacturing process, such as changes in control limits or target specifications after maintenance procedures or other specific events. This shift in data distribution makes the existing models less accurate in predicting the future behavior of the production process.
Seasonality instead refers to the periodical and cyclic change in production data that happens over time due to factors such as calendar effects, weather patterns, and demand cycles. These seasonal factors can significantly affect production data and can be accounted for by adjusting the data based on the time of the year or other seasonal factors [51].

Preprocessing is a critical step in data analysis that involves preparing raw data for further processing and analysis. This stage can include cleaning the data (previous stage) and it also often involves transforming the data, such as normalizing numerical values to a common scale, to make the data more suitable for analysis. By preprocessing data, analysts can ensure that the input data for machine learning models or other analytical tools is accurate, consistent, and ready for meaningful analysis [52]. During this step, normalization takes place to ensure consistency in measurement scales across different datasets. This normalization often uses methods like min-max normalization, and it typically involves adjusting the data values to a common (standard) scale with no distorting variances in the ranges of values [26].

The next step is data processing that use artificial intelligence and machine learning algorithms in order to produce a desired output. Different several factors let this step differ slightly across operations, such as the data source (data lakes, online databases, linked devices, etc) and the intended application of the output [53]. According to [49], the process of using mathematical models and algorithms to study data and capture insights to support decision-making is defined in the *Figure 20* as "Analytic Model". This stage is crucial for interpreting, analyzing, and producing insightful data. It follows the initial stages of data collection and verification, serving as a conduit to transform processed data into conclusions or knowledge applicable to practical contexts. This model facilitates an enterprise system's decision-making process by providing an ordered approach to data analysis.
In the post-processing step (last box of *Figure 20*), the choices or information obtained from the analytical model are used. At this stage, the processed data has been analyzed and

comprehended using the analytical model before being utilized for operational decisions or practical applications.
Human interaction may be necessary at this point, especially if the system isn't fully automated and graphical user interfaces, or GUIs, are required for users to view, examine, and understand the data. This ensures that the insights gained from the data analysis are effectively integrated into the operational processes, facilitating better decision-making and strategic planning within the organization [49].

4.4. Inefficiency Characterization

In this phase of the framework, the aim is to characterize some of the inefficiencies that may happen during the previous activities. In order to do this, it is necessary to make an analysis on production data. During this phase, it is possible to see the state and the condition of production data during different stages and make comparisons between them and check for patterns. In the case in the previous stage the identified activity was data transfer, it is possible to check and monitor the transfer time to check the latency, the quantity and size of data sent during time and monitor the bandwidth.
According to the source [28], in the case of manufacturing floor, to accurately identify operations leading to high energy consumption, the audit emphasizes the importance of integrating energy data with production data. This approach allows to make an analysis that considers the impact of specific production activities on energy usage, enabling the identification of inefficiencies depicted in operations.

In particular, eight inefficiency characterization have been diagnosed. Some of them are not mutually exclusive between each other, so that they can occur simultaneously. These box shown in *Figure 21* represent different types of problems which are listed as follows: redundant data, excessive data, dirty data, variability of workload, high latency, high variability of bandwidth usage, high frequency of requests and low frequency of requests.

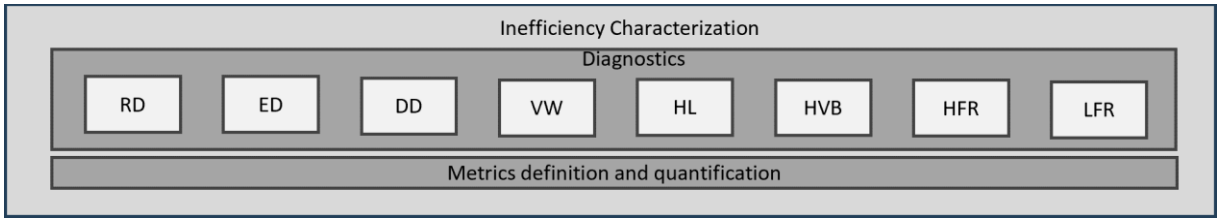


Figure 21. Inefficiency Characterization Overview phase of the Framework

Redundant data own multiple meanings within the context of technology and information management. In the setting of computer or network systems, it refers to the components that are strategically deployed alongside primary resources to provide a safety net in case of failure (as storing the same information in different cloud locations). These redundant systems are crucial to keep continuity and reliability in case some issues happen and there is demand for this information.
Another definition is given by unnecessary or repetitively duplicated information, which may not contribute additional value. This aspect highlights the importance of efficiency and relevance in data management because these data are stored several time occupying storage and not providing value [54].
In the domain of data transfer, it instead refers to redundant bits. This redundancy type is expressed as extra binary digits that are required in a data transfer to check and verify that no

information was lost during the process. This form of redundancy is needed to maintain data integrity and ensure accurate communication between devices [54].
Lastly, when it comes to safeguarding a storage array, redundant data is employed to protect against data loss in the event of a hard disk failure. This redundancy is essential for data recovery strategies, providing a layer of protection that minimizes the risk of losing valuable information [54].

Excessive data can be similar to redundant data, but it has wider meaning. In fact, it refers to the amount of data gathered from the manufacturing floor and transmitted to information systems that is too much respect to the one needed. Indeed, according to [55], big data does not always mean to add insightful or relevant information for machine learning models. Indeed, large datasets can lead to unnecessary computational operations and consequently energy costs. It is possible to say it because the experiments done in this study [55] showed that with a significant reduction in sample size used in the machine learning algorithm (data processing), up to randomly discarding around 75% of the samples, does not drastically affect the accuracy of classifiers used in the study. This indicates a clear redundancy in the quantity of information in most datasets.

Dirty data refers to information within a database or dataset that has lost critical attributes, rendering it inadequate for its initial intended use by the organization. Such attributes include accuracy, accessibility, and completeness, among others. The degradation of data quality can occur at any stage of the data handling process. Additionally, the issues of dirty data are exacerbated by untimely and inconsistent information flow within an organization. This problem is particularly pronounced during the integration of data from multiple systems that operate under different standards. An example of this is when one system treats names as a single field while another separates them into first and last names; this discrepancy leads to data inconsistencies, where one format is deemed valid and the other requires data cleansing to meet organizational standards [56]. The recurrent process of adjusting and correcting data increase processing time and so energy usage.

“Slowdown variability”, according to [57], in computing systems refers to the fluctuation in job completion times relative to their optimal completion time under ideal, non-competitive resource conditions. This variability arises due to the heterogeneous nature of jobs, differing in size and complexity, and their competition for shared resources like CPU, memory, and network bandwidth. Additionally, scheduling policies that do not adequately differentiate between job sizes and types contribute to this issue by treating dissimilar jobs similarly, leading to inefficient resource allocation and utilization. The manifestation of this variability is observed as unpredictable and often increased job completion times, especially for smaller jobs that could otherwise be completed quickly but are delayed by larger, resource-intensive jobs. The implications of high slowdown variability are significant, leading to decreased system performance, lowered efficiency, and diminished user satisfaction. It complicates resource management and scheduling decisions, making it challenging to optimize for throughput, fairness, and efficient resource use [57].

The delay in network communication is known as network latency. It describes the time required for data to move over a network. High-latency networks are those with longer delays, while low latency networks are quickly. High network latency results in slower application

performance and high failure rates [58], implying more time to do the data transfer and more energy associated to it.

The quantity of data that is possible to send in a period of time through the network connection is known as bandwidth. It is measured as bits per seconds (Bps) [59]. A network with higher bandwidth is able to transfer the same amount of data faster respect to a network with lower bandwidth.

Frequency of requests is the amount of requests per period of time that are demanded in cloud manufacturing operations and it is highlighted as the frequency. An high demand of service regarding a server can represent an inefficiency and the same applies if a server receive only a few requests.

These inefficiencies can be measured and counted in order to have metrics to understand the severity of these inefficiencies.

According to [60], manufacturing industries find themselves in a challenging position due to several obstacles in the digital process. Key challenges include the cultural and organizational divide between Information Technology (IT) and Operational Technology (OT) teams (this will be explored in the next chapter), different priorities and languages, which can lead to misalignment of objectives and hinder collaborative efforts. Additionally, there's a lack of standardized systems and processes across the manufacturing landscape, making it difficult to achieve seamless integration and interoperability. This results in operational inefficiencies and a slower response to market demands, ultimately impacting the effectiveness of digital transformation initiatives within the sector.
According to the study done in [60], industries found out several issues in handling data. This will be shown in the following table (*Table 3*), presenting the obstacle and the number of companies facing this issue (%).
In the following table (*Table 3*) of the next page some statistics regarding the difficulty in handling data is shown, information taken from the source [60].

Table 3. Statistics on data handling-related issues, adapted from [60]

Difficulty in handling data	**Number of companies facing the issue (%)**
Lack in accuracy of data	40
Lack of clean data	38
Insufficient knowledge to comprehend and work with IT data	35
Incapacity to adapt data models to new requirements	34
Incapacity of incorporating data from several systems	34
Inadequate data management organize between the IT and OT departments	33
Utilizing a small portion of the available data	33
Absence of useful data from outdated OT devices	33
Insufficient knowledge to comprehend and work with OT data	32
Not having the necessary data available when needed	31
Incapacity to obtain sufficient data quickly	31
Challenges in exploring and discovering data	29
Incapacity to remove data from OT equipment for usage in another location	28

These obstacles are indicative of the inefficiencies outlined in the framework, particularly in terms of data management challenges such as dirty data, redundant data, and excessive data. They highlight the issue of workload variability, underscored by the difficulties in timely information access. The inability to promptly receive necessary information or delays can often be attributed to computing operational bottlenecks, further worsening the inefficiency issues within the digital transformation journey of manufacturing industries.

4.4.1. Metrics definition and quantification

It is possible to define metrics for the above-mentioned inefficiency characterization states in order to quantify them.
Indeed, there is a method for quantifying the similarity between two sets of variables, which refers to redundancy. This method employs sophisticated mathematical techniques in order to compute this redundancy index, which is a value that ranges between 0 to 1and, which indicates the degree of similarity. A value nearing 1 suggests a high similarity between the sets, whereas a value approaching 0 indicates significant differences [61].
Regarding dirty data, the metrics used to quantify them are structured into three levels: generic, contextual, and comparative. Generic metrics are universal checks like data type/schema, missing values, and outliers. Contextual metrics are specific to the dataset and use cases, focusing on the relationships between columns and expected value ranges. Comparative metrics involve comparing the data against a master dataset to ensure accuracy and

consistency. These metrics guide the assessment and cleaning of data to improve its quality [62].
Response time is the amount of time that any workload takes to submit a request for work to the virtual environment and wait for the virtual environment to finish it [63]. Latency is also a metric, and it is described as the amount of time that passes between sending a packet and receiving it at its destination [63]. The difference between these two in that the first includes also the processing time from the server. Longer response times may be the result of a higher request rate per second, which suggests that the system is under more stress [64].
The term "throughput" (bandwidth) describes how many tasks a computing service or device can complete in each amount of time. Transactions-per-second is the standard unit of measurement for transaction processing systems[63] . In order to compute variability in the bandwidth usage it is possible to use statistics measure like the standard deviation, which quantifies the average amount of variability or dispersion in a dataset, in this case applied to bandwidth data.

4.5. Root Cause Analysis

This phase of the procedure has the objective to understand which are the primary causes of energy consumption by understanding the root of the inefficiencies leading to energy wastage. There are several methods to make a root cause analysis, such as the five whys, cause effect approach, statistical process charts, and others.
The five guys technique is a simple but effective methodology to uncover issues' cause, and its application focuses on asking "why" starting from the problem statement until the root cause is identified. So, the process works in this way [65]:

- Problem Statement: the definition of the problem is explicated.
- Ask "Why Did This Happen?": the process starts by asking the first why regarding the problem. This involves looking at the immediate causes and writing them down. It's a step-by-step process where each answer forms the basis of the next question.
- Check if This Is a Root Cause: After some "Why?" questions, there is the assessment to understand if the cause identified is the root cause of the problem. If it's not, it means you've likely identified another symptom or contributing factor, and you should return to step 2 to delve deeper.
- Repeat Steps 2 and 3: Continue asking "Why?" and assessing the answers until you reach the root cause of the problem. The goal is to continue until you uncover the fundamental cause.

Another way is the cause effect approach which is an effective tool for organizing information from various sources and is crucial when dealing with complex problems. It creates a logical flow showing the relationship between the different causes with the effect [66].
Using one or all of these methods, it is possible to deep dive in the original cause of the inefficiency with the aim to then address it with a dedicated solution.
In the following picture, there is the representation of this step of the framework, in which all root causes identified are shown (*Figure 22*).

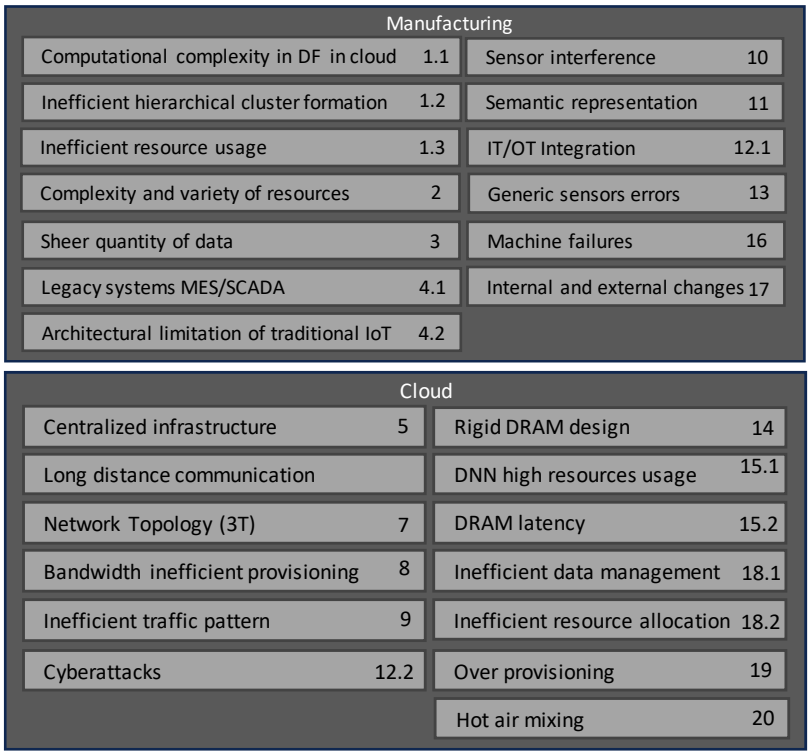


Figure 22. Framework section: Root Cause Analysis, divided in two boxes based on where the issue occurred.

As mentioned in previously chapters, data preprocessing is an important step in cloud manufacturing activities, that help to lead and achieve better insights and results. In order to do the processing phase, the data fusion process takes place. This step usually has been done at server cloud level or edge-fog layer, depending on the cloud architecture adopted, which are presented in chapter 3.5. If it is done in the cloud server level, the cloud server heavily retains this algorithm.
Indeed, this process is very complex because it agglomerates data coming from different sensors which aims at improving the accuracy or reliability of information. This can involve complex algorithms to filter, average, or merge the data, which requires high computational power [50]. In this case, data fusion is a very complex algorithm and as such it requires high computational operations, demanding high amount of energy.

Data fusion algorithm in cloud servers receive data coming from, in common manufacturing environments, the local Wireless Sensor Nodes (WSN) which is distributed in a logical hierarchical clusters. This logical setting on sensor nodes has been done to reduce data redundancy [67]. If this agglomeration technique of sensors (clustering) is not done well, the network will have more redundant data from these sensors [50].
This represents an inefficiency that affects both the data transfer and the successive operations that data need to receive.
A methodology to find this issue is to monitor when data transfers are done by sensors and check the data that has been sent. If data redundancy is noticed, it means there is a necessity for optimization.
Another issue concerning wireless sensor nodes is their resource usage: components remain active even when they are not used, increasing the number of sensors utilized for data transmission, increasing their energy, and reducing their lifetime.

Starting from the case of traditional cloud manufacturing infrastructure, that is the one without edge layer, there are several critical challenges, mostly concerning the handling and processing of data. Efficiently collecting data from a diverse range of manufacturing resources is one of the main challenges. Since these resources are dispersed over multiple locations and show diverse operational dynamics and sensor outputs, gathering data becomes a challenging task [42]. If the data collection process is not optimized, it may lead to the collection of dirty or redundant data, which requires additional computing work and wastes energy.

The work of data integration and transmission further complicates the picture. The sheer amount of data generated by these manufacturing resources is frequently too much for the infrastructure that is currently in place to handle. There are major obstacles in the way of achieving seamless data flow because of the bottlenecks created, which lead to inefficiencies in the management of data and its subsequent integration into cloud platforms [68].
The inefficiency in managing data flow can lead to servers and network equipment working harder and longer than necessary, thus consuming more energy.
Another issue related with the huge amount of data regards the fact that all this information is actually not very useful. As already mentioned in the section “Inefficiency Characterization” of this chapter, machine learning algorithms work very well even without a big portion of these data. For this reason, it is possible to say that many of these operation in transferring and analyzing data represents a waste in terms of time and energy resources [55].

Another point of concern is about the capability for real-time monitoring. Many traditional systems are insufficient in offering instantaneous feedback on the operational status of manufacturing resources. The gap in real-time data processing and analysis impacts critical decision-making processes and the optimization of resources. This gap is due to the traditional systems, which include legacy SCADA and MES, and also due to the architectural limitations of Internet of Things (IoT) technologies, making real-time oversight challenging [68]. MES and SCADA are effective when dealing with data storage, transmission, and querying from the physical workshop to the cloud in a centralized manner, but they have issues with real-time data query requirements of cloud applications. This can lead to poor performance in task scheduling, difficulties in quickly rescheduling manufacturing resources under abnormal conditions, and a significant overload on the cloud platform’s computing workload, resulting in slow response speeds [68].

The CM process explored is the centralized cloud-based access operations, that means the production data move always through a centralized process for storing, transmitting, and requesting the equipment data from the physical workshop and this process is inefficient.
The second cause related to scarce capability of real time data connectivity is due to the traditional IoT architectures, which despite they have permitted the connectivity and data transmission from manufacturing equipment to cloud platforms, they are not optimized for the high-volume, real-time data processing required in cloud manufacturing. These architectures often lead to inefficiencies in data transmission, storage, and analysis due to their centralized nature, which can overwhelm the cloud platform and create obstacles in real-time and responsive decision-making [68].
Indeed, as just described, they are not capable to guarantee real-time data requirements of cloud applications. For instance, order completion time prediction and equipment fault diagnosis may suffer from delays due to the inefficiencies in these methods.

This could cause serious overload on the cloud platform's computing workload because of slow response speeds of cloud applications to handle unforeseen events, poor performance of task scheduling, and low adapting capabilities.

Therefore, these methods contribute to inefficiencies in the context of cloud manufacturing systems, as they fail to meet the demands for real-time performance and efficient computational load distribution between the cloud and manufacturing layer [68] This means it can't promptly identify when something goes wrong or when there's a deviation from normal operations like if there are changes or anomalies in the operational environment. This delay in detection means that the system continues to operate under less-than-ideal conditions without immediate correction, leading to unnecessary or inefficient energy consumption.

Keeping the case of the traditional cloud manufacturing environment, according to [69], other issues emerge are the following:

- High latency is an important problem. Latency stems from the requirement to transfer massive volumes of data from manufacturing devices and sensors to remote cloud data centers for processing. The primary cause of high latency is the physical long-distance data must travel between the manufacturing floor and the cloud data centers.
  Indeed, delays come from the travels back and forth data have to do between the IoT nodes and the cloud for processing. This round-trip delay is particularly problematic for real-time applications, where rapidly data processing and answers are essential for operation. As a matter of fact, cloud manufacturing processes need for decisions and action to be taken and executed promptly, so any delay in data processing can lead to inefficiencies in the manufacturing floor. Strictly related to the long distance is the geographical system distribution even with edge fog layer. Indeed, another issue creating latency is the reliance on some specific data centers, so even if some edges are closer to the manufacturing floor, some information are kept in another faraway location [70].
  The consequences of high latency and delays can have an impact on the energy consumption in manufacturing floor such as, for example, prolonged machinery runtime and the higher time of transmission in the network also lead to more energy usage.

- The substantial consumption of bandwidth. This is due to the continuous transfer of large volumes of data from numerous sensors and devices across the manufacturing process to the cloud for analysis and storage.
  The cause of this bandwidth consumption is the reliance on centralized cloud services for processing all data generated in the manufacturing process. With the increasing number of IoT devices in industrial environments, the volume of data needing transmission exacerbates the demand for bandwidth (bottleneck).
  The energy implications of high bandwidth consumption are multifaceted. Firstly, the infrastructure required to support such data transfers, including data centers and network devices, consumes significant amounts of energy for operation and cooling. Secondly, the process of transferring data itself requires energy, and as the volume of data increases, so does the total energy consumed. This not only affects the operational costs from an energy perspective but also increases the carbon footprint of cloud manufacturing operations [69].

- Security Risks: The transmission of massive volumes of data, often containing sensitive information, across networks to the cloud introduces significant security risks. These risks include potential data breaches and unauthorized access during the data's transit. The lack of robust encryption and security measures for data in transit can expose it to interception and misuse. Given the sensitive nature of some IoT applications (monitoring, industrial control systems), the implications of such security breaches can be far-reaching, affecting not just the privacy but also the safety of individuals and the integrity of critical infrastructure [70]. This issue is mentioned because as the risk increases, the probability of having malicious process states also increases. Having cyberattacks increases the bottleneck and storage used in cloud servers, leading to higher energy consumption. This will be highlighted later in this section.

Another root for inefficiency is found in the topology of the network which can significantly influence energy consumption and efficiency in data centers. Indeed, there is a type of network distribution, that is the traditional three-tier (3T) topology, which can introduce inefficiencies due to their hierarchical structure, which might lead to bottlenecks, underutilized resources, and consequently, energy wastage. These inefficiencies arise from the rigid separation of the network into distinct layers, which can limit the bandwidth available for communication between servers and result in longer paths for data to travel, consuming more energy [71]. Inefficient routing in a poorly designed topology leads to increased power usage by network devices as they handle traffic.
This type of topology is illustrated in *Figure 23*.

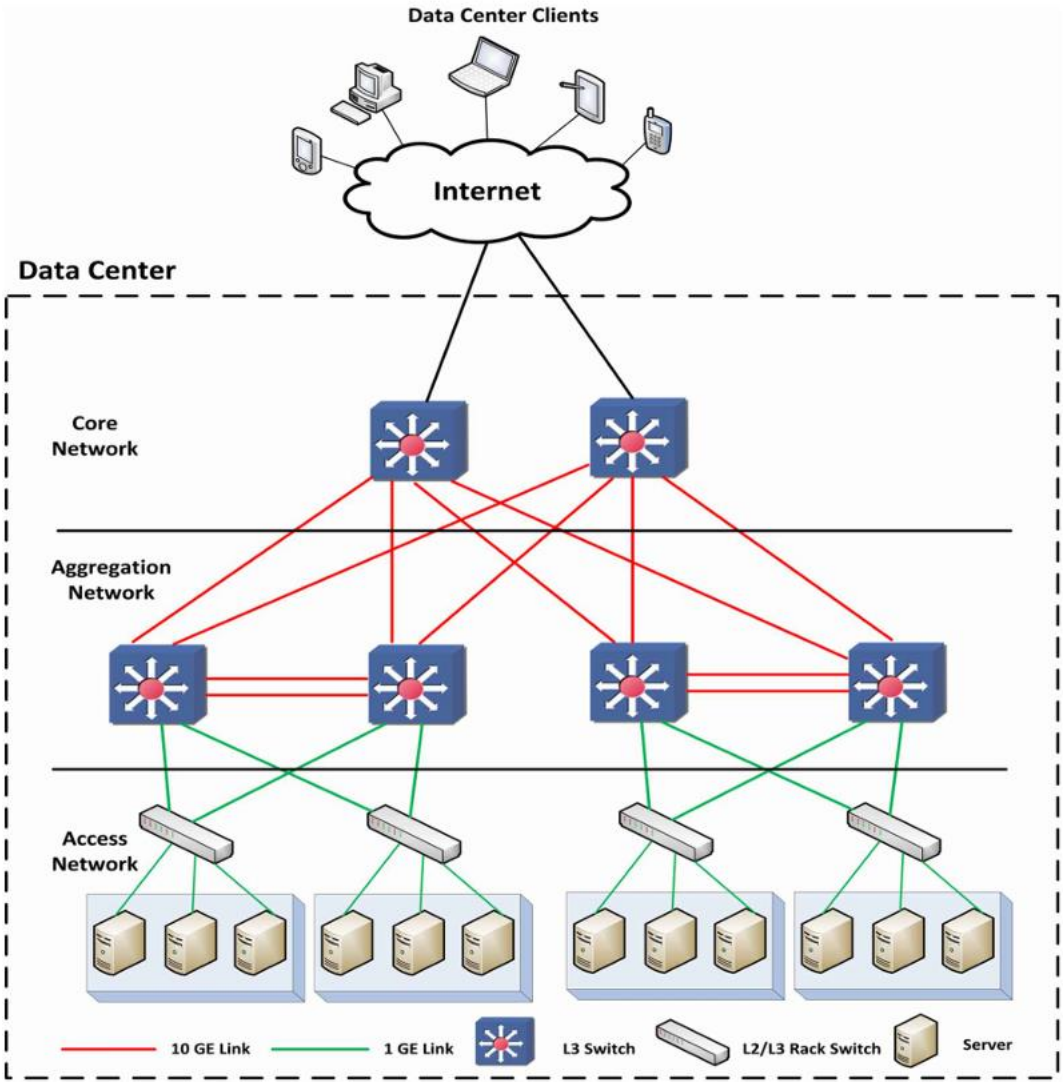


Figure 23. Three-Tier Network Topology, adapted from [72]

Another issue is around the bandwidth usage. Bandwidth provisioning involves allocating the appropriate level of network capacity to efficiently manage data traffic. When too much bandwidth is allocated (overprovisioning), it can lead to network segments that are underutilized, which wastes energy in a manner similar to when a low loaded server still consumes a substantial portion of its power. In the same way, allocating too little bandwidth (underprovisioning) can cause network congestion. This congestion forces data into queue and to travel with longer and less efficient paths, thereby increasing both the time and energy needed for data transmission. Achieving a balance by aligning bandwidth with the actual data requirements is key to optimizing energy consumption, ensuring that network components are effectively used without being left idle or becoming overloaded [71].

The transfer of vast amounts of data generated by IoT devices to the cloud affects this problem by asking a high demand of data processing which increases the bandwidth request. This is particularly problematic due to the continuous and simultaneous operation of IoT devices, which generate data at a rapid frequency. The need for high bandwidth becomes a critical issue as it not only increases operational costs significantly but also strains the available network infrastructure. This strain leads to higher expenses for organizations due to the need for upgraded infrastructure and potentially higher service costs to accommodate the extensive data transfer requirements [70]. The high bandwidth usage leads to higher energy consumption

due to the increased workload on network devices (e.g., routers, switches) and data centers to handle the traffic.

Regarding the network inefficiencies, linked to what is told until now, it is possible also to define the throughput, that is the rate at which data is processed or transmitted within a period, and the traffic patterns, which outline how data circulates across the network. These two factor are important when dealing with network efficiency. High demands for throughput and inefficient traffic routes can overburden network resources, necessitating equipment to function at, or close to, full capacity to accommodate the data load, thus elevating power consumption.
When networks have been assigned with fast processing or transmitting large volumes of data, it's essential for data distribution and flow to be optimized. Without efficient traffic patterns, data may congest specific routes, overwhelming some network paths while leaving others underused. By improving throughput handling and traffic flow, networks can avoid excessive strain on pathways and devices, leading to reduced energy consumption.
Moreover, network congestion, often stemming from the sheer of data heading to cloud centers for processing, further exacerbates these challenges. Existing network infrastructures may fall short of efficiently managing these large data volumes, causing delays, increased latency, and a drop in system performance. Such congestion not only increases operational costs and energy use but also affects real-time data-dependent applications, highlighting the necessity for enhanced network capacity and data flow optimization [70]. As it is possible to see, issues can arise from more root causes together because the cloud manufacturing infrastructure presents inherent elements working together.

Looking at the manufacturing floor, there is another important issue causing inefficiencies in data workloads: the sensor interference. This phenomenon happens frequently in cloud-based Internet of Things (IoT) networks, and then also in cloud manufacturing. The primary cause of this interference is the crowded radio frequency environment, in which several devices use the same frequency bands, resulting in signal quality degradation and collisions. As per reference [73], congestion is caused by sensor interference on shared frequencies, especially in the widely used 2.4 GHz band. In addition to limiting available bandwidth and causing packet and connection losses, this congestion also worsens the quality of communication channels. It is possible to say that sensor interference has a not negligible negative influence on data integrity, network dependability, and overall performance. This root cause has been identified in the framework in the following way.

The main cause of the issue stems from IoT devices' dual function as transmitters and information routers, leaving them vulnerable to possible interference from nearby devices. This interference significantly reduces the energy transmission efficiency throughout the network. The implications are extensive for cloud-based IoT networks, where timely transmission and data integrity are critical [73]. First, interference can corrupt data packets and make the information gathered and sent by sensors unreliable. Second, there is a compromise in the network's operational efficiency. Interference-induced retransmissions increase energy consumption and shorten the battery life of Internet of Things (IoT) devices. This has an effect on the sustainability of IoT solutions, particularly those installed in difficult-to-reach locations, in addition to increasing operating costs because batteries need to be charged or replaced more frequently. Interference has a negative impact on crucial Quality of Service metrics such as the transmission of packets percentage and end-to-end delay, which results in a poor user

experience. Moreover, the effectiveness and reliability of services that based on steady and reliable data feeds, like environmental control and smart infrastructure management, may be severely subjected to obstacles due to interference [73]. In addition, sensor interference poses a challenge to the scalability of IoT networks. The possibility of interference rises with network size, making it more challenging to maintain dependability and performance. This scalability problem may impede the growth of Internet of Things applications, making it more difficult for them to adopt new technologies and adjust to increasing demands.

Overall, it is possible to say that sensor interference significantly impacts data accuracy, network throughput, energy consumption, quality of service, and scalability in cloud-based IoT networks. These effects put in evidence the necessity of efficient interference mitigation techniques for ensuring the successful operations done by IoT networks, which is useful to fully utilize IoT in a variety of applications [73].

Another issue predominant in cloud manufacturing context is related to the semantic representation, which describes the method of organizing, characterizing, and presenting the information and knowledge of the production process tasks in a form that is both machine- and human-readable and structured. Structured representation allows a common and shared understanding of the knowledge. Semantic representation entails the modeling of manufacturing processes, resources, and products, so it needs to be well established in order to guarantee interoperability and efficient communication within cloud manufacturing systems [74]. In order to record and represent data related to manufacturing tasks, different methods are used such as tables, diagrams, and coding languages; each of these are applied in different way because they are created from different people[74]. The different styles applied bring in place different terms which are utilized to denote the same thing, while also the same word is employed to represent different things [74]. This may end up in an abundance of isolated, redundant, and inconsistent information representations, as well as potentially serious interoperability issues. This means that neither the uniformity of data not even the semantic correlation between the manufacturing tasks are guaranteed. Establishing a syntactically common and semantically in line manufacturing task description model is necessary [74].
To deep dive in the semantic representations inefficiencies, here are described some reasons [74]:

- Heterogeneity and Semantics: The challenge arises from the varied ways manufacturing tasks are conceptualized and documented across different systems, leading to inconsistencies in semantic representations. This variance makes it difficult to achieve a seamless exchange of information and interoperability among diverse manufacturing systems.

- Dynamics and Uncertainty: Manufacturing tasks are subject to change due to varying market demands, customer requirements, and the innovation of manufacturing technologies. This dynamism introduces uncertainties in maintaining the accuracy and relevance of semantic representations over time.

- Complexity in Matching Tasks with Services: Identifying and aligning manufacturing tasks with the most suitable services or resources is complicated by the broad range of manufacturing activities and their specific requirements. Effective matching requires

sophisticated semantic models that can accurately represent and interpret the needs of different tasks.

- Lack of a Unified Semantic Model: The absence of a standardized semantic framework for modeling manufacturing tasks hampers the ability to efficiently share information and collaborate among stakeholders in the cloud manufacturing environment. A unified model is necessary for permit semantic consistency and facilitating the integration of various manufacturing services and resources.

These issues underscore the necessity for advanced semantic modeling techniques and frameworks that can overcome the heterogeneity, manage the dynamics and complexity, and provide a unified approach for effective semantic representation in cloud manufacturing.

The semantic representation issues affect mainly three areas in data cloud computation:

- Increasing Complexity: The inherent heterogeneity and dynamic nature of manufacturing tasks introduce significant complexity into data processing and management. This complexity arises from the need to accurately capture, represent, and interpret the vast array of manufacturing tasks, each with unique characteristics and requirements. In order to have more efficient data management and computation, it is necessary to address these complexities with advanced semantic models to ensure data is efficiently managed and computed [74].

- Affecting Interoperability: The absence of unified semantic models severely hampers interoperability across different systems and services within the CMfg ecosystem. Interoperability is critical for seamless integration and communication between disparate systems, services, and components in the manufacturing landscape. The challenges faced in achieving interoperability due to inconsistent and non-standardized semantic representations complicate data exchange and integration processes. It is therefore necessary to reach an high level of interoperability which needs the development and adoption of standardized semantic frameworks that can accurately represent manufacturing tasks and enable efficient information exchange and collaboration across different platforms and systems [74].

- Impeding Real-time Processing: The complexities associated with non-standardized semantic representations and the dynamic nature of manufacturing tasks hinder the ability to process data in real-time effectively. Real-time data processing and decision-making are crucial for responsive and adaptive manufacturing operations, enabling timely adjustments to manufacturing processes in response to changing conditions and requirements. The challenges in semantic representation affect the capability of cloud manufacturing systems to support real-time data processing and decision-making [74].

Addressing the heterogeneity, dynamics, and complexity of manufacturing tasks requires sophisticated algorithms and models, leading to increased computational overhead and, consequently, higher energy consumption.
According to [74], it is really important to improve semantic representation in order to advance the capabilities of cloud manufacturing systems, enabling them to support complex, dynamic, and interoperable manufacturing operations effectively.

Different research have studied another issue in this system concerning the integration with Information Technology (IT) and Operational Technology (OT). The first one (IT) focuses on the management of information and data, and it includes the systems and technologies used for creating, storing, recovery, and transferring information across different environments. Examples of IT in manufacturing include enterprise resource planning (ERP) systems, data analytics platforms, and other software applications that process data for business decision-making, communication, and administrative functions.
The second one, OT, involves the direct monitoring and control of physical devices and processes in the manufacturing environment. OT systems are dedicated to managing and controlling manufacturing equipment, production lines, and industrial environments. Examples include programmable logic controllers (PLCs), industrial control systems (ICS), sensors, and actuators.
Integrating IT and OT in cloud manufacturing aims to create a seamless environment where data can flow freely between the information management systems (IT) and the operational machinery and processes (OT). This integration facilitates real-time data analysis, enhances operational efficiency, and supports more informed decision-making by leveraging the strengths of both domains [75]
Automation and control systems, such as SCADA (Supervisory Control and Data Acquisition), DCS (Distributed Control Systems) are often referred to as Operational Technology (OT) [46].
In the context of cloud manufacturing, Information Technology (IT) and Operational Technology (OT) are two separate domains that handle different aspects of the manufacturing environment, each with its own unique functionalities, technologies, and objectives [75]. For example, according to [76], data captured by traditional SCADA systems are complex to be exported and analyzed out of the SCADA itself.
There are then different issues related to IT/OT Integration that significantly affect data processing and data transfer in cloud manufacturing due to several factors [75]:

- Long Lifecycles of OT Devices: OT systems are difficult to update and integrate with new IT technologies, and this has consequences on the efficiency and timeliness of data processing and transfer. The problem is due to the fact that OT devices are not designed with frequent updates, integrating them with cloud manufacturing systems that rely on rapid data exchange and processing can be challenging [75].

- Compatibility: Due to the different lifecycles and technologies used in IT and OT, there can be compatibility issues that affect data transfer protocols and data processing standards. This disparity can lead to inefficiencies and errors in data exchange, requiring additional layers of data translation or adaptation, which slow down the overall process [45]. The integration of IT and OT systems aims to bridge the gap between data-centric IT environments and the operational, process-focused OT environments. This leads to a significant increase in system complexity due to the need to align two distinctively different systems. Managing this complexity is not easy because it arises from the integration of heterogeneous systems with differing requirements, protocols, and standards [77] .

- Applying Theoretical Concepts Practically is an obstacle, it is hard to adapt theoretical models, such as RAMI4.0 (Reference Architectural Model Industry 4.0), to actual IT-OT integration projects. While these models provide a structured approach to design and implement Industry 4.0-compliant systems, their application in real-world scenarios is

often challenging. This issue arises from the complexity of translating high-level architectural models into practical solutions that can be implemented in the diverse environments present in real industrial settings [77].

- Proprietary Solution Limitations: The proprietary nature of the original MES solution makes it hard to achieve seamless data flow and integration with other components of the IT and OT systems within the environment [78].

- Increased Cybersecurity Risks: Integrating IT systems with traditionally isolated OT systems opens up new possibilities for cyberattacks, potentially compromising the confidentiality, integrity, and availability of data. Ensuring secure data processing and transfer in this integrated environment requires advanced security measures [75] [79] [77].

According to [79], several operational consequences in manufacturing floor are coming from a cyber-attack, for instance:

- An interruption in the data being sent from a Remote Terminal Unit (RTU) to an OT system. Since the information could come from actuators, level sensors, alarm sensors, or turbine speed, this could ultimately result in a catastrophic event.
- Stopping a SCADA Master connection to initiate an event via a safety system.
- Making modifications to the values obtained from a SCADA Master. This could trigger an automatic reaction, such as shutting down a plant's portion, or it could trigger a human reaction that might result in an improper action.
- Alter or modify how equipment protection systems function, such as by increasing the speed of a plant's turbine, which destroys the blades [45].

The integration of IT and OT systems in cloud manufacturing introduces several issues that can impact data processing and data transfer, including challenges related to system compatibility, reliability, security, and the adoption of new technologies. These challenges necessitate careful planning and the implementation of advanced solutions to ensure efficient and secure data handling in cloud manufacturing environments.

As hinted previously in this section, some manufacturing events like sampling inspection practices, errors from sensors, breakdowns and failures of machines, and maintenance events can create misleading data that need more processing and analysis.

Sensors can lead to patchy and erroneous data, which do not accurately reflect on-site situations, that is because of the nature of generic sensors, Indeed the fundamental issue is given by the fact that generic sensors are designed to catch a wide range of data without being specifically adapted to the details of the particular manufacturing environment or process they are monitoring [18].
In manufacturing environment floor, there are two main ways to make measurement. One is direct measurement which involves the use of precisely designed sensors to either measure a physical property or something closely related throughout the manufacturing cycle. Although this method provides highly accurate data, it can be costly and needs to be customized for specific environments. An alternative approach is the indirect monitoring, which uses general sensors like current sensors and accelerometers to measure related properties through indirect

relationships. However, this method doesn't target the physical property directly but a related metric, which can complicate accurately identifying the intended measurement. For example, a high spindle current might indicate either a strong cutting force in metalworking or simply an increasing spindle speed, requiring extra context for accurate interpretation. The complexity inherent in manufacturing processes poses a significant challenge for indirect monitoring. Generic sensors, with their simplifications and assumptions, may not always achieve the high level of accuracy demanded by the evolving and increasingly complex manufacturing landscape [18]. The effort to correct or compensate for inaccuracies in data analysis due to the indirect measurement methods of generic sensors can lead to additional computational processing, further increasing energy use.
In addition to that, the complexity coming from the volume, variety, and velocity of the captured data often exceed the common analytical capacity This abundance of data can lead to increased energy consumption as more resources are required to store, process, and analyze this data effectively.

Another possible cause has been identified during research concerns the DRAM design.
The Dynamic Random-Access Memory (DRAM) activities that are energy inefficient; in this case, from research, it is known about two options: data transfer and data access, which are accordingly:

- Coarse-Grained Data Transfers: This leads to energy waste by transferring data not used by the processor due to the fixed size of data transfer bursts between DRAM and the memory controller.
- Coarse-Grained DRAM Row Activation: Activating large numbers of DRAM cells, even those not needed for many workloads, results in unnecessary energy consumption.

So, it is necessary to study how these two activities work in details in order to find the cause of the energy wastage.
Regarding the first task, DRAM memories send and receive data in 64-byte blocks, which are called cache blocks.
When the processor (CPU) demands data from memory, it usually ask for less than 64 bytes of information. However, due to the way the DRAM memory is organized, data is transferred in full 64-byte blocks each time.
This means that, although an entire 64-byte block is transferred, only a segment of this block is actually used by the processor. The rest of the block, which is not used, still uses space and requires energy to be transferred through the memory channel, and it consumes a lot of energy[38].
Thus, transferring parts of the cache block that are not being used is a waste of energy, as it consumes resources to transfer and maintain data that ultimately serves no purpose.
Another, related to previous, energy inefficiency of DRAM regards the data access. DRAM memory works by activating by applying power portions of memory to access data. These portions are quite large, generally between 8 and 16 kilobytes (KB) per activation.
When a program or operation requires access to data in DRAM memory, it doesn't need to use all data in the entire activated portion. Usually, only a small part of this data is actually used. This involves a waste of energy because DRAM is consuming energy to activate and keep all memory cells active in the portion, although many of these cells are not used. In other words, you are spending energy to activate parts of memory that are not going to be used [38].

Another inefficiency cause stems in the deep neural networks functioning. Deep neural networks are increasingly utilized in the cloud computing environment for processing, analysis, and monitoring tasks. Leveraging their capacity to model complex data relationships and patterns over time, these advanced computational models offer significant improvements in monitoring the performance of cloud applications. By effectively capturing and analyzing temporal dependencies in performance metrics, deep neural networks facilitate more accurate forecasting and dependency detection, enhancing cloud service reliability and efficiency [80].
Deep Neural Network (DNN) is described as a type of neural network that includes more than two layers. The structure of a DNN is such that it comprises multiple layers connected to each other, and they are used in a one-time training process before the DNN is ready for inference, which is the process of making predictions based on the data it has learned.
Overall, a DNN works by processing data through its layers: each layer receives input from the other one and it applies its learned elements to these inputs, and then produces the output. These outputs then serve as inputs to the next layer, and this process continues through the network. The interaction between the layers, through the transformation of input into output by applying weights, is important for the network's capability to learn and make accurate predictions.
One important feature of contemporary DNNs is their depth; in fact, they are made up of hundreds of layers, which leads to an enormous number of trainable weights. A feature of DNNs called overparameterization contributes to their high accuracy in tasks like language processing and image recognition, among others. DNNs are greatly enhanced in their ability to learn and generalize by overparameterization, which allows them to comprehend intricate input-output relationships and extract high-level semantics from data [36]
This explanation was needed to introduce what are the implications of these DNN in term of computational and memory resources, affecting as immediate consequence the energy consumption [36]. In the following bullets-point there is a better description of the root inefficiency:

- Computational and Memory Demands: DNNs are a key characteristic in advanced computing thanks to their ability to effectively tackle complex problems. However, these networks come with high computational and memory requirements, making it hard to have the requested energy efficiency, especially in edge device settings where low energy and real-time responses are critical. The growing complexity of DNNs, which is highlighted by the peak increase in the size of models, worsen these demands. There is a trend to reach bigger and more complex networks which create more worries around the computational resources and memory systems, highlighting the need for optimized architectures and dataflows specifically designed for efficient DNN functioning [36].

- Energy and Latency Concerns: One of the primary problems associated with DNN work is the energy consumption and latency associated with off-chip DRAM. This memory can consume between 30 to 80% of the system's energy in DNN accelerators [36]. The high latency of DRAM can result in a very high service time. This issue is particularly true for DNN because they not only ask for high computational power but also depend heavily on memory access. The latency challenges are even more worst in the case of irregular DNN inference, making necessary the research for optimization solutions regarding both energy efficiency and latency[36] .

Another important issue Is the data management at server level. Many inefficiencies states regarding the utilization of the server are the consequence of inefficient resource allocation, which can lead to have servers overloaded, underloaded or idle. Resource allocation and data distribution management are important factors to consider when dealing with energy consumption of cloud servers [57].

Also, another possible cause for idle servers, is the overprovisioning. This practice in cloud computing systems involves allocating more computing resources (such as CPU, memory, and storage) than what is currently needed to handle peak loads efficiently. This approach ensures that the cloud infrastructure can adapt to sudden increases in demand or workload spikes without compromising on performance or availability [81].

Another important trouble is given by cooling systems. Some cooling system are inefficient, as for instance, the fan because air is less efficient at absorbing heat compared to other cooling mediums. Additionally, the mechanical technologies used for air cooling generate heat themselves, further increasing energy consumption [82].
In data center air cooling systems, hot air mixing is a major problem that reduces cooling efficiency. It happens when hot air released from servers combines with cooled air intended to lower server temperatures. This procedure reduces the efficiency of cooling systems, resulting in wasteful energy use and possibly the formation of hot spots [83].

4.6 Energy Mitigation Strategies

This section proposes different solutions found in literature with the aim of applying corresponding solutions to inefficiencies and energy related problems. This is an important step because it is possible to associate the root cause of the inefficiency with the corresponding solution and implement it. Improving operational efficiency in cloud manufacturing environment brings to better energy consumption and allocation, reducing wastage. In the framework it is possible to link the causes with the solution thanks to the number of each box as shown in *Figure 24*.

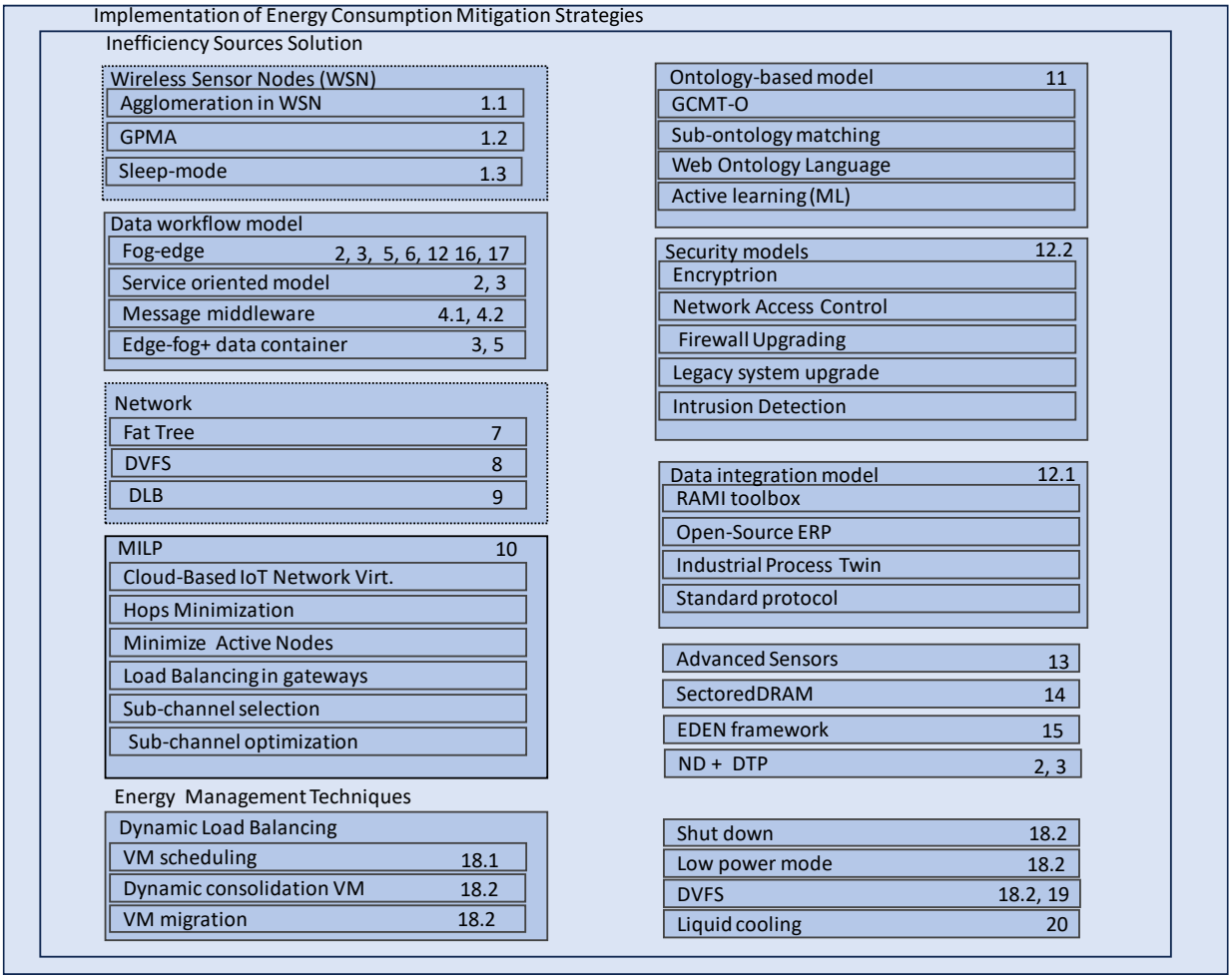


Figure 24. Framework section regarding the implementation of energy mitigation strategies

4.6.1. Energy Efficiency Improvement Technological Solutions

Following the structure of the previous section ("Root Cause Analysis"), this section illustrates several solutions at the same corresponding order.

The first root cause presented concerns the computational complexity provided from the data fusion algorithm at cloud server. According to the source [50], one solution proposed is about moving the agglomeration step of the data fusion technique to the WSN level by clustering sensors that gather the same type of information [50]. This solution aims to reduce the complexity of the data fusion operation in the cloud. By doing the agglomeration step before sending data to the cloud, cloud servers have less computation to do in order to merge data. This solution is highlighted in the framework regarding the solutions and belonging to the group "WSN", and it is illustrated in *Figure 24*.

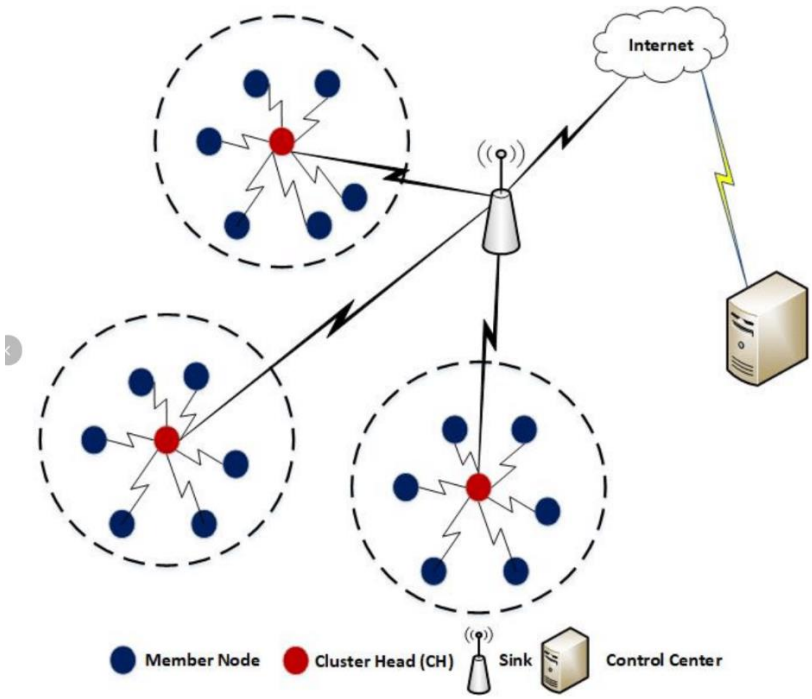


Figure 25. Clustering step hold in wireless sensor nodes to reduce computations on cloud [84]

The second solution proposed in the framework is the "Generalized Particle Model Algorithm" (GPMA) [50] which is used to optimize the hierarchical cluster formation process by allocating optimal paths to the base station to minimize energy consumption. This algorithm is designed to have a better hierarchical cluster formation process to reduce data redundancy. It also addresses the energy consumption problem faced by WSNs by reducing the complexity of the cluster formation process. It achieves this by optimizing the distribution of sensor nodes in the network, selecting the optimal nodes to form clusters, and allocating the best paths to the base station to minimize energy consumption [50].
Through the use of GPMA is possible to develop energy-efficient WSNs that can be used in a range of industrial applications, as such is the cloud manufacturing.
This solution is presented in the framework in the same category of the previous one as follows:
Overall, it is also feasible to reduce the number of active sensors and conserve energy by putting non-passive components into sleep mode when not used, this approach could help reduce the overall cost of WSN deployment by prolonging the sensor's life [50] and to have better data management and transfer by reducing redundancy.

The traditional cloud manufacturing model can be adjusted by the introduction of the edge fog layer, which acts as a link between the cloud and the manufacturing floor. The introduction of this middle step aims to do data processing and analysis closer to the manufacturing source, enabling the reduction of the quantity of data that needs to be sent to the cloud, thereby lowering latency and bandwidth consumption. Tasks like local data analytics, temporary data storage, and early decision-making can be handled at the edge-fog layer, improving real-time responses that are essential to manufacturing processes. By processing data locally and only transmitting the information that is required to the cloud, the edge fog layer can consequently help saving energy by reducing the amount of energy needed for both processing at cloud server layer and data transmission to cloud server [70]. In an Internet of Things (IoT) setting, the fog layer's role, according to the source [70], is the one of processing data and derive

insights from the information generated by IoT devices. Artificial intelligence applications and data mining are used in this processing. The fog layer works as a transitional layer between edge data collection and cloud data storage and processing. Because it allows data to be processed closer to the source, response times can be faster and less data needs to be sent to the cloud, which can result in more effective use of network resources and then lower energy consumption [70].
Figure 26 shows how the fog edge layers are introduced into the cloud manufacturing framework.

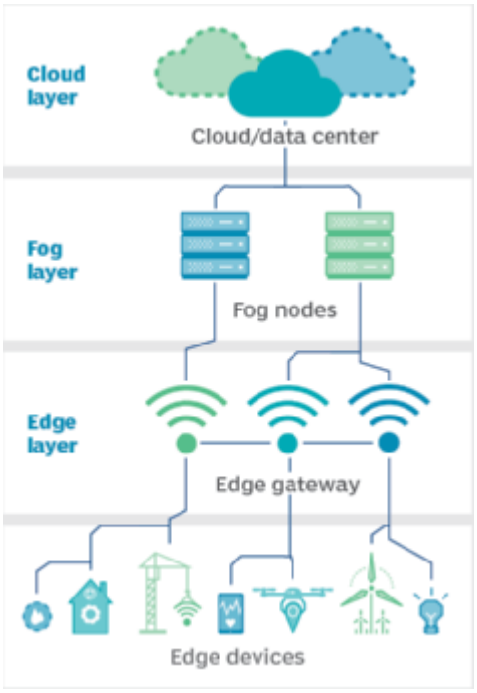


Figure 26. Fog-edge layer introduction in traditional cloud manufacturing system [85]

Introducing the fog layer means to create a dispersed and decentralized system that facilitates communication between cloud customers and specific service providers. The architecture of fog computing is intended to lower bandwidth bottlenecks and latency, which are prevalent in IoT environments. By extending the cloud network closer to the IoT endpoints, faster data processing and response times are made possible.
The fog layer also enhances other areas, such as:

- Ensuring that manufacturers maintain ownership of their data.
- Data processing and analysis happen at the source: this is called as data computing.
- Data sharing: Encouraging various stakeholders to share information with one another [70]
- Data aggregation: Putting together information from different sources to offer insightful analysis.

According to the paper [70], fog nodes can also be utilized as distributed system orchestrators, enabling Data Distribution-as-a-Service (D-DaaS), which means that these nodes manage the data flow between the primary High-Performance Cloud and Industrial Internet of Things (IIoT) endpoints, like for example with actual CNC machines. Users can access machine-level data by interacting with this cloud. The incorporation of fog computing can maximize the data movement in cloud manufacturing environment. By reducing the demand to transfer large

amounts of data to centralized cloud services for processing, this could potentially result in increased efficiency and lower energy consumption [70]
This solution is also capable to respond to huge variety and complexity of manufacturing system and resources, as there is a structured approach to enhance the operational capabilities of cloud manufacturing systems. This includes creating a service-oriented information model that provides a standardized description of operational data and attributes for manufacturing resources [42]. The following scheme in *Figure 27* shows how edge layer enables the information model from IIoT to cloud servers.

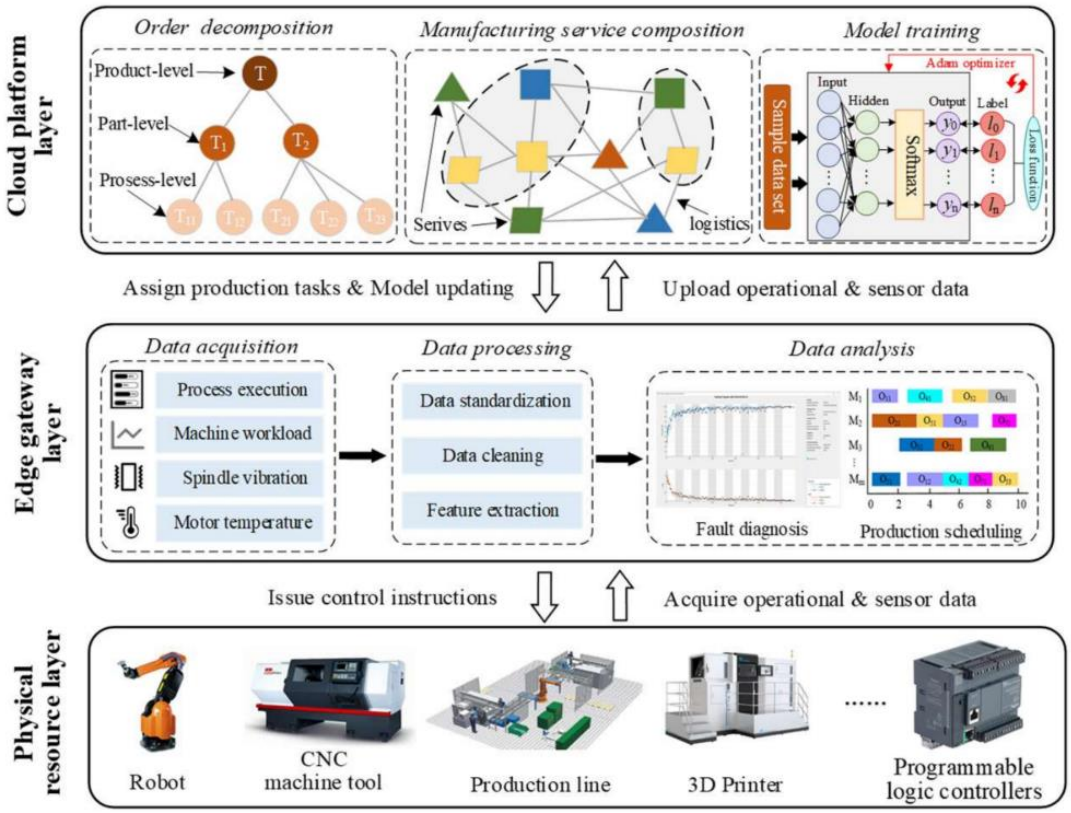


Figure 27. Service oriented model for standardization in manufacturing data operations, adapted from [42]

An important component to meet the real time requests in data transmission and integration is the message middleware. This is a software whose purpose is to facilitate efficient data distribution and enable remote monitoring, ensuring timely access to important operational data [42]. The middleware is used to cover the inability of legacy MES and SCADA software of making real-time operations, by creating this inherent communication between the systems, and also to improve the architectural limitations of Industrial Internet of Things regarding the capability to hold all the sheer volume of data. *Figure 28* illustrates how message middleware is introduced in the infrastructure.

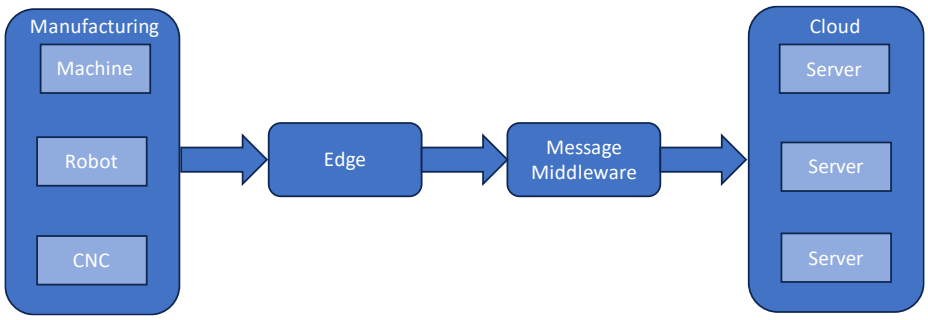


Figure 28. Message middleware to enable efficient data transmission, adapted from [42]

The paradigm including edge, service model and message middleware is designed to support the training and updating of AI models directly at edge gateways, with the purpose of increasing the system's efficiency and responsiveness. In this way, cloud manufacturing systems can achieve greater operational efficiency, improved real-time monitoring capabilities, and enhanced responsiveness removing bottlenecks in cloud servers and reducing the distances data need to make to be processed. The primary focus of these proposed solutions is to elevate operational efficiency and system responsiveness, but they also aid in energy conservation. By streamlining resource usage, optimizing manufacturing processes, and reducing redundant data processing and transmission, these improvements can lead to significant reductions in energy consumption [42].
In order to better understand how the data workflow in the edge-fog layer system, there is a definition to distinguish the different tasks among this infrastructure and see how the components interact between each other:

- Cloud operations focus on handling massive amounts of data and executing computationally intensive industrial applications. They handle more complex, less time-sensitive data processing tasks and oversee and assess data preprocessed by edge devices for broader decision-making processes. Cloud tasks include things like processing and analyzing large datasets that demand a lot of processing power, executing sophisticated industrial applications that require a lot of computation, storing and managing huge quantities of data from manufacturing processes, and doing in-depth analyses on preprocessed data to make well-informed decisions [42].

- Edge layer operations are made to distribute the computational load with the specific purpose on real-time data analysis applications. In order to lower the amount of data transferred to the cloud, they preprocess operational and sensor data from manufacturing resources, doing preliminary analysis and filtering. In addition, edge operations involve local data processing and analysis, sending only the information and insights needed to take additional action in the cloud. At the data generation point, using AI models for tasks like fault diagnosis and operational optimization reduces latency and improves response times [42].

- Industrial Internet of Things (IIoT) are an essential component of this model because they facilitate efficient communication and connection between manufacturing resources and enable instantaneous transmit of operational data to cloud platforms and edge nodes, improving the operational activities of manufacturing floor [42]. In the following picture (*Figure 29*) it is possible to see different areas in which industrial internet of things lie and help to Improve the manufacturing environment.

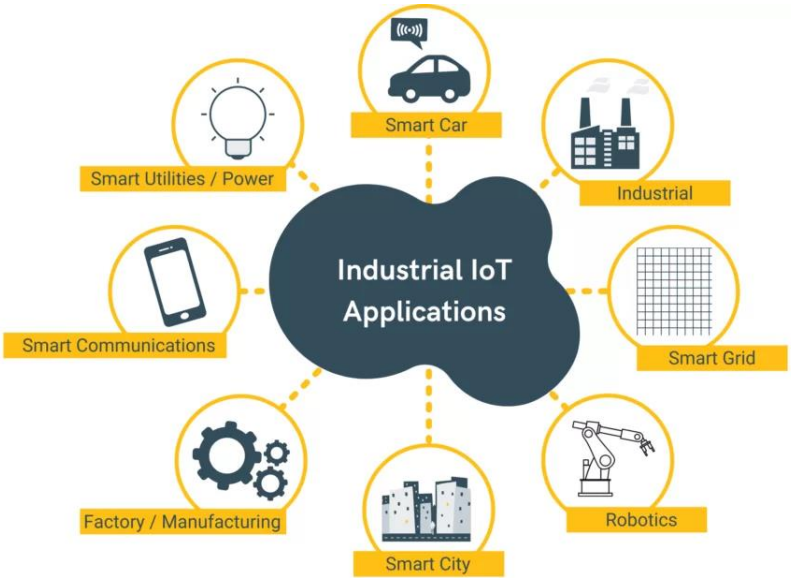


Figure 29. Illustration of Industrial Internet of Things Application, adopted from [28]

According to the source[69] , the edge fog layer is also a solution, in particular to address the issue of latency coming from long distance communication route and the issue of bandwidth usage by alleviating the amount of workload that arrives to cloud server due to the centralized organization. As a matter of fact, bringing a source of computation closer to the manufacturing site led to advancements in cloud technologies[69] .

- The suggested approach greatly lowers data processing latency by combining edge and fog computing layers. Decentralizing the processing duties enables data to be processed closer to its point of generation, resulting in this reduction. This means that, in the context of cloud manufacturing, data from sensors and manufacturing equipment can be processed locally or on-site instead of being transmitted to remote cloud data centers. Because of this, manufacturing operations' time-sensitive processes can be completed more quickly, improving overall operational efficiency and lowering energy consumption related to delays and longer operating hours.

- The solution uses local data processing to address the bandwidth consumption issue. The amount of data that needs to be moved to the cloud is greatly decreased by processing data on edge or fog devices. In addition to reducing bandwidth requirements, this local processing also lowers data transfer expenses. Additionally, by reducing the frequency with which massive amounts of data must be sent over the network, shorter transmission distances result in a significant reduction in energy consumption associated with data transmission. Furthermore, as suggested in [69], the implementation of big data management platforms on edge and fog devices makes use of these devices' enhanced processing power. This method considers the distributed nature of manufacturing data and allows for more effective and scalable processing. It also offers the flexibility to scale resources up or down based on real-time demands, contributing to more sustainable energy consumption and operational efficiency.

- Improving data security and privacy is an additional benefit of processing data closer to its source. The likelihood of security breaches and unauthorized access can be reduced by limiting the exposure of sensitive information over the internet and other networks and by keeping it localized. This is especially important in manufacturing settings where sensitive and proprietary data needs to be protected with strict security protocols.

According to the source [70], there is another solution which helps to increase efficiency in a traditional cloud manufacturing system, which is called “Hierarchical Data Management”. This organization includes the edge fog layer described until now, so it proposes a structure where data originates at the edge nodes (the ground floor), is preliminarily processed in the fog layer (the middle floors), and undergoes intensive processing in the cloud (the top floor). This solution adds something new, that is the introduction of data containers act. Data containers work like smart elevators, determining which data should be processed at which level to prevent bottlenecks and ensure timely processing. Data containers role is to manage, control and decide what is the best workflow data should do thanks to the use of the monitor [70]. This solution uses the dynamic load balancing as well,which will be explain later on in the section 4.6.1. The data container organization structure is given in *Figure 30,* and it is used in edge-fog layers.

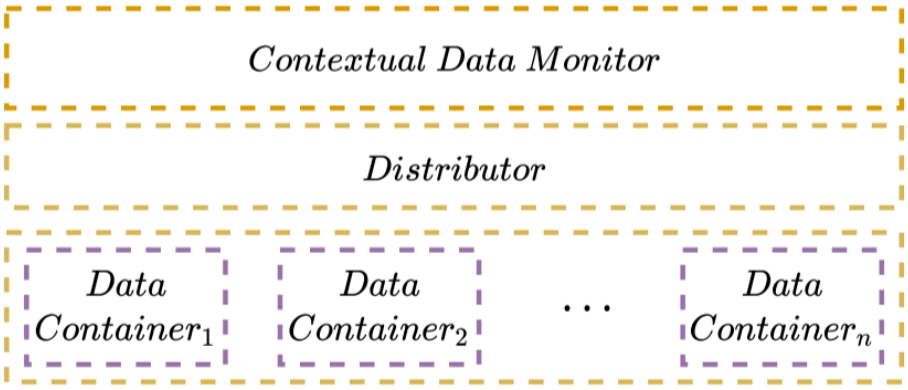


Figure 30. Hierarchical Data Management through data containers usage [70]

During the root cause analysis, it emerged that also the type of topology network adopted in the cloud might affect the data transmission efficiency and its environmental impact.
A network topology that better fits the environmental requests and not only, is the Fat Tree (FT) topology, which is designed to mitigate many of the issues related to transmission latency and bottlenecks by providing a more flexible and scalable network infrastructure that can handle large volumes of data traffic more efficiently. The FT topology increases bandwidth at higher layers of the network, reducing bottlenecks and supporting better load balancing across the network. This design can lead to improved energy efficiency since it can more effectively match network capacity to demand, reduce the distance data needs to travel, and decrease the probability of congestion [71]
Adopting an FT topology (*Figure 31*) can be seen as a potential solution to reduce energy wastage in data centers by ensuring that the network can handle data flows more efficiently and adaptively. However, the choice of network topology should also consider other factors such as the specific requirements of the data center's workload, the cost of transitioning to a new topology, and the potential need for new equipment and technologies to support the chosen architecture [71].

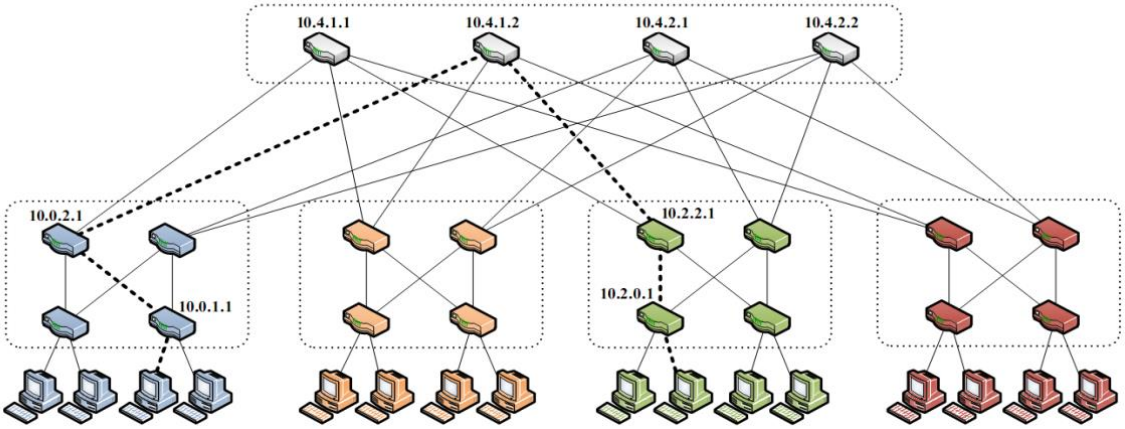


Figure 31. Fat Tree Network Topology for a more flexible and efficient data transmission [86]

Techniques like dynamic bandwidth allocation adjust capacity based on demand, reducing unnecessary energy expenditure regarding inefficient bandwidth usage [71].
Techniques like load balancing distribute traffic evenly across network resources, ensuring no single device consumes excessive power due to overuse [71].
These two solutions will be explained better later on this section.

As per reference [73], a novel strategy for addressing sensor interference inefficiency in a cloud-based Internet of Things (IoT) environment involves developing and putting into practice an energy-efficient traffic model. This model employs the "Mixed Integer Linear Programming (MILP)" [73], which uses several important strategies, to optimize network operations by lowering energy consumption and interference effects through the following methods:

- By using MILP, the model creates an optimized framework that considers both cloud computing resources and IoT equipment, thereby virtualizing the IoT network. This the virtualization process is essential for controlling the intricate structure of the network and facilitating effective resource allocation and route design [73].

- Finding the least number of hops required to connect IoT devices to cloud infrastructure is one of the main goals of the suggested model. Because fewer hops typically mean less power is needed for data to reach its destination, this approach substantially lowers the energy utilized during data transmission [73].

- The model simultaneously reduces the number of devices that must be active at any given time by carefully choosing routes that minimize the number of hops. This helps to lower possible points of interference within the network in addition to lowering overall energy consumption [73].

- The model distributes the load evenly among the different intermediaries to minimize traffic congestion. In order to avoid data transmission bottlenecks, which can result in higher interference and energy consumption, load balancing is crucial [73].

- The new feature of this model focuses on optimizing the selection of sub-channels based on their fading gains, aiming for energy efficiency. This approach allows for the identification and utilization of sub-channels that reduce interference and lower overall

network energy consumption. Essentially, the model prioritizes data transmission over sub-channels that offer the most energy-saving paths by leveraging their fading gains. The core strategy is to diminish the amount of power each Internet of Things (IoT) device needs to transmit data by selecting channels that optimize the beneficial effects of fading. Fading gain refers to the phenomenon where signal strength improves due to the constructive interference of multiple signal paths, which may occur through reflection, refraction, or scattering. As signals in wireless communications may take various paths to reach their destination, fading gain captures the enhanced signal strength at the receiver when these signals combine positively. This mathematical model aims to maximize the network's energy efficiency by choosing channels with the highest fading gain, thus enhancing signal strength through constructive signal path interference [73].

- An additional focus of the approach is the efficient utilization of time windows on energy-efficient subchannels. This temporal optimization lowers energy consumption and enhances communication channel performance by timing transmissions during times when interference is anticipated to be minimal [73].

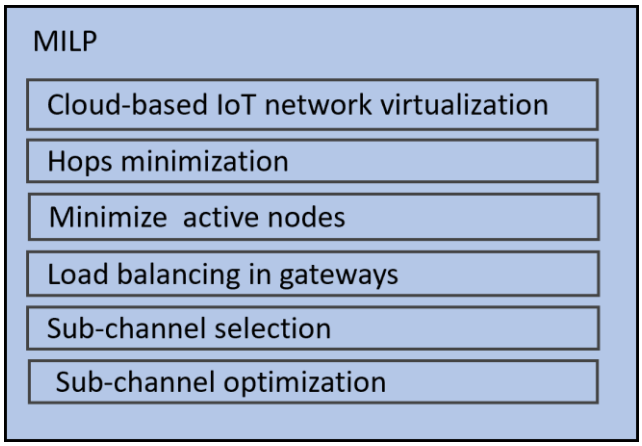


Figure 32. Mixed Integer Linear Programming model steps for efficient IoT utilization, Framework box representation

Through these strategies it is feasible to address the dual challenges of energy consumption and sensor interference in cloud based IoT networks. According to [73], the model implementation brings notable gains in network sustainability, dependability, and efficiency, demonstrating the possibility of sophisticated mathematical modeling and optimization methods to get around the intrinsic constraints of existing IoT infrastructures.

According to reference [74], an holistic procedure that makes use of innovative computational techniques and ontology-based modeling is recommended to address the problems associated with semantic representation in cloud manufacturing. Creating a semantic framework that accurately expresses and captures the complex nature of manufacturing tasks is the first step. This approach is called the "General Cloud Manufacturing Task Ontology" (GCMT_Ontology)[74], which is a structured semantic model designed to organize and define the vast array of concepts, attributes, and interactions inherent in manufacturing tasks. This ontology provides a foundational framework that enables machine- and human-readable manufacturing data representation.

The key to ensuring that this semantic model appropriately captures the unique characteristics of every production step is sub-ontology matching (*Figure 34*).
This process involves identifying the specific semantic elements of a task and integrating them with the broader GCMT_Ontology (*Figure 33*). This makes it possible to dynamically adapt the ontology to particular tasks, increasing the model's applicability and relevance across a variety of manufacturing scenarios.
The Web Ontology Language (OWL) is used to further support the semantic model. OWL is selected due to its high level of ontology consistency guarantee and computerized reasoning support. This helps preserve semantic representations and facilitates collaboration and seamless integration of manufacturing task data across multiple platforms and systems.
In order to expand the ontology, the plan also includes a method for adding new notions or connections that emerge during the matching process. Creating connections between old and new semantic terms is essential to maintaining a comprehensive and up-to-date ontology given the evolving nature of manufacturing tasks. An algorithm for machine learning is used to accomplish this [74].
This comprehensive framework is illustrated in the framework as shown in *Figure 33.*

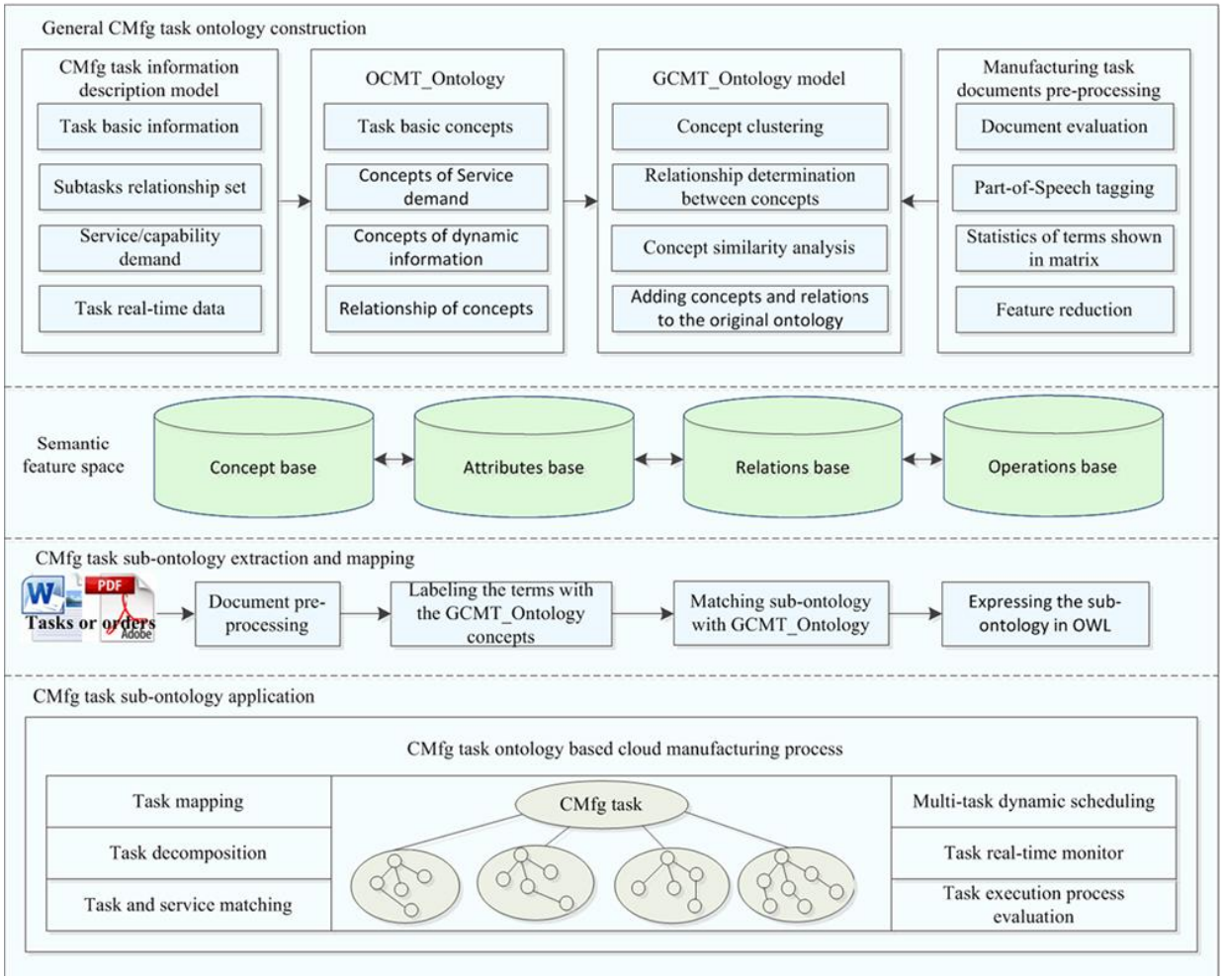


Figure 33. General Cloud Manufacturing Task Ontology framework to reduce semantic issues problems in cloud manufacturing, adapted from [74]

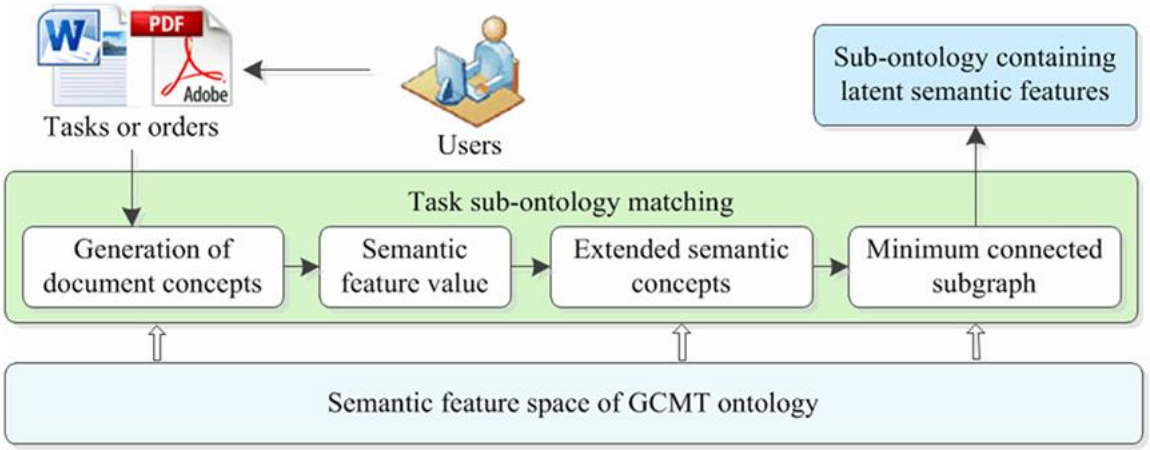


Figure 34. Sub ontology matching process, adapted from [74].

These elements merged together can provide a strong approach to solving the semantic representation issues in cloud manufacturing. By creating a dynamic, accurate, and thorough semantic framework, this approach aims to improve the efficacy, connectivity, and versatility of manufacturing processes in the environment of cloud computing, facilitating greater efficiency and seamless completion of manufacturing tasks [74].

Another inefficiency present in the cloud manufacturing management is the integration with Information Technology and Operational Technology. This inefficiency brings two main problems, the first concerns the security of information in the infrastructure and the other concerns the flow of information and its integrity.
The solutions listed are aimed at addressing the complex cybersecurity challenges posed by the integration of IT and OT systems in cloud manufacturing environments, focusing on securing the network layer as a critical foundation for broader security measures [75].

- Network Assets Identification: the identification and management of all network assets is the first step towards securing the IT/OT integration. Enabling thorough Network Access Control (NAC) and efficient segmentation requires completing this step. Deploying comprehensive security measures requires an understanding of every asset on the network, including legacy devices that might not support contemporary authentication methods [75].

- Network Access Control Implementation: In order to guarantee that only authorized devices are able to communicate over the network, Network Access Control is necessary. Strictly implementing NAC stops assets not in the asset management database from interacting, which lowers unwanted access. By facilitating the automatic assignment of network users to the appropriate Security Groups (SGs), NAC helps to minimize manual configuration errors and enforce micro-segmentation [75].

- Segmentation: To avoid unapproved moves within a network by imposing strict regulations within various areas, the concept makes use of multiple layers of segmentation. This consists of micro-segments made with Security Groups and VLANs (Virtual Local Area Networks) for more precise control, as well as macro-segments for general division. Micro-segmentation based on hierarchical groups facilitates

categorizing of groups or the passing mechanisms to streamline configuration and management procedures, which is particularly useful for large-scale deployments [75].

- Encryption: In order to guarantee communication integrity and confidentiality, the paper recommends using MACsec for hop-by-hop encryption, taking into account the unprotected nature of Industrial Ethernet (IE) traffic. MACsec is the Media Access Control Security, that is a link-level security standard that provides encryption and authentication between devices connected through Ethernet networks This precaution is especially crucial for older Internet Explorer protocols that aren't encrypted. While encryption prioritizes integrity over confidentiality for real-time input/output data, it still aids in data protection during network transmission [75].

- Intrusion Detection System (IDS) and Next Generation Firewall (NGFW) adoption: Intrusion Detection System (*Figure 35*) monitors for suspicious activities by inspecting traffic and utilizing anomaly detection algorithms and signature recognition. It works in conjunction with Next Generation Firewall, which can actively block suspicious traffic. These systems are crucial for securing communication channels, especially those crossing micro-segment boundaries, to prevent unwanted lateral movement and potential intrusions [75].

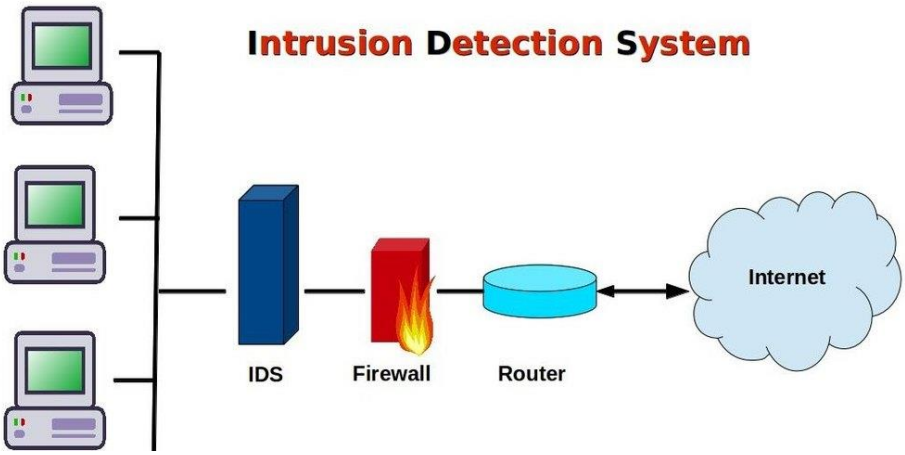


Figure 35. Intrusion Detection System, adapted from [87]

The solutions provided until now regarding IT-OT integration are illustrated as follows:

- The management and security of computational resources are being profoundly altered by the manufacturing sector's adoption of edge cloud technology. These resources were historically dispersed throughout the factory floor, which frequently resulted in operational inefficiencies and security threats. Manufacturing is streamlined and secured by combining them onto an edge cloud platform. This configuration makes it possible to implement cutting-edge IT/OT security measures, which are essential for preventing contemporary risks [75].

- Additionally, the edge cloud allows for the virtualization of essential manufacturing operations, like those managed by Programmable Logic Controllers (PLCs), onto a unified platform. This centralization not only makes systems easier to manage and maintain but also enhances security through better access control and monitoring. According to the paper, this technological progress helps the manufacturing sector

meet the growing need for immediate data processing, consistent uptime, and robust reliability without sacrificing security [75].

- Updating and Upgrading Legacy Systems: this is a fundamental action to do in order to overcome vulnerabilities to cyber-attacks, dependence on outdated software, lack of security updates and a better interoperability in data integration that leads to better process optimization [79].

Information Technology and Operational Technology systems often have contradicting requirements due to their different characteristics, leading to increased overall system complexity. These solutions aim to address system complexity created.

- Utilization of the RAMI Toolbox: The RAMI (Reference Architecture Model Industrie) Toolbox is designed to enhance the usability and applicability of RAMI4.0, a framework for Industry 4.0 system architecture. The toolbox offers functionalities like automatic model transformation, interfaces to other tools, and model-checking possibilities. It supports a Model-Based Systems Engineering (MBSE) approach, ensuring compatibility with RAMI4.0 and holistic traceability between models. This helps guide users through complex system engineering tasks, addressing the integration challenge by enabling a clear and structured development process that aligns with RAMI4.0 standards.

- Introduction of the Industrial Business Process Twin (IBPT): The IBPT acts as an intermediary between IT and OT, aiming to reduce the complexity of OT systems for IT stakeholders and vice versa. By using a semantic information modeling approach across RAMI4.0 layers, the IBPT abstracts system functionality and facilitates the decoupling of OT and IT concerns. This not only simplifies the integration process but also ensures bidirectional exchanges between IT and OT components, promoting seamless interconnectivity and orchestration of activities.

- Adoption of Open-Source ERP System: the replacement of the SAP system with the Odoo ERP system (formerly Open ERP), that is an open-source and customizable solution. This approach addresses the customization and integration issues presented by proprietary solutions, making it easier to adapt the ERP system [78].

- Integration through Standard Protocols: The use of standard protocols for communication between the manufacturing module and resources is useful. Indeed, this ensures that tasks can be assigned and managed efficiently, supporting the customizability and interoperability of the system [78] .

The mentioned solutions are illustrated in the framework in *Figure 36.*

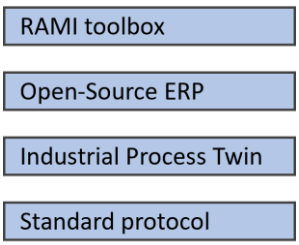


Figure 36. Different solutions to ensure interoperability regarding IT-OT integration systems, Framework representation

Regarding sensor inaccuracies, some solutions are proposed according to [18]:

- Development of Advanced Sensors: Emphasizing the need for innovative sensor technology to facilitate accurate and direct measurement in complex manufacturing scenarios, leading to high fidelity and precision in data collection [18]. An example of this is illustrated in *Figure 37.*
- Exclusive Sensor Design: Despite the push towards leveraging big data, there is a noted trend towards designing exclusive sensors tailored for specific measurement tasks in the manufacturing process to ensure precision and reliability [18].

The framework shows these solutions in the same unique box:

Figure 37. Advanced vision sensor example to have detailed data, adapted from[88]

Another inefficiency to take care of concerns the DRAM component and its function. According to the source [38], Sectored DRAM is a corrective solution for DRAM inefficiencies. Indeed, it aims to improve system performance and reduce energy consumption on the memory channel by enabling fine-grained DRAM access and activation.

The "Sectored DRAM" [38] solution introduces a fine-grained DRAM architecture designed to improve system performance and reduce energy consumption. Traditional DRAM architectures access and activate large blocks of data, which can be inefficient, particularly for applications that require only small portions of these blocks. This inefficiency leads to higher energy consumption and reduced performance, especially for memory-intensive workloads with irregular access patterns.

This DRAM architecture addresses this problem by enabling more precise access to smaller data segments within the memory. This fine-grained access reduces the amount of energy wasted on unneeded data and improves overall system performance by decreasing memory access times. The key contributions of Sectored DRAM include the introduction of mechanisms like Variable Burst Length and Sectored Activation, which together allow for this fine-grained access while maintaining low hardware complexity and minimal area overhead on the DRAM chip [38].

The framework presents this solution in the following way. The source presented some results, in particular it states that Sectored DRAM can reduce energy consumption by 20%, improve system performance by 17% on average, and decrease overall system energy consumption by

14% compared to systems with coarse-grained DRAM. The implementation of “Sectored DRAM” [38] is estimated to take up only 1.7% of a DRAM chip area, making it a low-complexity solution for enhancing energy efficiency [38].

“Energy-Efficient Deep Neural Network Inference Using Approximate DRAM” [36] is a novel framework designed to enhance the performance and energy efficiency of DNN (Deep Neural Network) inference systems by utilizing approximate DRAM (Dynamic Random-Access Memory). The growing memory demands of DNN workloads have made main memory a significant contributor to the overall system energy consumption and latency. EDEN addresses these challenges by leveraging the intrinsic error tolerance of neural networks and exploiting the potential of approximate memory technologies to reduce energy consumption and improve inference performance while maintaining a specified target accuracy.

At its bases, EDEN works on two insights: the first is that neural networks inherently possess a capacity to tolerate errors in their computations, which can be leveraged to allow for the use of less reliable, yet more energy-efficient and faster memory systems; while the second is DRAM can be operated under approximate conditions, such as reduced supply voltage and access latencies beyond standard specifications, to decrease energy consumption and latency at the cost of increased bit error rates. However, these bit errors introduced by approximate DRAM do not significantly impact the accuracy of DNNs due to their error tolerance capabilities.
EDEN's approach comprises several key steps:

- EDEN introduces a method called "curricular retraining" to increase a DNN's tolerance to errors. This involves retraining the DNN using data that simulates the error characteristics of the target approximate DRAM device, thereby enhancing the network's resilience to such errors.

- After improving the DNN's error tolerance, EDEN characterizes the error resilience of different data types within the DNN (e.g., weights, input, and output feature maps) and maps these data types to specific partitions of the approximate DRAM. This mapping ensures that the DNN's accuracy requirements are met by assigning data elements with lower error tolerance to DRAM partitions with lower bit error rates, and vice versa.

- By carefully managing the allocation of DNN data across the DRAM and utilizing the retrained, error tolerant DNN, EDEN enables significant reductions in DRAM energy consumption and latency. The framework allows for a systematic approach to adjusting DRAM operational parameters (e.g., supply voltage, access latencies) to achieve optimal performance and energy efficiency within the bounds of the desired DNN accuracy [36].

The framework has shown significant improvements in energy consumption and performance: By reducing DRAM supply voltage and latency, EDEN achieves an average DRAM energy reduction of 32% across various computing architectures, including CPUs and GPUs, for instance.
This procedure in *Figure 38* which aims at increasing energy efficiency in deep neural networks is described in the framework as in the following image:

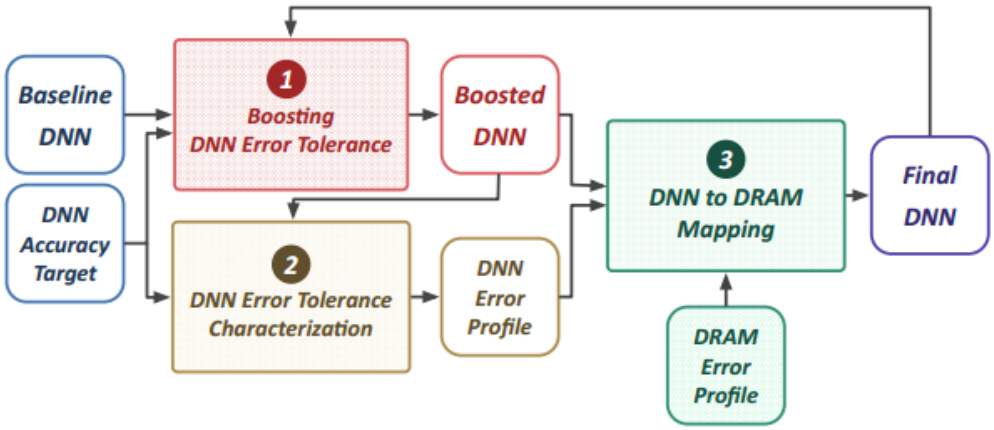


Figure 38. Framework description of the Energy-Efficient Deep Neural Network Inference Using Approximate DRAM, adapted from [36]

The inefficiency concerning the quantity of data gathered, stored, and elaborated can be addressed with the following solution, as proposed in this paper [55], which includes:

- Characterizing the dataset using metrics that assess redundancy and complexity.
- Identifying and removing redundant data to reduce the dataset size.
- Applying big data preprocessing and machine learning algorithms on this reduced, more manageable dataset to achieve efficient and effective data analysis.

This solution introduces and employs two new metrics for assessing big datasets in classification problems, Neighborhood Density (ND) and Decision Tree Progression (DTP), which are used to measure the dataset's density and the impact of data reduction on classification accuracy, accordingly. These metrics provide valuable information regarding the redundancy and complexity of the data, which can help in making informed decisions about data preprocessing and the necessity of using the entire dataset for machine learning tasks.
By measuring how density changes with a reduction in the dataset, ND helps identify the degree of redundancy within the data. A small change in ND after discarding a significant portion of data suggests that the dataset has a high redundancy, indicating that it might be possible to reduce the dataset size without losing essential information.
DTP is used to assess whether a dataset contains redundant information that does not contribute significantly to the predictive power of the model. A small difference in accuracy suggests that much of the data may be redundant, and thus, the dataset can be reduced without substantially affecting the model's performance [55].

4.6.2. Green Cloud Computing

The process of managing and making use of computing resources, like data centers, in a way that optimizes energy efficiency is known as "green cloud computing." Reducing resource energy consumption to the lowest feasible level is the goal. About 60% of the energy used in data centers is used for computing resource processes and refrigeration systems, which are crucial components. Effective processing and operation of data centers depend on the

implementation of energy resource management, which enhances performance in a number of critical areas. These include controlling traffic loads, cutting down on overall power consumption, and taking care of energy costs associated with businesses. The goal of the green cloud computing approach is to improve these areas in order to decrease energy consumption, improve load balancing, and eventually increase profitability [89].
Using renewable energy sources, recycling equipment after it can no longer be used, repurposing server heat to warm nearby buildings, and enhancing data center energy efficiency are some of the steps companies have taken to enhance the effectiveness and environmental impacts of their cloud computing [89].
Virtual machine scheduling, which is essential in cloud data centers to route client applications to the appropriate servers, is one technique to increase energy efficiency. Virtual machine scheduling has been put into place to ensure that a minimum amount of hosts are in an operational state for the purpose to improve service quality and efficiently manage power consumption. "Dynamic Consolidation of Virtual Machine" (DCVM) is the term for this capability [6]. Dynamic load balancing, or DLB, is a methodical technique designed to effectively distribute cloud computing resources (like CPU, RAM, storage, and bandwidth) between virtual machines (VMs) with the objective to ensure optimal resource utilization. It is another approach, including the preceding ones, to tackle the problem of resource allocation. Different approaches can be taken to implement these strategies depending on the methodology and algorithm chosen. In order to handle dynamic overloading problems, the DLB model moves virtual machines (VMs) from exceeding demand hosts to alternative hosts until the load is balanced. This process is known as VM migration and consolidation.
According to [6], there are four categories that address energy efficiency in cloud data centers, respectively:

- Heuristic methods which are rule-based approaches that make decisions on resource allocation and task scheduling to reduce energy consumption, utilizing simple but effective rules or formulas. One example is the First Fit Decreasing algorithm (FFD)[6].
- Metaheuristic methods which involve complex algorithms that simulate natural processes or apply game theory to find efficient solutions for energy optimization, balancing computational resources efficiently. One example is the Genetic Algorithm [90].
- Machine Learning technique that, by analyzing historical data, are able to predict future demands and optimize resource allocation to save energy, adapting to changing conditions automatically.
- Statistical methods utilize data analysis to identify patterns and make informed decisions about energy management, focusing on optimizing existing resources and predicting future needs based on trends.

This classification is shown in the following scheme (*Figure 39*).

Figure 39. Schematic illustration of the four categories of energy management strategies, adapted from [6]

4.6.2.1. Energy Management Techniques

For each of the mentioned energy management strategy's category, there is the explanation of how they work, and some algorithms found in the literature researches.
Regarding the first determined category (Heuristic), three different algorithms will be explained as depicted in *Figure 40*.

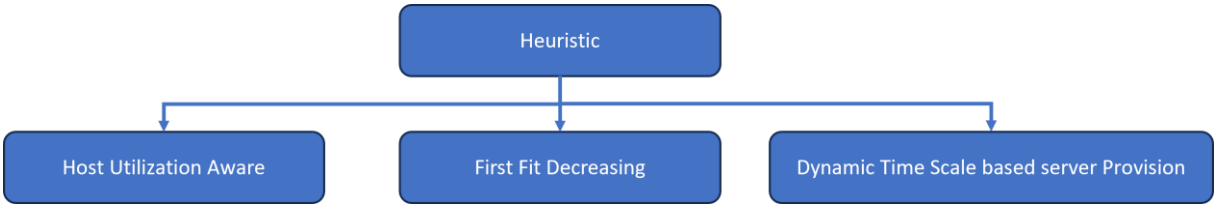


Figure 40. Scheme of heuristic algorithms described in this work.

Host Utilization Aware Algorithm:

The Host Utilization Aware (HUA) algorithm is one strategy that has been mentioned in the literature to address the underutilization problem. It belongs to the heuristic category and finds underutilized servers in a data center, or servers whose utilization of resources is less than a predetermined dynamic threshold. Because this threshold is dynamic and determined by the load and utilization levels throughout the data center, resource management can be done in a way that is more responsive and versatile. The strategy's key component is its capacity to adapt dynamically to the data center's workload demands, guaranteeing that resources are allocated as energy-efficiently as possible [10].
By contrasting servers' utilization with the dynamically calculated lower threshold, it is possible to spot underloaded servers. Virtual machines (VMs) on these underutilized servers can be identified and then moved to other servers with more capacity to handle additional workloads. In order to preserve service levels, this procedure is meticulously controlled to reduce any negative effects on the performance of the apps operating on these virtual machines [10].
The migration procedure is not random; instead, it is regulated by algorithms designed to maximize the placement of virtual machines (VMs) taking into account a number of variables, such as the present and anticipated demand on the servers, the VMs' resource needs, and the objective of reducing the data center's overall power consumption. To find the most effective distribution of resources, this frequently entails intricate computations and forecasts [10].

This strategy has two main objectives. The first is to lower energy consumption by turning off servers that are not in use, which will lower the data center's operating expenses and environmental effect. Its second goal is to increase the remaining servers' utilization efficiency, which will guarantee that resources are used as efficiently as possible rather than wasted [10]. In cloud computing environments, where resource demand can vary greatly, this method of handling underloaded servers is essential. Through the implementation of dynamic management tactics, cloud providers can guarantee optimal service delivery to their consumers by coordinating the demands of the effectiveness of costs, ecological responsibility, and reliability with performance [10].
It's crucial to note that while this algorithm is intended to address the issues with underloaded servers' utilization, it also recognizes the existence and difficulties of overloaded servers. In order to minimize energy consumption by shutting down idle hosts, it optimizes the placement of virtual machines (VMs) from overloaded servers to others. This also makes sure that the redistribution does not result in overloading [10].

First Fit Decreasing Algorithm

The First Fit Decreasing (FFD) algorithm is an adaptation of the First Fit algorithm that is usually used in bin packing problems. In the context of container placement in cloud data centers, FFD is a heuristic approach used to efficiently pack containers into Virtual Machines (VMs) or Physical Machines (PMs), with the objective of minimizing the total amount of active machines and, consequently, reducing energy consumption [13]. FFD works by sorting containers in decreasing order based on their RAM requirements before placing them into VMs or PMs. Thanks to the sorting phase it is possible to consider first the largest containers, which can lead to a more efficient packing by filling up the VMs or PMs with the largest items early in the process. Then,by checking the space in the virtual or physical server, it is possible to reiterate the phase to then take and place the container in the first VM or PM that has enough remaining RAM capacity to accommodate it. By packing the containers as tightly as possible according to their RAM requirements, the algorithm aims to minimize the number of VMs or PMs that need to be active, thereby reducing the overall energy consumption of the data center [13].

Dynamic Time Scale based server Provision

The Dynamic Time Scale based server Provision (DTSP) is an other heuristic technique. Based on the present and anticipated workload, the primary functionality consists of dynamically adjusting the quantity of servers that are in operation and placing idle servers in a low-power state [35].
The DTSP approach assesses the server workload by examining a number of important variables, including variations in request arrival times, the volume of requests that are received (both historical and present), and the average time required to process these requests. The dynamic character of workloads, which can fluctuate greatly over brief periods of time, is taken into account in this analysis [35].
The approach prefers to switch servers that aren't at the moment needed to a low-power state rather than completely shutting them down, which can cause a delay in response time when they need to be supplied back on. As a result, the servers use less energy and can reactivate more quickly when needed to handle an increase in workload [35].

The number of servers that are in an active or low-power state can be proactively adjusted by the DTSP method by precisely determining the workload demands through the previously mentioned factors. By making this adjustment, idle servers won't waste energy and there will be just the right number of active servers to manage the workload effectively. The ultimate objective of this strategy is to minimize energy consumption while maintaining or even improving the Quality of Service (QoS), or the datacenter's capacity to process requests in a reasonable amount of time. This approach is especially important in contemporary datacenters, where workloads can be very unpredictable and conventional server provisioning techniques could result in either excessive energy consumption (from having too many idle servers) or inadequate service capacity during periods of high demand [35].
According to the source [35], DTSP allocates additional servers to guarantee Quality of Service (QoS), which leads to increased average power usage. However, this rise in power consumption is balanced by dynamically adjusting the server count according to workload needs and switching idle servers to a low-power mode whenever feasible. As a result, despite its higher average power usage compared to other approaches, DTSP still manages to enhance energy efficiency measured in 'J/m^2'. It achieves this by utilizing less energy for the same workload while maintaining QoS. The extent of energy efficiency improvement is influenced by the variability of the workload and the specific QoS requirements. In conducted experiments, DTSP showed a 60% improvement in energy efficiency over alternative methods, as for example compared to Mithra method.

In the following section, the energy management strategies description continues with a meta-algorithm typology as shown in the following scheme of *Figure 41*.

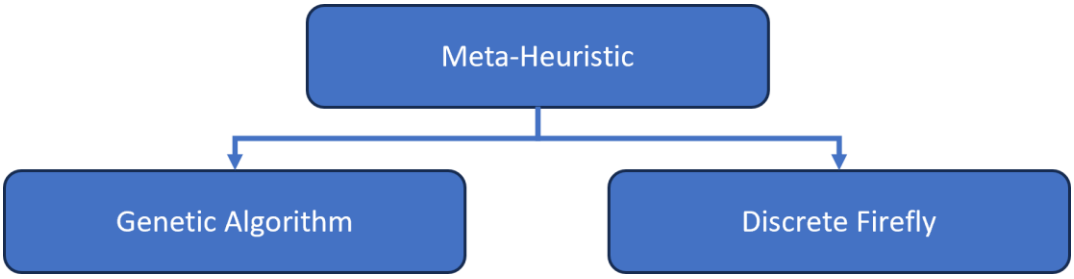


Figure 41. Scheme representation of described meta-heuristic approaches, adapted from [90]

Genetic Algorithm with Neural Networks

In cloud computing, the Genetic Algorithm (GA) is used for virtual machine scheduling to solve the problem of overloaded and idle servers. The goal is to distribute resources as efficiently as possible, which is becoming more difficult as demand for cloud computing services rises. As per reference [90], a novel approach to accomplish this objective integrates Genetic Algorithms (GA) and Neural Networks (NN) through the optimization of resource scheduling and allocation in cloud environments. This hybrid approach uses a "Neural Network Task Classification" [57] (N2TC) system to classify tasks and then a "Genetic Algorithm Task Assignment" [90] (GATA) process to assign resources to them.
As a member of the meta-heuristic group, the genetic algorithm put forth in [90] is a computerized technique influenced by the concepts of genetics and natural evolution. In order to find the best solution, this algorithm repeatedly chooses, combines, and modifies solutions

to optimization problems. GAs are employed in cloud computing to effectively divide computational workloads among available resources in order to maximize efficiency, economy, and response times [90]. In fact, fairness should be the primary consideration when defining the parameters for effective operation. This means that resources should be allocated to tasks based on their assigned importance or used in an equal manner for all tasks. By reducing the number of active servers and hosts, which maximizes energy consumption, cloud computing can reduce energy waste. Reducing the length of intervals is the aim of achieving a shorter makespan, which aims to accelerate task completion. Good load balancing is essential because it distributes work among resources so that no one is left idle and no one is overloaded. The costs that cloud users must pay for all necessary services, including processor use, data storage, and data transfer, are taken into account. Finally, when resources are utilized to the greatest extent possible with the least amount of time and resources wasted, system efficiency is maximized.

According to the source [90], the aim of this combined capability is the allocation of cloud computing resources in a way that balances several factors such as minimizing the execution time of tasks and reducing costs. The proposed GA-NN hybrid approach classifies tasks based on their requirements and characteristics, and then optimally assigns resources to these tasks. The goal is to improve system efficiency by ensuring that resources are utilized most effectively, thereby enhancing the overall performance of cloud services [90].

The variability and complexity of cloud-based computing environments—where demands and resources readily accessible can change quickly, are the driving forces behind the adoption of this hybrid strategy. Conventional approaches may not be flexible enough or may optimize for a single goal, such as cost or execution time, neglecting to take the overall performance of the system into account. Neural networks and genetic algorithms work together to provide a more complex and flexible solution that can take into account several goals and adjust to changing circumstances more effectively [90].

Discrete Firefly Algorithm

The other meta heuristic approach is the discrete Firefly Algorithm for Container Placement (DFF) which is a technique inspired by the natural behavior of fireflies, specifically the analogy is with fireflies' ability to attract each other with varying brightness levels.

Indeed, each firefly's "brightness" stands for the quality of a solution to an optimization problem, or, to put it more practically, the optimal placement of containers in cloud computing environments to maximize resource utilization and minimize energy consumption [90].

The model's operation can be broken down into several steps. The process starts with a population of solutions, where each "firefly" or solution stands for a possible method of distributing containers among virtual machines (VMs) in data centers.

Then, according to their brightness, which is correlated with the potency of the solution they stand for, fireflies are drawn to one another. A firefly attracts more attention from others the brighter it is (the better the solution) [90].

Fireflies mimic the search for optimal container placements by migrating towards brighter ones. Attractiveness, which diminishes with distance, influences this movement and promotes investigation of the solution space. Solutions are iteratively updated based on the movements of fireflies toward better solutions. The algorithm employs a discretization strategy to adapt the original firefly algorithm, which was designed for continuous spaces, to the discrete nature of the container placement problem [90].

By efficiently distributing containers among the fewest virtual machines and physical servers, DFF aims to minimize energy consumption and optimize container placement in cloud data centers.
Therefore, the objective is the same as it was in earlier methodologies: to maximize the use of computing resources, guaranteeing that physical servers and virtual machines are utilized to their full potential and minimizing waste, all while maintaining the same level of service due to the proper operation of container resources, which prevents overprovisioning or needless resource allocation [90].

In the next section there is the description of machine learning category in cloud computing for sustainability purposes.

Machine Learning

Machine learning (ML) is significantly advancing green cloud computing by enabling dynamic and efficient resource management that responds to real-time workloads, thus addressing the critical issue of energy consumption in cloud architectures. ML's capacity to learn from workload patterns and adjust is fundamental to resource management. It leverages data from cloud operations to forecast resource requirements on dispersed servers. This feature makes it possible to allocate the right amount of resources, closely match supply and demand, avoid wasting resources, and save energy [89]. Moreover, ML enhances virtual resource scheduling, which is a crucial procedure for preserving cloud applications' Quality of Service (QoS). It provides an effective framework for managing and allocating resources, preventing server overloads and needless operating expenses. Cloud servers can dynamically adjust to shifting workloads by allocating resources where they are most needed and scaling down when demand declines thanks to machine learning models [89]. It's impressive how ML can improve scalability. It can achieves this by supporting the parallelism management strategy, which reduces the intricacy involved in resource allocation and computations. Workload prediction models, especially the ones based on reinforcement learning, have been created to improve the efficiency of power management by setting the actions performed on cloud servers. Energy conservation depends critically on having a more sophisticated understanding of when and how to use resources, which is made possible by these predictive models [89].
Green cloud computing's primary goal is to reduce the amount of energy used by both software and hardware. Resource scheduling policies are optimized by ML integration into cloud management frameworks, which is essential for attaining high-performance and energy-efficient execution. Through this integration, intricate cloud usage patterns are translated into practical energy-saving tactics that don't compromise service quality.
Additionally, ML provides insights that support operational decisions that are in line with energy-saving objectives, thereby informing decision-making in resource management. The application of machine learning (ML) algorithms in a Green Service Allocator (GSA) demonstrates their usefulness in managing cooperative applications and rationally scheduling tasks to balance user requirements with energy conservation and quality of service (QoS) goals [89].
Machine learning includes the deep learning algorithms (*Figure 42*), which will be described in the next section.

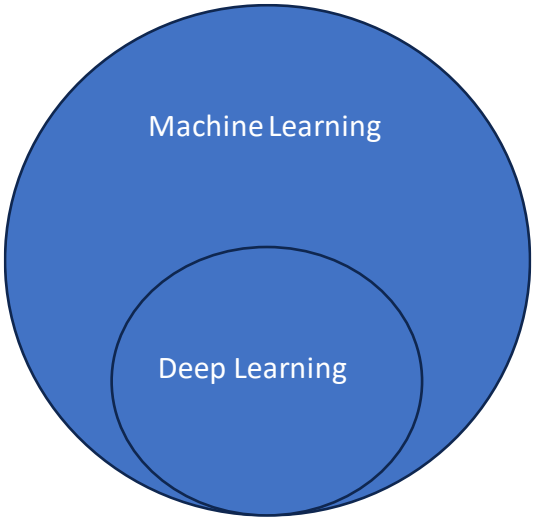


Figure 42. Visual representation of Machine Learning group, adapted from [89], [91]

Deep Learning

Deep Learning (DL), a subset of machine learning with complex neural networks, significantly enhances green cloud computing by optimizing energy efficiency in resource management. It displays an advanced strategy for sustainable operations by managing workloads to lower energy consumption [89]. Large dataset processing by DL makes it possible to make precise predictions about cloud resource requirements. Energy-saving measures like turning off idle servers and optimizing resource allocation based on actual demand are made easier by this predictive capability. DL models predict usage peaks and troughs by evaluating historical data, encouraging effective resource distribution and reducing energy waste [89].

A key application of DL in cloud computing is load balancing, ensuring even workload distribution across servers to prevent overloads that increase energy use and decrease performance. DL optimizes task allocation among virtual machines, balancing operational demand with energy consumption, thus boosting cloud efficiency.

An example of DL's application is DeepMind's neural network predicting the energy output of Google's wind fleet farm. Trained on historical data and weather forecasts, it demonstrates DL's capacity to utilize diverse data for accurate real-time predictions, optimizing energy production and minimizing waste [89].

While machine learning processes structured data effectively, DL excels in analyzing complex or unstructured data, making it ideal for tasks requiring extensive data pattern analysis. DL's advanced predictive analytics and autonomous learning from data without explicit programming distinguish it as a powerful tool for sustainable and efficient cloud resource management. *Figure 43* shows the difference between ML and DL which lies in the number of hidden layers for learning.

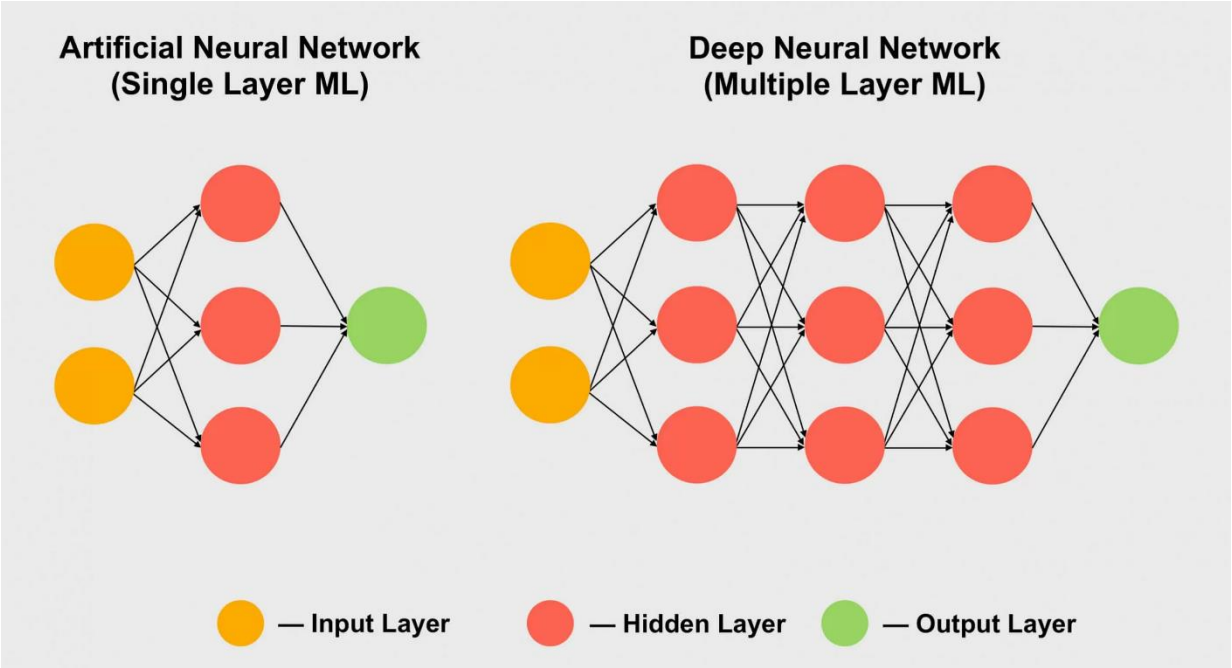


Figure 43. Machine learning and deep learning differences representation, adapted from [91]

Dynamic Load Balancing with Prediction Algorithm

According to the source [34], Dynamic Load Balancing (DLB) in cloud computing is a systematic approach designed to efficiently distribute cloud resources (such as CPU, RAM, Storage, and Bandwidth) among virtual machines (VMs) to ensure optimal resource utilization and prevent overloading. This technique is used to address the issue of overloading occurs when a host experiences resource shortages due to high workloads, often exacerbated by the demands of federated learning tasks. The DLB model uses VM migration and consolidation as core techniques to address dynamic overloading issues by moving VMs from oversubscribed hosts to others until the load is balanced. This approach is particularly effective in handling sudden spikes in workload, which can lead to instant overloading and affect cloud service reliability and performance [34].
The proposed dynamic load balancing model (DLBM) includes a spike detection algorithm that proactively identifies potential workload spikes before they occur. In order to do this, the algorithm employs both machine learning and statistical methodologies. This enables the system to preemptively rebalance resources, thus avoiding host overloading and minimizing the occurrence of service level agreement (SLA) violations. The model employs a systematic process that begins with the continuous monitoring of resource utilization across VMs and hosts, using a monitoring system deployed on the hypervisor. This system periodically collects data on resource consumption, which is then used to assess the load status of each host and make informed decisions about resource allocation, VM migration, and consolidation [34].

The main steps to apply this strategy are [34]:

- VM Migration and Consolidation Process in which the Identification of Oversubscribed Hosts (OSH) happens. A host is considered oversubscribed (OSH) when the cumulative resource demand from its VMs exceeds the host's upper utilization threshold. This condition indicates that the host is under pressure and potentially overloading, which could lead to degraded performance and service availability issues.

- The total resource requesting value for a host is calculated by summing up the demands of all VMs on that host.

- For the migration to be effective, it's essential to identify underutilized hosts that can accommodate additional VMs without becoming oversubscribed themselves. This process involves analyzing hosts' current load and their capacity to take on more work without compromising performance.

- Once suitable target hosts are identified, the migration process involves moving VMs from the overloaded host to these targets. This operation must be carefully managed to minimize downtime and ensure that the migrated services resume normal operation as swiftly as possible.

The DLBM's proactive approach represents a significant advancement in cloud computing, offering a solution to the limitations of previous models in managing sudden workload spikes. Through experimental analysis using the CloudSim toolkit (which is an available software to simulate cloud computing frameworks and services [6]) and a real-world workload dataset, the DLBM has been shown to effectively identify workload spikes, control recurrent migrations, and enhance the overall stability and efficiency of cloud services [34].
The benefits of DLB are several and takes its role in enhancing resource utilization efficiency, improving service availability, and reducing the likelihood of service level agreement violations. By dynamically adjusting to workload changes, DLB enables cloud providers to maintain a balanced distribution of computing tasks. This results in optimized performance, lower energy consumption, and higher service quality, effectively mitigating the challenges posed by unpredictable workload fluctuations and preventing overloaded servers [34].

The following section describes techniques belonging to the last category presented in the scheme of *Figure 37*, that is the statistical methods.

Regression and other statistical methods

In the context of optimizing energy consumption in cloud data centers, various statistical methods including regression analysis have been employed to enhance virtual machine (VM) management for energy efficiency. These methods play an important role in forecasting, decision-making, and operational optimization, contributing significantly to reducing the overall energy footprint of cloud infrastructure [6].
Statistical methods, such as mean, standard deviation, regression analysis, and others, are utilized extensively for VM management tasks like detecting overloaded and underloaded hosts, predicting resource requirements, and optimizing VM allocation, migration, and placement. These approaches make it possible to create models that can predict cloud data center states with high accuracy, which allows for the implementation of preventive measures that improve energy efficiency. One of the main statistical techniques that is highlighted is linear regression, which is used to predict the hosts' future CPU utilization. Using this approach, the expected future CPU utilization (dependent variable) and the current CPU utilization (independent variable) are related mathematically. Cloud data centers can prevent situations

of underutilization or overutilization, both of which are harmful to energy efficiency, by accurately projecting future resource demands [6].
Regression and other statistical techniques are used to optimize energy consumption in cloud data centers, bringing with them a number of significant benefits that when taken as a whole lead to more sustainable and effective operations. First and foremost, these techniques' capacity for prediction is noteworthy as a fundamental component of energy optimization. Data centers can make proactive adjustments by accurately forecasting resource utilization and future states of system operations. By facilitating prompt decisions that avoid resource waste and guarantee that the infrastructure is used as efficiently as possible, this foresight improves efficiency [6].
Moreover, resource optimization is a direct result of these methodologies being applied skillfully. The key to attaining energy efficiency is precisely matching the supply of physical resources with the demand for virtual machines (VMs). Regression modeling and other statistical instruments can be used by data centers to drastically reduce idle resources. This optimization minimizes wasteful energy use since resources are used and distributed in a way that closely matches actual demand, preventing both overuse and underuse [6].

Another advantage of using statistical models in this field is their adaptability. Because of the dynamic nature of cloud computing, which involves shifting operational conditions and varying demands, accurate models must also be flexible. These models can be updated and improved upon in response to new data, which promotes a culture of continuous improvement. By utilizing new insights and adjusting to new challenges, this iterative process guarantees that energy optimization strategies stay effective over time and improve performance [6].
In conclusion, the utilization of statistical analysis and regression has resulted in measurable enhancements in energy efficiency. Not only are these benefits theoretical, but they have also been proven to be effective through significant energy consumption reductions and improved quality of service (QoS). Through the methodical application of these techniques, data centers have demonstrated notable progress in energy utilization, underscoring the concrete advantages of integrating statistical methods into energy management strategies [6].
Despite the benefits, challenges such as the need for extensive historical data, model complexity, and the balance between model accuracy and computational overhead must be addressed. Additionally, the dynamic nature of cloud computing requires these models to be highly adaptable and scalable [6].

In the next section there is the description of a solution not belonging to previous categories, because of the type of functioning of the approach that directly affect the hardware.

Dynamic Voltage and Frequency Scaling

One important energy-saving technology that has been highlighted for its uses in data centers is dynamic voltage and frequency scaling, or DVFS. According to the source [30], the DVFS optimizes energy consumption by varying the CPU's operating frequency and voltage in accordance with the computational demands of the application. This method efficiently reduces energy consumption by taking advantage of the relationship between circuit power consumption and the square of both voltage and frequency [30]. This technique does not fall into any of the previously listed categories because it works directly with hardware.

The literature [30] put in evidence this methodology as a crucial optimization technology because, due to its significant reduction in power consumption, it has demonstrated great promise in cluster computing within data centers, saving businesses a significant amount of money. Numerous DVFS-based energy-saving strategies, including the Lowest-DVFS, δ-Advanced-DVFS, and Adaptive-DVFS, have been developed and evaluated. The mentioned methodologies are different in the way they function. For instance, the Lowest-DVFS operates at the lowest speed necessary for tasks, weighing energy savings against performance losses. In contrast, Adaptive-DVFS dynamically modifies CPU parameters in response to workload demands, providing greater adaptability and energy savings without significantly sacrificing Quality of Service (QoS) [30].
Furthermore, by modifying computation frequencies and carrying out sensible task scheduling, task scheduling algorithms based on DVFS have been developed to enhance system resource utilization. For example, by optimizing task deployment and frequency adjustment, data center management and request scheduling algorithms that use DVFS seek to minimize power costs while taking into account the geographic distribution of data centers and variations in electricity prices [30].

A particularly creative method that is highlighted in the source [30] is the online scheduler CPU MISER system, which makes use of DVFS technology. In order to dynamically modify the target frequency of processors based on load predictions, CPU MISER is made up of a DVFS scheduler, a workload predictor, and a performance monitor. This combination of components greatly reduces energy consumption [30].
The same source [30] proposes a formulation of a DVFS scheduling problem that guarantees the completion of a particular task within a given performance loss constraint and minimizes total energy consumption is also discussed. To capture the relationship between workload, frequency, and performance loss as a result of frequency scaling, complex modeling is required [30].
The efficiency of DVFS as an energy-saving technique in data centers is demonstrated by these studies and the technological solutions built upon it. They show that even though DVFS can drastically lower server energy consumption, more investigation is required to solve issues like memory consumption and communication overhead in heterogeneous environments, guaranteeing the wider application and optimization of DVFS in actual data center operations [30].
The following section addresses the issue of cooling system with the usage of liquid cooling systems.

Liquid Cooling

One solution to address the high energy consumption of cooling systems in data centers is liquid cooling [30]. This approach is divided into direct and indirect liquid cooling technologies:

- Direct Liquid Cooling: This entails the electronic components and a liquid coolant coming into direct contact. Heat is directly absorbed by the coolant and then expelled from the system by the components. The two further categories of direct liquid cooling are immersion and spray cooling methods. Immersion cooling, which involves fully submerging components in a non-conductive liquid, has drawn significant interest from the industry due to its exceptional heat dissipation efficiency [30].

- Indirect Liquid Cooling: it ensures that the cooling liquid does not make direct contact with the electronic components, in contrast to direct liquid cooling. Rather, heat is transmitted to the coolant via a secondary medium, like a radiator or heat exchanger (*Figure 44*). Since there is no direct contact between the coolant and electronics with this method, it can be used with coolants that have high heat transfer capacities, which reduces power consumption and increases safety [30]. *Figure 44* shows an illustration of what just mentioned.

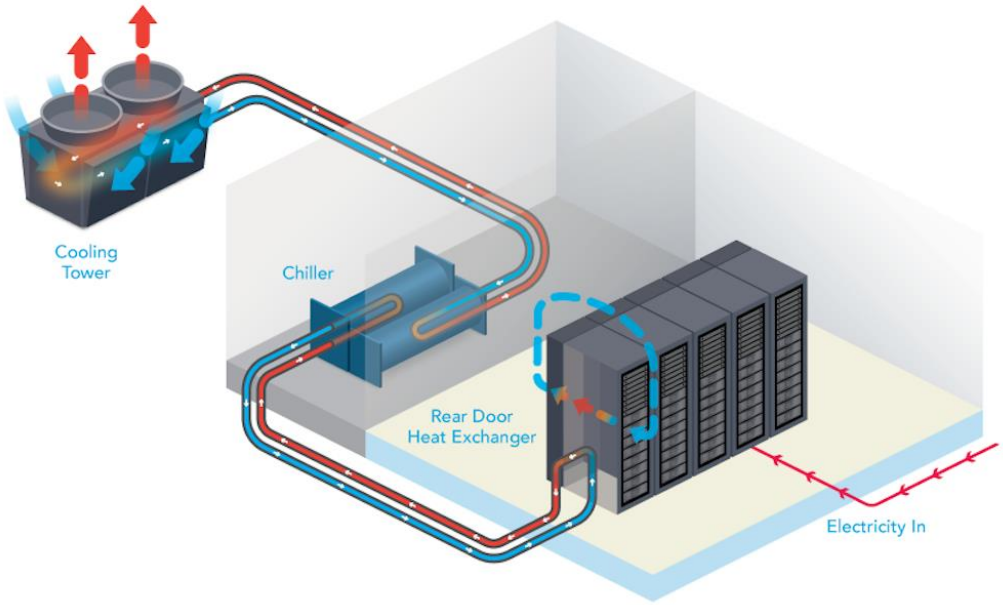


Figure 44. Illustration of indirect cooling liquid, adapted from [92]

Some considerations regarding this solution are that, despite its efficiency, liquid cooling's deployment and maintenance are more complex than traditional air cooling, particularly for indirect liquid cooling which requires locked compartments and complex systems.
Furthermore, although liquid cooling's high efficiency makes it a desirable option for high-performance computing clusters, adoption of the technology is hampered by the complexity of system maintenance and the absence of established procedures [30].

4.7. Energy Savings Quantification

After implementing one or more corrective solutions to address energy inefficiencies, it is crucial to closely monitor energy usage to evaluate the impact of these solutions (first step of the procedure). This monitoring process involves collecting data on energy consumption before and after the implementation of the solutions. By comparing these datasets, one can quantify the improvements made and assess the effectiveness of the corrective actions taken.

The purpose of this monitoring is not only to measure the immediate benefits but also to identify whether the implemented solutions have fully addressed the underlying issues. If the monitoring reveals that energy inefficiencies persist or if the improvements are not as significant as expected, this indicates that there may be additional root causes that were not initially identified.

In such cases, it is necessary to reiterate the process: analyze the new data, identify any remaining or new inefficiencies, and then implement further corrective solutions. This cycle of implementation, monitoring, and reassessment should continue until the monitoring data confirms that all identified issues have been effectively resolved and energy performance has been optimized.

5. Case Studies

In this section, some hypothetical case studies will be presented in order to show the application of the framework presented in chapter 4.
These cases have been created with data and insights taken from the literature, but they are not gathered from real case, however the aim to simulate possible realistic situations.
Firstly, a general description of the procedure flow through the use of the following flowchart (Figure 43).

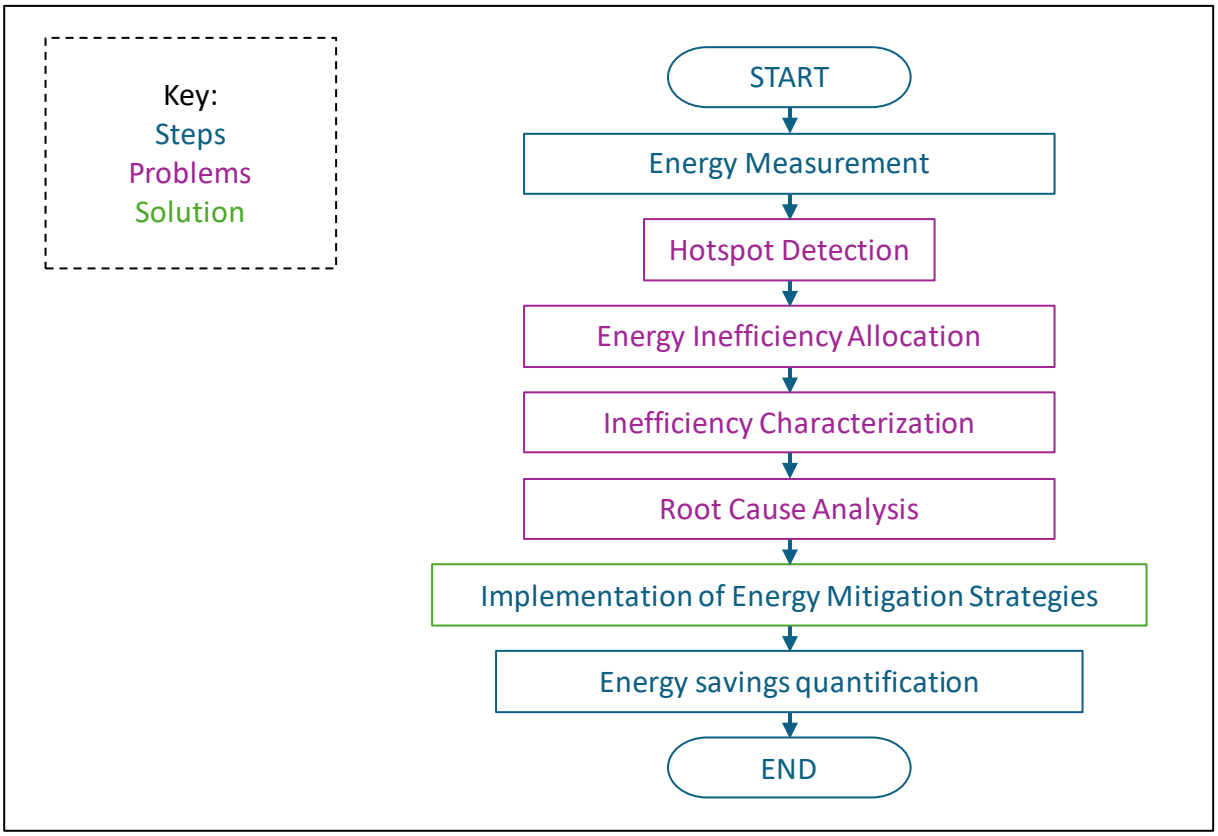


Figure 45. General Framework steps for energy optimization in cloud manufacturing

5.1. First Case Study: overloaded server

The first case scenario regards the study of a specific server: *IBM X3550M3, Figure 46.*

Figure 46. IBM X3550M3 Server, adapted from [93]

The decision on this specific server depends on the specific energy consumption values got from literature [1], in this case, the server shows high energy values for overloading states.

The procedure starts by monitoring the energy consumption. In this case, for the sake of simplicity, each value is taken with a frequency of one value per hour and the monitoring period is five days:

- Frequency of data gather: 1 value per hour.
- Period: 5 days
- Energy Consumption unit of measurement: Watts

The first thing to do suggested by the framework is to measure the energy consumption.
Energy consumption in cloud manufacturing can come from different resources so the tracking is done to the overall system with different platforms.
In the framework different tools are described to provide an energy measurement.
In this case, the monitoring phase is done with the first tool shown, the DEM system (Distributed Energy Meter), as highlighted in chapter 4 [19].
This tool is able to estimate precisely power consumption of CPUs, memory and disk using a multi-component power consumption model.
The first step is the energy monitoring from the slave nodes regarding the cloud server which are sent to the master node for aggregation. At the same time, the system gathers data regarding the resources utilization such as CPU, disk and memory is. For the sake of simplicity, the case study presents only the tracking on CPU. When all this data has been got, a mathematical relationship defines the energy consumption of the hardware component based on their utilization level [19].
Table 4 shows the server's power consumption values taken in the monitoring phase.
Figure 47 shows a graph with the corresponding pattern.
Table 5 shows the CPU's utilization level during the same time period.
Figure 48 shows the trend of the CPU usage.

Data related to energy consumption and CPU utilization have been synthetically generated to cover relevant scenario for proof of concept purposes from source [1].

Table 4. Power Consumption Monitoring Phase

Time (h)	Day 1	Day 2	Day 3	Day 4	Day 5
1	110	140	170	200	208
2	110	150	170	200	208
3	110	150	180	200	208
4	110	150	180	190	208
5	110	150	180	190	208
6	110	150	180	190	208
7	110	150	180	190	215
8	110	150	180	190	215
9	110	150	180	190	215
10	110	160	180	190	215
11	118	130	180	190	215
12	118	130	180	190	215
13	118	130	180	190	215
14	118	130	180	190	215
15	118	130	180	190	215
16	130	130	180	208	215
17	130	130	200	208	215
18	130	130	200	208	215
19	130	140	200	208	215
20	130	130	200	208	215
21	130	130	200	208	215
22	130	130	200	208	215
23	130	130	200	208	215
24	150	170	200	208	215

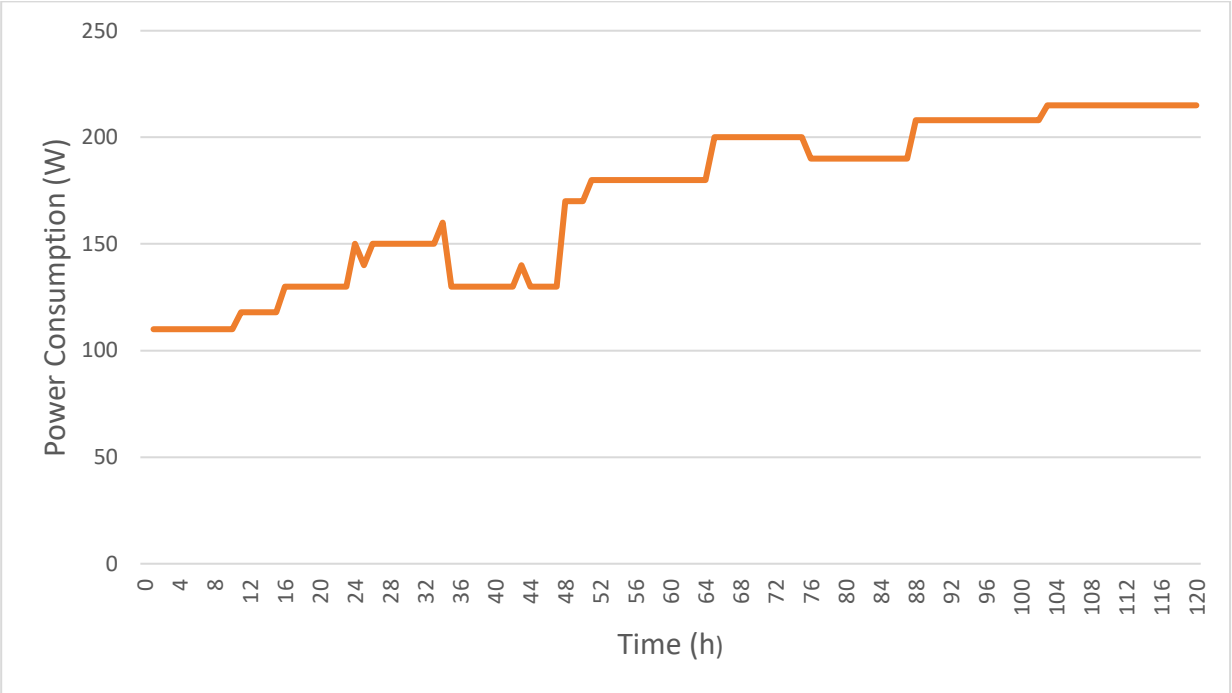


Figure 47. Power Tracking Trend on server

The following table shows values which refer to CPU usage in % during the same time period.

Table 5. CPU's utilization levels

Time (h)	Day 1	Day 2	Day 3	Day 4	Day 5
1	30%	50%	70%	90%	100%
2	30%	50%	70%	90%	100%
3	30%	50%	80%	90%	100%
4	30%	50%	80%	90%	100%
5	30%	50%	80%	90%	100%
6	30%	50%	80%	90%	100%
7	30%	50%	80%	90%	100%
8	30%	50%	80%	90%	100%
9	30%	50%	80%	90%	100%
10	30%	70%	80%	90%	100%
11	30%	50%	80%	90%	100%
12	30%	50%	80%	90%	100%
13	30%	50%	80%	90%	100%
14	30%	50%	80%	90%	100%
15	30%	50%	80%	90%	100%
16	50%	50%	80%	100%	100%
17	50%	50%	90%	100%	100%
18	50%	50%	90%	100%	100%
19	50%	50%	90%	100%	100%
20	50%	50%	90%	100%	100%
21	50%	50%	90%	100%	100%
22	50%	50%	90%	100%	100%
23	50%	50%	90%	100%	100%
24	60%	70%	90%	100%	100%

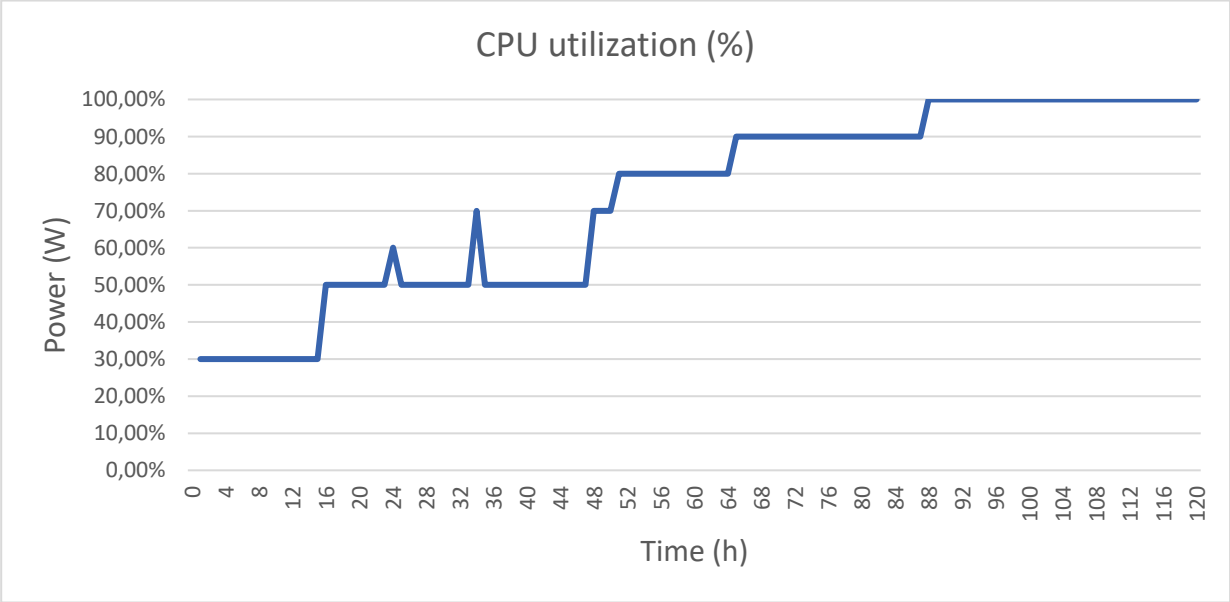


Figure 48. CPU's utilization trend

From the graph in *Figure 45*, it is possible to see that the energy usage of the server increases and stabilizes at the end of the tracking time period. Looking at the CPU's usage level in *Figure 48*, it is possible to notice a similar behavior.
Thanks to the DEM tool [1], it is possible to find the energy allocated to different hardware components.

The next phase is the hotspot identification:
According to section 4.2, by applying the algorithm that dynamically decides threshold and find the server state, it is possible to say that the server is in overload state [34]. This is understandable because, as defined in section 4.2, the server is not capable to accept any other workload due to the fact that the CPU's level is at maximum.
Indeed, to understand if the server is overloaded, the control range of CPU's usage level is from 90% to 100%, so according to the graph of CPU trend, it is possible to notice the CPU level is at 90 % since the hour 66 until reaching the maximum level. It is possible to say the server is in overload.

First scenario:
The third phase is the energy inefficiency allocation:
During this phase, there is the investigation on which program and tasks the CPU is working on, and it is possible to define the type of workload. Thanks to this type of tools capable of checking the state of programs simultaneously with the CPU's usage level, it is possible to say that the SCADA system (information gathering) and MES (interface between IT and OT technologies) are running and by looking at the different tasks done, it is possible to understand which type of workload is loading. In this case, the data transmission and data alignment are the activities requiring computational resources. Thanks to the first phase in which it was assessed how much energy is allocated to the CPU, it is possible to link the energy consumption from the CPU to the programs and tasks occurred.

The fourth step is the energy Inefficiency Characterization (analysis on production data): this phase tries to find what is the inefficiency caused by the SCADA and MES programs and data processing being so hardware and energy intensive. In this case, through analysis made on production data, it is possible to see that the inefficiency consists of a high number of dirty data present.

The fifth step is the Root Cause Analysis:
Though the use of different diagnostic methods such as 5W, C/E, Process control charts, it is possible to understand the root cause of the previously found inefficiency. In this case, the following causes are identified.
IT/OT Integration:

- OT designed with low upgrades (old systems)
- Compatibility problems in data transfer protocols and processing standards
- complex application of theoretical architectures model for the integration in real case
- proprietary solution of MES

The sixth step is the solution Implementation:
In order to address these challenges, some solutions are identified and applied, as such:

- Legacy System Upgrade for SCADA

- Standard protocols
- RAMI4.0 toolbox
- Industrial Business Process Twin, which acts as an intermediary between IT and OT, aiming to reduce the complexity of OT systems for IT stakeholders and vice-versa.

The final step is the Energy Monitoring:
This phase is important to understand and evaluate the positive impact of implementing the solutions, by going back to the first monitoring phase and then quantification of energy savings.

Second scenario:
The same case of overloaded server is taken.
The third phase is the energy inefficiency allocation:
During this phase, there is the investigation on which program and tasks the CPU is working on, and it is possible to define the type of workload. Thanks to this type of tools capable of checking the state of programs simultaneously with the CPU's usage level, it is possible to say that the Database Management system (information manager) and MES (interface between IT and OT technologies) are running and by looking at the different tasks done, it is possible to understand which type of workload is loading. In this case, the data processing is the activity requiring computational resources. Thanks to the first phase in which it was assessed how much energy is allocated to the CPU, it is possible to link the energy consumption from the CPU to the programs and tasks occurred.

The fourth step is the energy Inefficiency Characterization (analysis on production data): this phase tries to find what is the inefficiency causing the program and data processing being so hardware and energy intensive. In this case, through data analysis made on production data coming from IoT, it is possible to see that the concerning activity is due to the high quantity of data gathered and elaborated all the time, that is expressed in the framework as excessive data.
The fifth step is the Root Cause Analysis:
Though the use of different diagnostic methods such as 5W, C/E, Process control charts and through the continuation of previous data analysis, it is possible to understand the root cause of the previously found inefficiency. In this case, the excessive data elaborated is the inefficiency, so it is plausible that the root cause is the sheer volume and variety of data gathered that needs a lot of computational resources to be analyzed and elaborated. The issue is that all this volume of data could be not necessary, as previously said and stated in section 4, it is excessive respect to the usefulness they provide.

The sixth step is the implementation of the corresponding solution:
In this case, it possible to measure the density of the dataset and the effect of data reduction on classification accuracy. This solution introduces and uses two new metrics for evaluating big datasets in classification problems: Neighborhood Density (ND) and Decision Tree Progression (DTP). When deciding whether to use the full dataset for machine learning tasks or to preprocess the data, these metrics offer insightful information about the complexity and redundancy of the data. ND assists in determining the level of redundancy in the data by analyzing how density changes as the dataset is reduced. It may be possible to reduce the dataset size without losing important information if there is a slight change in ND following the significant portion of the data being discarded. This suggests that the dataset has a high redundancy.

DTP is used to determine whether a dataset has redundant data that has little bearing on the model's ability to predict the future. The dataset can be reduced without significantly impairing the model's performance if there is a slight variation in accuracy, which indicates that a large portion of the data may be redundant [55].

In the following image (*Figure 49*), a schematic summary of what presented in this first case.

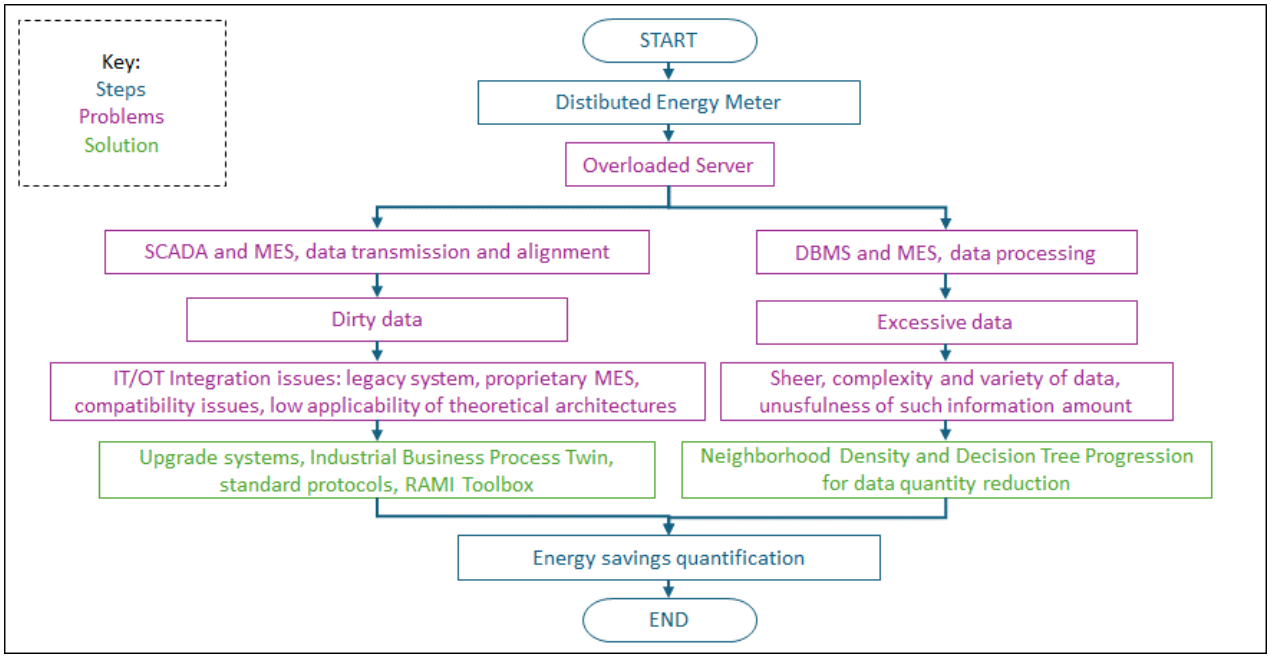


Figure 49. Framework application of overloaded server case study

5.2. Case Study: Idle server

The first case scenario regards the study of a specific server: HP ProLiant G3 (*Figure 50*).

Figure 50. HP ProLiant G3, adopted from resource [94]

The decision on this specific server depends on its energy consumption values, this server shows high values during idle times [1].

The procedure starts by monitoring the energy consumption. In this case, for the sake of simplicity, each value is taken with a frequency of one value per hour and the monitoring period is five days:

- Frequency of data gather: 1 value per hour.
- Period: 5 days
- Power Consumption unit of measurement: Watts

The first phase is the energy measurement. In this case, it is used the second method presented in the framework, denoted as “A measurement-based approach” [20] which use a specific power analyzer, the Voltech PM1000+, that is connected to the server and its power supply in order to monitor the server's power usage. This analyzer takes value every second, but for the sake of simplicity in the case study there a value per hour, and tracks the total power used by the server in real time. At the same time, there is the monitoring of how much of the server's hardware, that are the network, disk, and CPU, are in use at any given moment. Then, it is firstly determined the energy contribution from the CPU and then evaluate the distinct contributions from each part to the server's overall energy usage [20].

Data related to energy consumption and CPU utilization have been synthetically generated to cover relevant scenario for proof-of-concept purposes from source [1].

Figure 51. Voltech PM1000+ power analyzer, adapted from [95]

Table 5. Power Consumption Monitoring Phase

Time (h)	Day 1	Day 2	Day 3	Day 4	Day 5
1	100	108	103	102	104
2	100	108	103	102	104
3	100	108	103	102	104
4	100	108	103	102	104
5	100	108	110	102	104
6	100	103	110	102	104
7	100	103	110	104	104
8	105	103	110	104	104
9	105	103	110	104	104
10	105	103	106	104	104
11	105	103	106	104	110
12	105	103	106	104	110
13	105	103	106	104	110
14	105	103	106	104	110
15	105	103	106	104	110
16	105	103	106	104	110
17	105	103	106	104	105
18	108	103	106	104	105
19	108	103	106	104	105
20	108	103	102	103	105
21	108	103	102	107	105
22	108	103	102	104	105
23	108	103	102	104	105
24	108	103	102	104	105

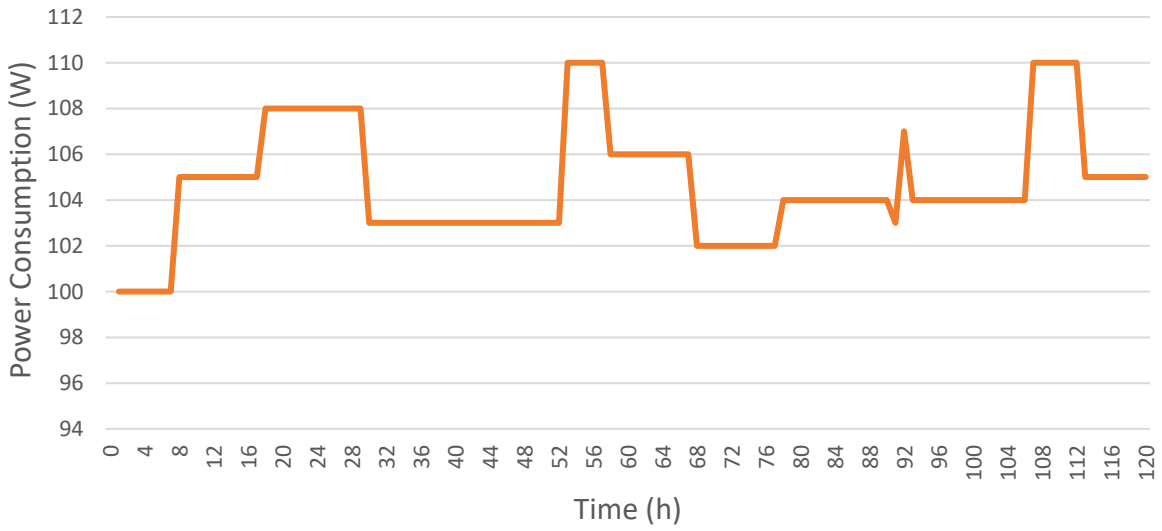


Figure 52. Power consumption pattern over time

Table 6. CPU's utilization level

Time (h)	Day 1	Day 2	Day 3	Day 4	Day 5
1	0%	3.6 %	2.5 %	2%	2.9%
2	0%	3.6 %	2.5 %	2%	2.9%
3	0%	3.6 %	2.5 %	2%	2.9%
4	0%	3.6 %	2.5 %	2%	2.9%
5	0%	3.6 %	8%	2%	2.9%
6	0%	2.5 %	8%	2.7%	2.9%
7	0%	2.5 %	8%	2.7%	2.9%
8	3%	2.5 %	8%	2.7%	2.9%
9	3%	2.5 %	8%	2.7%	2.9%
10	3%	2.5 %	3.2 %	2.7%	2.9%
11	3%	2.5 %	3.2 %	2.7%	8%
12	3%	2.5 %	3.2 %	2.7%	8%
13	3%	2.5 %	3.2 %	2.7%	8%
14	3%	2.5 %	3.2 %	2.7%	8%
15	3%	2.5 %	3.2 %	2.7%	8%
16	3%	2.5 %	3.2 %	2.7%	8%
17	3%	2.5 %	3.2 %	2.7%	3%
18	3.6 %	2.5 %	3.2 %	2.7%	3%
19	3.6 %	2.5 %	3.2 %	2.5%	3%
20	3.6 %	2.5 %	2%	4%	3%
21	3.6 %	2.5 %	2%	2.9%	3%
22	3.6 %	2.5 %	2%	2.9%	3%
23	3.6 %	2.5 %	2%	2.9%	3%
24	3.6 %	2.5 %	2%	2.9%	3%

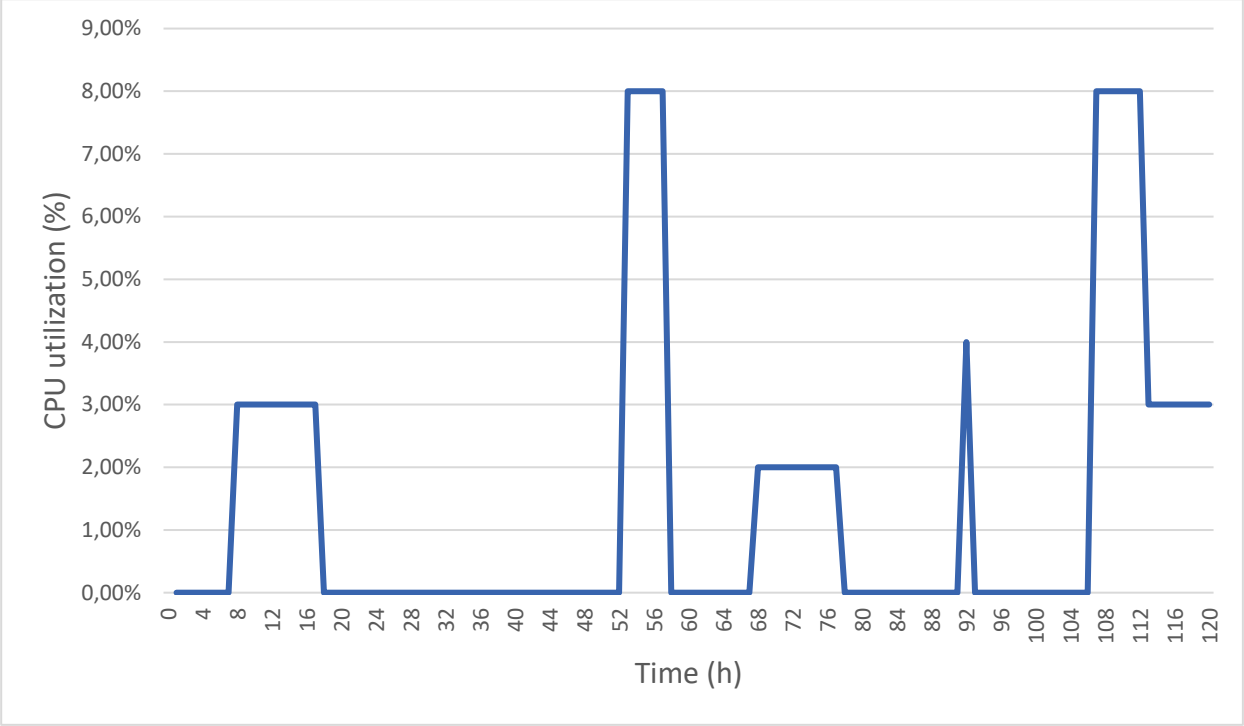


Figure 51. CPU's utilization level trend over time

As it is possible to see, the energy consumption of the server is quite high, while the CPU's usage is most of the time low or almost 0.

The next phase is the hotspot detection. In this case, the behavior of server is easily understandable to be in idle state, because the CPU is rarely used and for a low portion. This means the server is being powered all time at the maximum capacity for this state, even if it is not used, which remarks an inefficient energy usage.

The next phase is the energy inefficiency allocation to activities. In this case, it is still possible to use the profiling tool to understand the tasks in waiting or running, but the result is that almost no activity has been run for this period of time in the server. So, it is not possible to detect a program or a task, of course.

The next phase is the inefficiency characterization. This phase tries to claim how to quantify the issue and a data analysis should be done. In this case, it is easy to understand that the inefficiency is the low frequency of requests: tasks do not arrive to this server.

After this, the root cause analysis phase comes.
In this phase, using detection tools like 5 Whys and cause-effect, it is possible to understand that overprovisioning occurred. Because overprovisioning is needed to the cloud servers for urgent reasons or to cope with high and sudden computing demands, the server cannot be shut down.

The next and final step is the implementation of the corresponding solution. According to [35], the Dynamic Time Scale based server Provision could be applied to solve this problem, because this approach still provides overprovisioning to cover peak of demands and highly variable

workload, by keeping the same or better quality of service, and increasing the energy efficiency of the server. This is done through the ability of this approach to adjust the resources when needed and putting the server in a low power mode when idle. This methodology demonstrates an increase in energy efficiency of 60% in term of Joule/m^2.

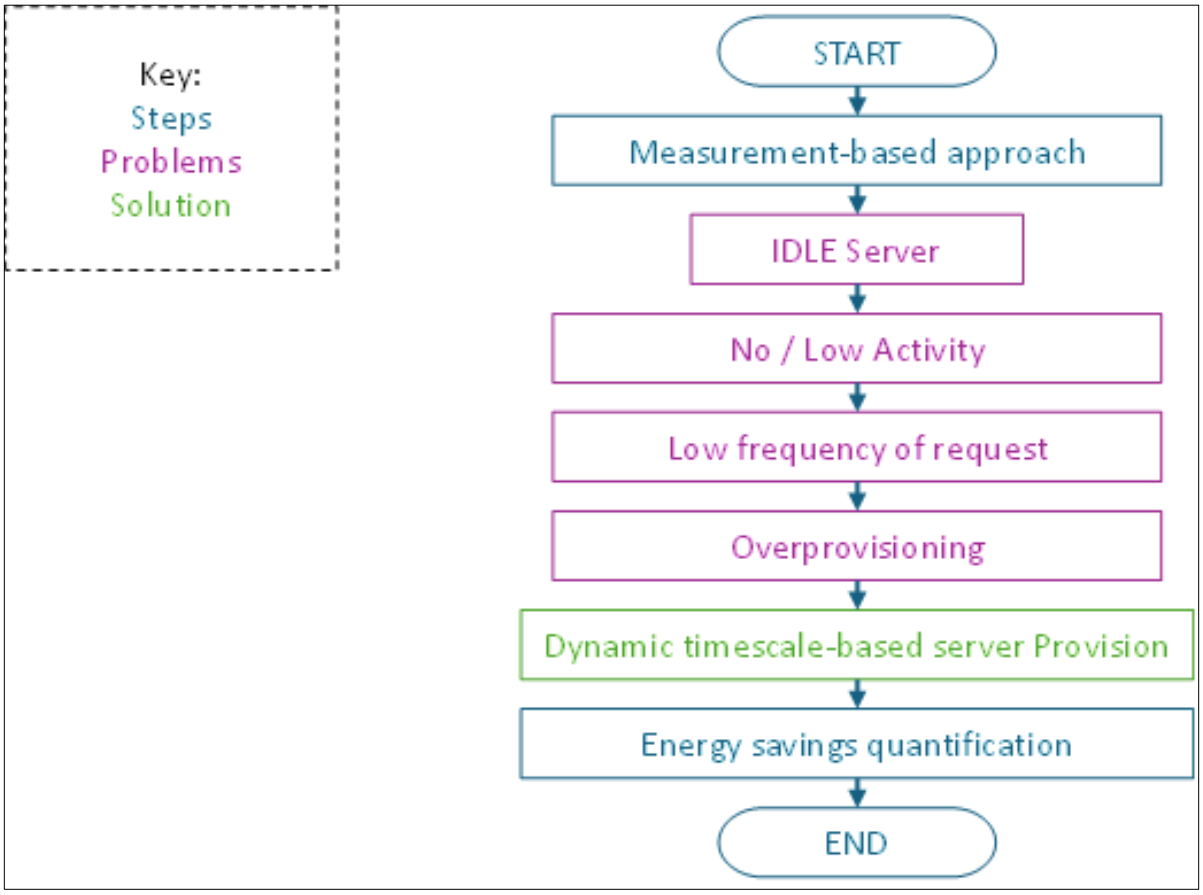


Figure 52. Framework application on idle server case study

6. Conclusions

The energetically demanding nature of cloud data centers, together with the increasing trend among corporations to migrate towards cloud computing systems, has significantly guided this research into exploring the cloud manufacturing landscape. This thesis is centrally focused on devising a framework aimed at optimizing energy consumption within the context of cloud manufacturing, recognizing the critical need for sustainable practices in this evolving field.

The framework introduced in this study marks a crucial advancement in bridging the existent knowledge gap highlighted in literature concerning the analysis of energy consumption dynamics in cloud manufacturing settings. It provides a holistic view by employing essential tools and methodologies together, designed to analyze and thoroughly comprehend the origins of energy inefficiency. This effort is not merely academic but serves a practical tool for future implementations.

Through the application of varied energy measurement methodologies, this research account for the consumption patterns associated with different hardware components of servers, establishing correlations that pinpoint energy-intensive states. The utilization of dynamic algorithms facilitates this process, enabling a nuanced understanding of energy consumption behaviors. Following the identification of high-energy states, the framework guides an exploration into the distribution of energy consumption across various programs and tools over time. This help in identifying energy-intensive operations, setting the stage for a more detailed analysis of inefficiencies, which is further assisted by production data analytics and the monitoring of specific variables aimed at quantifying these inefficiencies.

The procedural steps of the framework give the basis for the root cause analysis phase, which aims to identify the fundamental causes of the inefficiencies that are seen in previous steps. Identifying the root cause is crucial as it directly informs the implementation of targeted solutions designed to enhance both operational and energy efficiency within cloud manufacturing systems. The cyclic nature of the framework, characterized by a subsequent phase of implementation energy measurement, is crucial for evaluating the efficacy of the deployed solutions and quantifying the realized energy savings. This iterative approach underscores the framework's foundational principle of perpetual enhancement.

The practical applicability and robustness of the proposed framework have been validated through its deployment in two hypothetical case studies. These applications demonstrate the framework's potential to support more environmentally and energy-efficient operations in cloud manufacturing environments, while also reaffirming its adaptability and resilience respect to different possible situations.

In essence, this framework does not merely fill a void in existing literature; it provides the way for sustainable practices in cloud manufacturing. By addressing critical energy inefficiencies and offering a systematic approach for their mitigation, this research contributes to the broader vision of fostering a more sustainable and environmentally responsible industrial landscape, aiming to reduce industry carbon footprint and utility cost.
Future explorations and advancements in energy optimization techniques are essential for realizing the full potential of cloud manufacturing as a paradigm of sustainable innovatio

References

[1] A. Katal, S. Dahiya, and T. Choudhury, “Energy efficiency in cloud computing data centers: a survey on software technologies,” *Cluster Comput*, vol. 26, no. 3, pp. 1845–1875, Jun. 2023, doi: 10.1007/s10586-022-03713-0.

[2] M. Jumde and S. Dongre, “Analysis on energy efficient green cloud computing,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jun. 2021. doi: 10.1088/1742-6596/1913/1/012100.

[3] V. Neves, M. Pit, R. Reiser, A. Yamin, and M. Pilla, “Samsara architecture: Exploring situation awareness in cloud computing management,” *Sustainable Computing: Informatics and Systems*, vol. 29, Mar. 2021, doi: 10.1016/j.suscom.2020.100475.

[4] “https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks.”

[5] “https://cloud.google.com/learn/what-is-cloud-computing?hl=it .”

[6] S. S. Panwar, M. M. S. Rauthan, and V. Barthwal, “A systematic review on effective energy utilization management strategies in cloud data centers,” *Journal of Cloud Computing*, vol. 11, no. 1. Springer Science and Business Media Deutschland GmbH, Dec. 01, 2022. doi: 10.1186/s13677-022-00368-5.

[7] “https://www.educba.com/what-is-salesforce-technology/?source=leftnav .”

[8] N. Agarwal, “Green Cloud Computing: Carbon Emission Impact And Energy Efficiency”, [Online]. Available: www.ijstr.org

[9] “https://starship-knowledge.com/iaas-vs-paas-vs-saas.”

[10] N. Patel and H. Patel, “Energy efficient strategy for placement of virtual machines selected from underloaded servers in compute Cloud,” *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 6, pp. 700–708, Jul. 2020, doi: 10.1016/j.jksuci.2017.11.003.

[11] “https://marketbusinessnews.com/cloud-computing-deployment-models-an-overview-of-different-types/268425/.”

[12] “https://openclipart.org/detail/303483/virtualization.”

[13] A. Katal, T. Choudhury, and S. Dahiya, “Energy optimized container placement for cloud data centers: a meta-heuristic approach,” *Journal of Supercomputing*, vol. 80, no. 1, pp. 98–140, Jan. 2024, doi: 10.1007/s11227-023-05462-2.

[14] L. Haghnegahdar, S. S. Joshi, and N. B. Dahotre, “From IoT-based cloud manufacturing approach to intelligent additive manufacturing: industrial Internet of Things—an overview,” *International Journal of Advanced Manufacturing Technology*, vol. 119, no. 3–4. Springer Science and Business Media Deutschland GmbH, pp. 1461–1478, Mar. 01, 2022. doi: 10.1007/s00170-021-08436-x.

[15] S. Chiappa, E. Videla, V. Viana-Céspedes, P. Piñeyro, and D. A. Rossit, “Cloud manufacturing architectures: State-of-art, research challenges and platforms description,” *J Ind Inf Integr*, vol. 34, Aug. 2023, doi: 10.1016/j.jii.2023.100472.

[16] W. Wei, F. Zhou, and P. F. Liang, “Product platform architecture for cloud manufacturing,” *Adv Manuf*, vol. 8, no. 3, pp. 331–343, Sep. 2020, doi: 10.1007/s40436-020-00306-1.

[17] T. Lojka, M. Bundzel, and I. Zolotová, “Service-oriented Architecture and Cloud Manufacturing.”

[18] K. Xu *et al.*, “Advanced Data Collection and Analysis in Data-Driven Manufacturing Process,” *Chinese Journal of Mechanical Engineering (English Edition)*, vol. 33, no. 1. Springer, Dec. 01, 2020. doi: 10.1186/s10033-020-00459-x.

[19] W. Lin, H. Wang, Y. Zhang, D. Qi, J. Z. Wang, and V. Chang, “A cloud server energy consumption measurement system for heterogeneous cloud environments,” *Inf Sci (N Y)*, vol. 468, pp. 47–62, Nov. 2018, doi: 10.1016/j.ins.2018.08.032.

[20] J. Arjona, A. Chatzipapas, A. F. Anta, and V. Mancuso, “A Measurement-based Analysis of the Energy Consumption of Data Center Servers,” Feb. 2014, [Online]. Available: http://arxiv.org/abs/1402.0804

[21] G. Conti, D. Jimenez, A. del Rio, S. Castano-Solis, J. Serrano, and J. Fraile-Ardanuy, “A Multi-Port Hardware Energy Meter System for Data Centers and Server Farms Monitoring,” *Sensors*, vol. 23, no. 1, Jan. 2023, doi: 10.3390/s23010119.

[22] “https://www.vectornav.com/resources/inertial-navigation-primer/hardware/synccomm.”

[23] “https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/.”
[24] “https://www.phoronix.com/review/pts_50_release.”
[25] Z. Zhou, M. Shojafar, M. Alazab, and F. Li, “IECL: An Intelligent Energy Consumption Model for Cloud Manufacturing,” *IEEE Trans Industr Inform*, vol. 18, no. 12, pp. 8967–8976, Dec. 2022, doi: 10.1109/TII.2022.3165085.
[26] W. Lin, Y. Zhang, W. Wu, S. Fong, L. He, and J. Chang, “An adaptive workload-aware power consumption measuring method for servers in cloud data centers,” *Computing*, vol. 105, no. 3, pp. 515–538, Mar. 2023, doi: 10.1007/s00607-020-00819-4.
[27] N. Frigerio, A. Matta, and M. Rasella, “Energy monitoring of manufacturing plants: A real case application,” in *Procedia CIRP*, Elsevier B.V., 2022, pp. 770–775. doi: 10.1016/j.procir.2022.02.128.
[28] “https://www.record-evolution.de/en/blog/industrial-iot-platform-vs-iot-platform-mes-whats-the-difference/.”
[29] “https://nrel.gov/computational-science/measuring-efficiency-pue.html .”
[30] H. Cheng, B. Liu, W. Lin, Z. Ma, K. Li, and C. H. Hsu, “A survey of energy-saving technologies in cloud data centers,” *Journal of Supercomputing*, vol. 77, no. 11, pp. 13385–13420, Nov. 2021, doi: 10.1007/s11227-021-03805-5.
[31] Kirti, M. Smriti, K. Shaily, and D. Sudha, “Minimization of Energy Consumption in Cloud,” in *2023 3rd International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies, ICAECT 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ICAECT57570.2023.10117732.
[32] “ https://www-stage.avinetworks.com/glossary/server-overload/.”
[33] T. Voigt and P. Gunningberg, “Adaptive Resource-based Web Server Admission Control,” 2002.
[34] A. Patil and R. Patil, “Proactive and dynamic load balancing model for workload spike detection in cloud,” *Measurement: Sensors*, vol. 27, Jun. 2023, doi: 10.1016/j.measen.2023.100799.
[35] C. Hu, Y. Guo, Y. Deng, and L. Lang, “Improve the Energy Efficiency of Datacenters With the Awareness of Workload Variability,” *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1260–1273, Jun. 2022, doi: 10.1109/TNSM.2022.3144508.
[36] S. Koppula *et al.*, “EDEN: Enabling energy-efficient, high-performance deep neural network inference using approximate DRAM,” in *Proceedings of the Annual International Symposium on Microarchitecture, MICRO*, IEEE Computer Society, Oct. 2019, pp. 166–181. doi: 10.1145/3352460.3358280.
[37] “https://thecloudblog.net/lab/practical-top-down-resource-monitoring-of-a-kubernetes-cluster-with-metrics-server/.”
[38] A. Olgun *et al.*, “Sectored DRAM: A Practical Energy-Efficient and High-Performance Fine-Grained DRAM Architecture,” Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.13795
[39] “https://www.zerounoweb.it/cloud-computing/cloud-manufacturing-che-cose-i-vantaggi-e-le-tecniche/ .”
[40] “https://www.salesforce.com/ap/resources/articles/what-is-a-cloud-based-crm/.”
[41] “https://pro.arcgis.com/en/pro-app/latest/help/data/cad/what-is-cad-data.htm.”
[42] “https://www.techtarget.com/searchcustomerexperience/definition/CRM-customer-relationship-management .”
[43] Z. N. Jawad and V. Balázs, “Machine learning-driven optimization of enterprise resource planning (ERP) systems: a comprehensive review,” *Beni-Suef University Journal of Basic and Applied Sciences*, vol. 13, no. 1. Springer Science and Business Media Deutschland GmbH, Dec. 01, 2024. doi: 10.1186/s43088-023-00460-y.
[44] “https://safetyculture.com/topics/quality-management-system/.”
[45] “https://www.ibm.com/topics/mes-system.”
[46] S. Bonnaud, C. Didier, and A. Kohler, “Industry 4.0 and Cognitive Manufacturing Architecture Patterns, Use Cases and IBM Solutions.”
[47] “https://www.techtarget.com/searchdatamanagement/definition/database-management-system.”
[48] “https://www.databricks.com/glossary/data-analysis-platform.”

[49] Z. Bi, Y. Jin, P. Maropoulos, W. J. Zhang, and L. Wang, “Internet of things (IoT) and big data analytics (BDA) for digital manufacturing (DM),” *Int J Prod Res*, vol. 61, no. 12, pp. 4004–4021, 2023, doi: 10.1080/00207543.2021.1953181.
[50] A. Tsanousa *et al.*, “A Review of Multisensor Data Fusion Solutions in Smart Manufacturing: Systems and Trends,” *Sensors*, vol. 22, no. 5. MDPI, Mar. 01, 2022. doi: 10.3390/s22051734.
[51] S. Kang, E. Kim, J. Shim, S. Cho, W. Chang, and J. Kim, “Mining the relationship between production and customer service data for failure analysis of industrial products,” *Comput Ind Eng*, vol. 106, pp. 137–146, Apr. 2017, doi: 10.1016/j.cie.2017.01.028.
[52] D. Hölscher, T. Bayer, C. Reich, P. Ruf, and F. Gut, *A Big Data Quality Preprocessing and Domain Analysis Provisioner Framework using Cloud Infrastructures*. 2018. [Online]. Available: https://www.researchgate.net/publication/324918883
[53] “https://www.simplilearn.com/what-is-data-processing-article.”
[54] “https://www.techtarget.com/searchstorage/definition/redundant .”
[55] J. Maillo, I. Triguero, and F. Herrera, “Redundancy and Complexity Metrics for Big Data Classification: Towards Smart Data,” *IEEE Access*, vol. 8, pp. 87918–87928, 2020, doi: 10.1109/ACCESS.2020.2991800.
[56] “https://venturebeat.com/data-infrastructure/what-is-dirty-data-sources-impact-key-strategies/.”
[57] B. Ghit and D. Epema, “Reducing Job Slowdown Variability for Data-Intensive Workloads,” in *Proceedings - IEEE Computer Society’s Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, MASCOTS*, IEEE Computer Society, Nov. 2015, pp. 61–70. doi: 10.1109/MASCOTS.2015.24.
[58] “https://aws.amazon.com/it/what-is/latency/.”
[59] “https://www.lifewire.com/what-is-bandwidth-2625809.”
[60] “Use IT And OT Integration As A Lever For Digital Transformation.”
[61] “https://www.statology.org/jaccard-similarity/.”
[62] “https://www.datasciencecentral.com/dirty-data-quality-assessment-amp-cleaning-measures/.”
[63] “https://guidingmetrics.com/content/cloud-services-industrys-10-most-critical-metrics/.”
[64] “https://www.indeed.com/career-advice/career-development/response-time-testing.”
[65] M. S. Ahmedani, “Root Cause Analysis: A Quality Tool using 5 Whys”, doi: 10.13140/RG.2.2.24886.73288.
[66] “https://www.sologic.com/en-gb/about/root-cause-analysis.”
[67] “Error-Aware Data Clustering for In-Network Data Reduction in Wireless Sensor Networks M. K. Alam,* Azrina Abd Aziz,1 S. A. Latif,2 and Azlan Awang ”.
[68] Y. Zhang, D. Tang, H. Zhu, S. Zhou, and Z. Zhao, “An Efficient IIoT Gateway for Cloud–Edge Collaboration in Cloud Manufacturing,” *Machines*, vol. 10, no. 10, Oct. 2022, doi: 10.3390/machines10100850.
[69] T. Shwe and M. Aritsugi, “Optimizing Data Processing: A Comparative Study of Big Data Platforms in Edge, Fog, and Cloud Layers,” *Applied Sciences*, vol. 14, no. 1, p. 452, Jan. 2024, doi: 10.3390/app14010452.
[70] A. Barron, D. D. Sanchez-Gallegos, D. Carrizales-Espinoza, J. L. Gonzalez-Compean, and M. Morales-Sandoval, “On the Efficient Delivery and Storage of IoT Data in Edge–Fog–Cloud Environments,” *Sensors*, vol. 22, no. 18, Sep. 2022, doi: 10.3390/s22187016.
[71] O. Popoola and B. Pranggono, “On energy consumption of switch-centric data center networks,” *Journal of Supercomputing*, vol. 74, no. 1, pp. 334–369, Jan. 2018, doi: 10.1007/s11227-017-2132-5.
[72] S. K. Uzaman, A. U. R. Khan, J. Shuja, T. Maqsood, F. Rehman, and S. Mustafa, “A systems overview of commercial data centers: Initial energy and cost analysis,” *International Journal of Information Technology and Web Engineering*, vol. 14, no. 1, pp. 42–65, Jan. 2019, doi: 10.4018/IJITWE.2019010103.
[73] H. M. Al-Kadhim and H. S. Al-Raweshidy, “Energy-Efficient Traffic in Cloud-Based IoT,” *IEEE Sens J*, vol. 23, no. 22, pp. 28035–28043, Nov. 2023, doi: 10.1109/JSEN.2023.3323805.

[74] T. Wang, S. Guo, and C. G. Lee, “Manufacturing task semantic modeling and description in cloud manufacturing system,” *International Journal of Advanced Manufacturing Technology*, vol. 71, no. 9–12, pp. 2017–2031, 2014, doi: 10.1007/s00170-014-5607-z.
[75] T. Kampa, C. K. Müller, and D. Großmann, “Interlocking IT/OT security for edge cloud-enabled manufacturing,” *Ad Hoc Networks*, vol. 154. Elsevier B.V., Mar. 01, 2024. doi: 10.1016/j.adhoc.2023.103384.
[76] A. Garcia, X. Oregui, J. Franco, U. Arrieta, J. Ferreres, and J. A. Valencia, “Time Series Manufacturing Data Edge Monitoring and Visualization to Support Industrial Maintenance Teams,” *SN Comput Sci*, vol. 5, no. 1, Jan. 2024, doi: 10.1007/s42979-023-02442-4.
[77] G. Schäfer, H. Waclawek, S. Riedmann, C. Binder, C. Neureiter, and S. Huber, “IT/OT Integration by Design,” May 2023, [Online]. Available: http://arxiv.org/abs/2305.19735
[78] D. Palade, C. Moller, C. Li, and S. Mantravadi, “An Open Platform for Smart Production: IT/OT Integration in a Smart Factory,” in *International Conference on Enterprise Information Systems, ICEIS - Proceedings*, Science and Technology Publications, Lda, 2021, pp. 707–714. doi: 10.5220/0010436807070714.
[79] G. Murray, M. N. Johnstone, and C. Valli, “The convergence of IT and OT in critical infrastructure,” pp. 149–155, 2017, doi: 10.4225/75/5a84f7b595b4e.
[80] J.-Y. Nie, Institute of Electrical and Electronics Engineers, and IEEE Computer Society, *2017 IEEE International Conference on Big Data : proceedings : Dec 11- 14, 2017, Boston, MA, USA*.
[81] R. Sakamoto *et al.*, “Analyzing resource trade-offs in hardware overprovisioned supercomputers,” in *Proceedings - 2018 IEEE 32nd International Parallel and Distributed Processing Symposium, IPDPS 2018*, Institute of Electrical and Electronics Engineers Inc., Aug. 2018, pp. 526–535. doi: 10.1109/IPDPS.2018.00062.
[82] “https://insidehpc.com/2021/05/energy-efficiency-comparison-air-cooling-vs-liquid-cooling/.”
[83] S. Xu, H. Zhang, and Z. Wang, “Thermal Management and Energy Consumption in Air, Liquid, and Free Cooling Systems for Data Centers: A Review,” *Energies*, vol. 16, no. 3. MDPI, Feb. 01, 2023. doi: 10.3390/en16031279.
[84] M. S. U. Din, M. A. U. Rehman, R. Ullah, C. W. Park, and B. S. Kim, “Towards network lifetime enhancement of resource constrained iot devices in heterogeneous wireless sensor networks,” *Sensors (Switzerland)*, vol. 20, no. 15, pp. 1–23, Aug. 2020, doi: 10.3390/s20154156.
[85] “https://www.techtarget.com/iotagenda/feature/Fog-nodes-simplify-edge-vs-cloud-computing-choice.”
[86] “https://www.cs.cornell.edu/courses/cs5413/2014fa/lectures/08-fattree.pdf.”
[87] “https://www.indiamart.com/proddetail/intruder-detection-system-2698828997.html.”
[88] “https://www.keyence.eu/products/vision/vision-sensor/iv3/.”
[89] M. B. Hassan, R. A. Saeed, O. Khalifa, E. S. Ali, R. A. Mokhtar, and A. A. Hashim, “Green Machine Learning for Green Cloud Energy Efficiency,” in *2022 IEEE 2nd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering, MI-STA 2022 - Proceeding*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 288–294. doi: 10.1109/MI-STA54861.2022.9837531.
[90] M. Manavi, Y. Zhang, and G. Chen, “Resource Allocation in Cloud Computing Using Genetic Algorithm and Neural Network,” Aug. 2023, [Online]. Available: http://arxiv.org/abs/2308.11782
[91] “https://www.techspot.com/article/2048-machine-learning-explained/.”
[92] “https://www.upsite.com/blog/defining-liquid-cooling-in-the-data-center/.”
[93] “https://www.itcreations.com/ibm/ibm-system-x3550-m3-server.”
[94] “https://dcomcomputers.com/compaq-proliant-g3-dl360-server-2-80ghz-2gb.html.”
[95] “https://in.element14.com/voltech/pm1000/power-analyser-single-phase/dp/1553887.”