Vrije Universiteit Amsterdam

Bachelor Thesis

Evaluating the Environmental Costs of Client-Server Gaming

Author: Yuval Nir (2708117)

1st supervisor: Jesse Donkervliet
daily supervisor: Jesse Donkervliet
2nd reader: Alexandru Iosup

A thesis submitted in fulfillment of the requirements for the VU Bachelor of Science degree in Computer Science

August 25, 2023

Abstract

As online gaming continues to surge in popularity, its energy footprint, largely attributed to data centers, is a growing concern. This thesis aims to investigate the environmental impact of online gaming and to identify pathways for reducing its energy consumption. Employing a focused approach, the study develops an initial model for energy usage in gaming data centers and incorporates it into the OpenDC discrete-event simulator. A series of experiments are conducted to explore the influence of various factors like network usage and data rates on the overall energy consumption. The results offer preliminary insights into energy optimization within this domain, including the incorporation of network usage into power consumption calculations and the generation of dynamic workloads for different gaming scenarios. Although the model and implementation have limitations and make simplifying assumptions, they lay a foundation for future research to build upon, with the ultimate aim of steering the online gaming industry toward more sustainable practices.

Contents

1

Introduction

Video gaming is a popular leisure activity worldwide. With the rise of powerful gaming computers and consoles, as well as the widespread availability of high-speed internet, video games become more and more accessible and immersive. It is estimated that there are approximately 3.2 billion video-game players worldwide in 2022 (1). A big part of these games takes place online. Either as games that run locally and are connected to a server that manages the game's states, thus enabling a shared virtual world, and also as 'cloud-gaming' services, that run a game on a remote server, and stream the game as a video to the client's machine. To support these technologies on a large scale, online games operate as distributed systems in cloud data centers.

The growth of the online gaming industry has put pressure on internet service providers and data centers to support this demand, which poses a significant challenge in terms of energy consumption and emissions. The energy consumption of data centers within the European Union is expected to increase by 28% by 2030 compared to 2018 (2). The study also highlights that there is no singular solution to achieve sustainable data centers by the target year.

The environmental impact of playing online games remains an unclear and under-researched area. As computer scientists, it is our moral obligation to ensure that the rapid growth of cloud technologies in the gaming industry is sustainable and does not harm the environment. The challenge of optimizing energy efficiency is compounded by the fact that most gaming data servers are housed within data centers that serve multiple purposes, making it difficult to ensure energy efficiency and reduce the environmental impact (3). The energy consumption and emissions associated with online gaming data centers are a growing concern, and it is important for stakeholders to address this issue in a sustainable manner.

The goal of this project is to investigate the environmental cost of playing online video games, and how can we design our data centers to decrease this cost. Data centers are spread all around the globe, thus keeping them energy efficient is a global concern, and with this project, we hope to reach conclusions that can have a positive effect on the environment going forward.

1.1 Problem Statement

The energy consumption of video games is a growing concern, particularly with regards to server room design in cloud-based gaming ecosystems. Despite the widespread use of online video games, the energy consumption associated with these games is not well-quantified, making it difficult to address this issue.

To better understand the energy consumption of video games on the cloud side, there is a need to develop tools that can help analyze key performance and energy metrics for short- and long-term scenarios. These tools can assist in assessing energy use, performance, and operational costs, allowing informed decisions to be made to optimize energy consumption in the operation of online video games.

To achieve these goals, it is necessary to gather data on energy consumption and performance metrics for cloud-based gaming systems. This data can be used to create models that help predict energy usage under different scenarios, allowing developers and operators to make informed decisions on how to optimize their systems. By doing so, it will become possible to promote sustainable energy practices and reduce carbon emissions in the operation of online video games.

1.1.1 Challenges

One of the most significant challenges in accurately modeling the energy consumption of a gaming ecosystem lies in the scarcity and inaccessibility of data. Owing to a variety of factors such as proprietary information, data privacy regulations, and the sheer complexity and dynamism of the gaming ecosystem, gathering detailed, relevant, and reliable data can prove to be a monumental task. Accurate modeling requires granular data about server utilization, network traffic, user behavior, infrastructure details, energy usage of hardware, and many other factors. However, such data is rarely publicly shared. Furthermore, gaming companies may be reluctant to share such detailed information due to competitive reasons or data protection regulations. This lack of comprehensive data forms a considerable obstacle to creating a truly accurate and effective model for energy consumption in a gaming ecosystem.

1.2 Research Questions

1. **How to design a model for energy usage in the gaming ecosystem?**

   In order to better understand and improve the energy efficiency of data centers in the gaming ecosystem, it's crucial to develop a model for their energy usage. A well-constructed model can provide valuable insights for optimizing server utilization, cooling systems, network infrastructure, and other components that significantly influence a data center's energy consumption. This in turn can lead to substantial cost savings, reduce carbon footprint, and contribute to sustainability efforts.

   However, this task is intricate and challenging, owing to the complexity of the gaming ecosystem and the variety of client-server games it supports. Each type of game, from massive multiplayer online games to competitive eSports, has unique demands and patterns of server usage. Additionally, the need to balance performance and energy efficiency adds another layer of complexity. Furthermore, obtaining detailed, real-time data to construct and validate such a model can prove difficult, owing to factors like privacy issues, proprietary information, and the sheer scale of the required data. Despite these challenges, the pursuit of creating an accurate model for energy usage is a critical step toward a more sustainable gaming industry.

2. **How to implement such a model into a discrete-event simulator?**

   Incorporating an energy usage model into a discrete-event simulator such as OpenDC is a significant step towards understanding energy consumption in gaming data centers. This initial model addresses several key aspects including player activity, data center layout, server specifications, type of games, and network activity, forming the groundwork for more detailed simulations in the future.

   However, this task presents certain challenges. One must accurately represent the intricate gaming landscape and diverse server workloads, while also grappling with the limited availability of detailed real-world data. Further, validating the model against real-world scenarios requires access to granular data, which may not always be accessible.

Despite these challenges, this first step is crucial in the journey to improve energy efficiency in gaming data centers. It paves the way for more sophisticated models in the future, allowing for continuous enhancements in energy efficiency as our understanding grows and data accessibility improves.

1.3 Thesis Contributions

This thesis presents preliminary contributions to the field of data center energy optimization, specifically within the context of online gaming. These contributions are as follows:

- **Initial Model for Energy Consumption in a Gaming Ecosystem:** We have attempted to develop an initial model that estimates energy consumption in a gaming data center based on player activity, server workload, and server specifications. While far from definitive, this model is a first step in understanding the correlation between gaming operations and energy use in data center environments.
- **Incorporation of Network Usage in OpenDC Simulator:** By integrating network usage into the power consumption calculations of the OpenDC simulator, We have taken a modest step towards enhancing the simulator's ability to approximate real-world data center operations.
- **Procedure to Generate Dynamic Workloads Using Game-Specific Traces:** We have established a rudimentary procedure to generate dynamic workloads using game-specific traces, an early attempt to simulate a variety of games and a range of energy consumption scenarios within gaming data centers.

1.4 Plagiarism Declaration

I confirm that this thesis work is my own work, is not copied from any other source (person, Internet, or machine), and has not been submitted elsewhere for assessment.

1.5 Thesis Structure

The structure of this thesis is as follows:

Chapter 2 provides essential background information and terminology to facilitate the understanding of the discussions that follow. This includes an overview of existing data center architectures, the concept of virtualization, and a brief explanation of the client-server architecture.

Chapter 3 introduces our model for calculating energy consumption in a gaming ecosystem. It describes the key elements of the model, the assumptions we have made, and how it can be adapted to accommodate a variety of games and server specifications.

Chapter 4 details the implementation of our model into the OpenDC simulator. It outlines the modifications we made to the existing data center model and explains how we simulated different types of games using dynamic workloads.

Chapter 5 evaluates our model and its implementation, discussing the results of our experiments, the insights they provide into energy consumption and incorporation of network components into the OpenDC simulator, and the limitations of our research.

Chapter 6 relates our work to previous research in the field, discussing how our study complements and extends existing knowledge on dynamic resource provisioning, data center energy management, and data center simulation and modeling.

Finally, Chapter 7 concludes the thesis. It recaps our findings, discusses the limitations of our work, and offers suggestions for future research in gaming data center energy optimization.

2

Background

Online gaming encompasses a broad spectrum of digital entertainment offerings, ranging from social and mobile games to the vast domain of massively multiplayer online games (MMOGs). Our primary focus here is on a subset of online games, specifically state-exchanging or client-server games (4). The simplified architecture of these games is depicted in Figure 2.1, which will be referenced throughout this chapter.

2.1 Client-Server Online Gaming

Each player, running the game on their local device, initiates actions and decisions within their gaming client 1. These commands are sent through the internet 2 and reach the game server, which processes them. Post-processing, updates are disseminated back to the players, with their local clients rendering graphics and reflecting game state alterations 9.

The client-server architecture, though beneficial in terms of centralized management and enhanced file oversight, poses challenges related to infrastructure maintenance and the technical expertise required for its management (5). Particularly in MMOGs, this structure has been a mainstay since the genre's inception. These games tend to use a distributed server setup due to the sheer number of simultaneous player connections and the resultant server load (4).

MMOGs often employ a four-tier server framework that includes the client, proxy, application/game, and database layers. These distinct layers handle various tasks from security and connection management to game state upkeep and database operations (4).

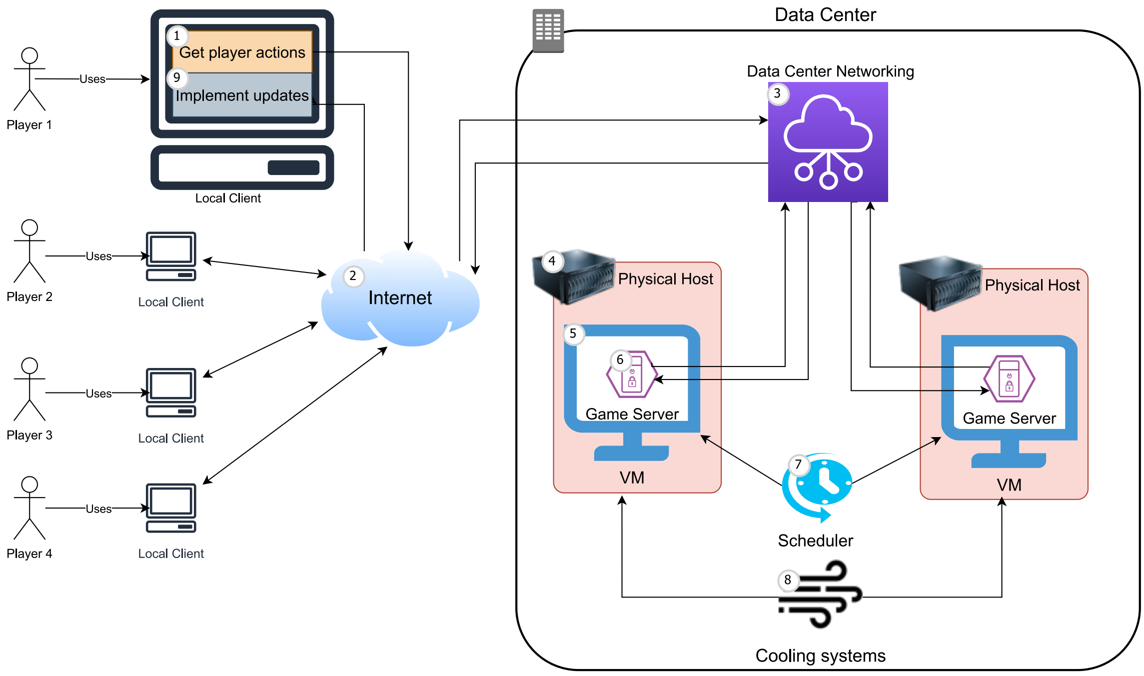

Figure 2.1: Simplified Client-Server Architecture.

2.2 Virtualization in Gaming

Virtualization technology is a cornerstone in modern computing, enabling optimal utilization of a physical machine's capabilities by divvying up its resources across multiple virtual environments or guest systems (6). In the architecture, the physical machine (or host) operates a software layer called the hypervisor. This hypervisor manages virtual machines (VMs) 5 which operate on the host. The VMs perceive the host's resources, such as CPU, memory, and storage, as a shared pool.

For online gaming, virtualization holds the distinct advantage of scalability. This facilitates dynamic creation or decommissioning of VMs in response to fluctuating player numbers, maintaining gameplay quality in the face of variable user interactions (7). However, the benefits of virtualization also come with challenges, especially concerning power consumption and resource allocation.

2.3 Data Centers

Data centers, the backbone of internet-based services like client-server gaming platforms, are critical in the digital services realm (8). These centers, while essential, are also notable for their high energy demands, influenced by factors ranging from hardware specifics to cooling requirements.

Data center networking 3 encompasses routers, switches, and other devices facilitating communication between servers and external networks. These devices ensure data packet routing, load balancing, and connectivity maintenance. Efficient networking is paramount for latency-sensitive applications like online gaming (9).

Schedulers 7 in data centers determine how tasks are assigned to computational resources. They prioritize tasks based on a variety of factors such as current load, task urgency, or specific resource requirements. Effective scheduling can significantly impact power efficiency and resource utilization (10).

Cooling systems 8 mitigate the heat generated by data center components, ensuring optimal operational conditions. They range from air-cooled solutions to more complex liquid cooling systems, all aiming to enhance energy efficiency (11).

Despite their extensive energy consumption, data centers present ample opportunities for efficiency enhancements. Crucial to these improvements are accurate power consumption models, which can aid component and system design, predict energy efficiency trends, and optimize energy consumption (12). Moreover, strategies for energy-aware resource allocation have emerged as a promising approach to efficient management of data center resources, reducing both the operational costs and environmental impact (13). The inherent operational and management costs of data centers highlight the urgency of these strategies, emphasizing the need for effective energy efficiency measures (14).

Given the dynamic and interactive nature of online gaming, accurately modeling a data center's energy consumption specific to this industry presents unique challenges. The potential for robust estimation methods tailored to gaming workloads is a promising avenue for research, holding significant implications for the efficient operation of gaming platforms and services.

3

Model for Energy Usage in a Gaming Ecosystem

In this section, we introduce our design for a model intended to calculate energy consumption within a gaming ecosystem. Figure 3.1 provides an overview of our model, depicting the following key elements:

- Element 1: Number of players, the starting point of our model.
- Element 2: CPU usage, calculated from the number of players and formulas modeled according to game genres.
- Element 3: Network usage, calculated from the number of players and formulas modeled according to data rate levels.
- Element 4: Components that can be easily added, such as GPU usage or Cooling usage.
- Element 5: CPU Energy usage, derived from CPU usage (Element 2).
- Element 6: NIC Energy usage, derived from network usage (Element 3).
- Element 7: The physical component that matches Element 4, used to model additional energy usage.
- Element 8: Total energy usage of the physical host, which accumulates to the total energy usage of a data center.

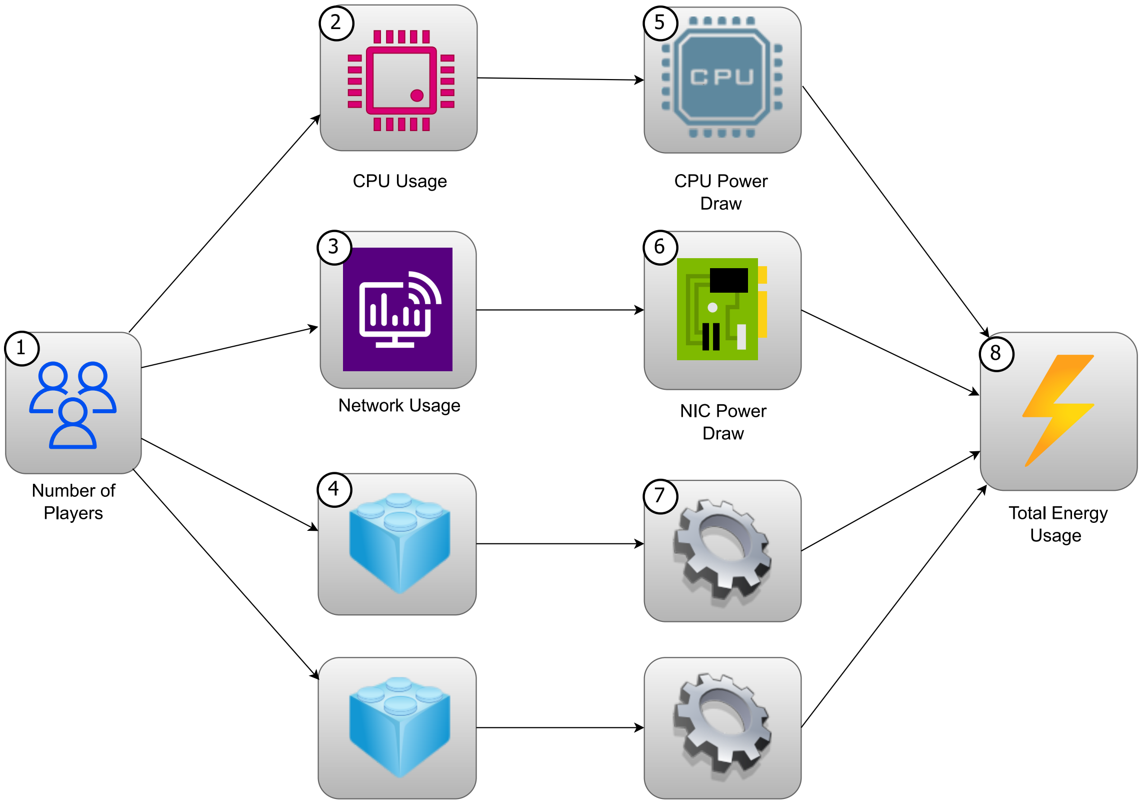

Figure 3.1: Our Model for Energy Usage in Online Gaming.

We outline the essential needs and specifications for this model and suggest potential solutions to fulfill these requirements. The following subsections will detail each component of the model, referring back to the corresponding elements as appropriate.

3.1 Model Requirements

R1. **Datacenter-scale Energy Modeling**

While network simulators and modeling tools such as NS3 (15) offer useful insights, they primarily focus on packet-level network interactions. Such granularity, though valuable in certain contexts, may become prohibitively expensive when scaled to model a datacenter's energy consumption. Therefore, our model emphasizes datacenter-scale energy modeling, deliberately opting to forego packet-level specificity in favor of broader, more macroscopic inputs and outputs. This approach allows us to assess energy usage patterns across the entire datacenter while keeping computational costs manageable, thereby better serving our specific objectives of energy optimization and efficiency. This large-scale focus does not dilute our model's effectiveness; instead, it enhances its ability to address energy concerns that are specifically relevant to datacenter operations in the gaming industry.

R2. **Multi-NIC Compatibility and Agnosticism**

In an ever-evolving data center technology landscape, the choice of Network Interface Cards (NICs) has a substantial impact on energy consumption. Therefore, the energy model must be robust enough to account for the unique characteristics of any NIC. This requirement acknowledges that different data centers may employ varying types of NICs, each with unique energy profiles. Our model is, thus, NIC-inclusive—it does not restrict or bias towards any particular brand or model, but rather, is designed to accurately model the energy consumption characteristics of any NIC presented to it. This universal compatibility ensures our model's relevance and adaptability across various data center configurations and emerging networking technologies.

R3. **Support energy usage of popular game genres**

Different game genres have significantly different communication patterns and thus, different energy consumption characteristics. Therefore, the energy model should be adaptable and flexible enough to cater to these varying gaming scenarios and provide energy consumption predictions across different genres of online games.

3.2 Design Overview of a Model for Energy Usage in Online Gaming

This section discusses the proposed architecture of our model, designed to offer a varied view of energy consumption within the gaming ecosystem. While the model does not interact directly with games, it utilizes gaming-related and data center-specific metrics, such as player counts, game genre, and network traffic. The goal is to facilitate a deeper understanding of the relationship between player activity, CPU and network usage, and energy consumption.

3.2.1 Data Rate for Different Games (R1)

To address the requirement of datacenter-scale energy modeling (R1), our model integrates a network usage 3 component into the workloads. This addition provides a more nuanced simulation of the real-time network demand dynamics in a gaming server, contributing to a more realistic representation of a datacenter environment.

Instead of using fixed data rates, our model introduces a novel way of calculating network usage. The model takes the number of players and their interactions as key inputs, represented by linear, square, or cubic forms. This decision is based on research indicating that network usage may scale proportionally to player interactions, which can vary dramatically in complexity and volume. The selected forms represent different levels of network usage:

Square Data Rate

Massively Multiplayer Online Role-Playing Games (MMORPGs) operate on different principles compared to other game types. The server has much heavier duties in terms of information updates and game configurations, and players may interact with each other or affect each other's surroundings, leading to additional updates. According to (16), the size of packets created by MMORPG game servers would generally be in the range of 4–636 bytes (servers) and 1–154 bytes (clients). The interarrival time of packet arrival ranges between 0–3179 ms (servers) and 0–1264 ms (clients). With this information, we model the MMORPG data rate as a square relationship:

$$N = P \times n^2$$

where:

- N is the total network usage,
- P is the network usage for a single player,
- n is the number of players.

This square relationship reflects the inherent complexity of MMORPGs, where server packet sizes are drastically different from client packets, indicating a higher duty on the server.

Cubic Data Rate

First-Person Shooter (FPS) games inherently exhibit complex server workloads and are characterized by high-paced interactions. An interaction between two players may also need to be updated to the rest of the players, which adds to the complexity and demand. FPS games are characterized by packet sizes in the range of 5–300 bytes (servers) and 15–110 bytes (clients), with interarrival times of 10–200 ms (servers) and 5–120 ms (clients) as reported in (16). These specifics, especially the shorter interarrival times, led us to model the data rate for FPS games as a cubic relationship:

$$N = P \times n^3$$

where:

- N is the total network usage,
- P is the network usage for a single player,
- n is the number of players.

The cubic relationship allows us to represent the intense and fast-paced interactions that are inherent in FPS games.

Linear Data Rate

The linear data rate can be applied to scenarios where network communication is relatively simple. In these cases, players may have small packet sizes and not too high demands on the network, possibly because each player's activities only affect their immediate surroundings, so the overall network demand isn't too high. Our model expresses this relationship:

$$N = P \times n$$

where:

- N is the total network usage,
- P is the network usage for a single player,
- n is the number of players.

3.2.2 From Utilization to Power Consumption (R2)

To address the second requirement, Multi-NIC Compatibility and Agnosticism (R2), our model incorporates a selection of network power models. These models, inspired by the foundational work by He et al on CPU Power Models (17), are designed to cater to the varying power characteristics of different NICs. An essential aspect of our model is the ability to accept user-defined values for both the idle power and max power of a NIC, enabling detailed customization of NIC-specific power consumption 6. These models are as follows:

- ZeroIdle: $P = 0$, assumes the NIC consumes no power.
- Constant: $P = s$, where s is a constant.
- Sqrt: $P = P_{idle} + (P_{max} - P_{idle}) \cdot \sqrt{u}$, considers power consumption as a square root function of network usage u.
- Linear: $P = P_{idle} + (P_{max} - P_{idle}) \cdot u$, treats power consumption as a linear function of network usage.
- Square: $P = P_{idle} + (P_{max} - P_{idle}) \cdot u^2$, calculates power consumption as a square function of network usage.
- Cubic: $P = P_{idle} + (P_{max} - P_{idle}) \cdot u^3$, calculates power consumption as a cubic function of network usage.

In these formulas, P denotes the NIC power consumption, P_{idle} is the power consumption when the NIC is idle, P_{max} is the maximum power that the NIC can consume, and u is the network usage.
These models enable our approach to accommodate an array of NIC power characteristics, thereby fulfilling the requirement for Multi-NIC Compatibility and Agnosticism. Paired with the input of user-defined NIC specifications, our approach provides a detailed and nuanced representation of NIC power consumption, thereby contributing to a comprehensive view of energy dynamics within a gaming data center.

This feature, in conjunction with the network usage data from the game-specific workload computations, enhances the accuracy of our energy consumption predictions, giving due consideration to the energy drawn by network activity within the gaming data center.

3.2.3 Modeling CPU Utilization for Different Game Genres (R3)

To address the third requirement, supporting energy usage of popular game genres (R3), our model employs game-specific computational methods to capture the unique CPU usage
2 patterns characteristic to different game genres. Specifically, we cater to three major game genres - Turn-Based Strategy Games (TBS), First-Person Shooter (FPS) games, and Massively Multiplayer Online Role-Playing Games (MMORPGs). By applying distinct mechanisms tailored to these game genres, we are able to provide a representation of their unique CPU usage patterns, contributing to an understanding of their energy usage within a data center environment.

Significantly, the calculated CPU usage directly impacts the CPU power consumption within the physical hosts of the data center, which in turn contributes to the total energy consumption of the data center. This sequence provides a representation of the energy dynamics within gaming data centers and provides a foundation for the genre-specific discussions in the following subsections.

Massively Multiplayer Online Role-Playing Game Workload

For MMORPGs, the CPU usage (U) is calculated using a formula based on the number of players (P) and a predetermined CPU usage attributed to each individual player (N). Specifically, our model computes CPU usage using a formula that integrates both linear and quadratic components:

$$U = N \times (P + P^2)$$

This formulation draws upon insights from the traffic characterization in "Game Traffic Analysis: An MMORPG Perspective" by Chen et al. (18).

The linear term, $N \times P$, captures the basic computational demands for individual player activities, as seen in the packet size distribution and basic server packet rates observed by Chen et al. It reflects the fundamental aspect of CPU usage that grows linearly with the number of players.

The quadratic term, $N \times P^2$, is more intricate and encapsulates additional CPU usage attributed to complex interactions between players. This complexity arises from various characteristics detailed in Chen et al.'s work, such as the clustering nature of player actions, the temporal and spatial locality in game nature, and the player interactions that intensify with an expanding player base. Together, these terms offer a way to grasp the MMORPG workload, drawing from the insights shared in Chen et al.'s paper.

First-Person Shooter Game Workload

For FPS games are known for their intricate server workloads, especially in terms of CPU utilization, which is closely tied to the number of active players in the game. The creation of mathematical models to represent this complexity has been an area of prior research. A prime example of this is the study "Modeling Ping Times in First Person Shooter Games" by Degrande et al. (19), which applied Upstream Modeling, Downstream Modeling, and a combination of Queues to intricately model the network traffic in FPS games, capturing various parameters that influence server workload.

In contrast to this detailed modeling, our work adopts a simplified approach while still expressing the essence of this complexity. Guided by specific assumptions made to tailor the model to CPU formula for FPS games, we propose the equation

$$U = P^N$$

where U stands for the server workload, P represents the number of players, and N is a constant. As the number of players increases, there is a corresponding exponential increase in the complexity of the game state, leading to a noticeable growth in server workload. This simplified model emphasizes the dominant factor in resource usage — the number of active players, aligning with the interactive and dynamic nature of FPS games. It provides an intuitive and tractable way to represent the essential complexity while omitting some of the more detailed aspects found in the original formula presented in Degrande et al.'s paper.

Turn-Based Strategy Game Workload

For TBS, the CPU usage (U) is calculated using a linear formula based on the number of players (P) and a predetermined CPU usage attributed to each individual player (N):

$$U = P \times N$$

As TBS games typically progress in a sequential order of turns, they exhibit steady and predictable server workloads. Therefore, this linear computation aptly models the CPU usage for TBS games, reflecting the consistent impact each player has on the overall CPU workload.

4

Implementation

In this chapter, we discuss the execution of our gaming server simulation extension to the OpenDC data center (20) (21). We outline the components we integrated into OpenDC, such as network usage, network components, and network power models. Moreover, we delve into our approach for workload generation. A simplified overview of the added components and their relationship to existing ones is presented in figure 4.1. As we move forward, we'll provide details on how each component was implemented and how the workload was generated to reflect gaming scenarios.

The experimental design is crafted to consider an array of variables like input workload, data center layouts, allocation policy, and performance metrics, facilitating a comprehensive exploration of the scenarios proposed by the simulation.

4.1 Workload generation

Within our extension to OpenDC, we've implemented specialized functions for workload generation 1 that dynamically create workloads by incorporating gaming traces and processing data rate levels. These functions interpret the traces, which consist of timestamps and corresponding player counts, presented in CSV format. For each row in the trace file, they calculate the CPU usage 2 based on the number of players using our CPU utilization formulas tailored to different game genres, as discussed in 3.2.3. Simultaneously, the network usage 3 is determined, again, based on the number of players and is anchored in our data rate formulas provided in 3.2.1. Lastly, the required number of cores is determined based on the player count within the VM. This dynamic workload creation is pivotal in simulating the server loads of various types of games, as it combines game-specific data

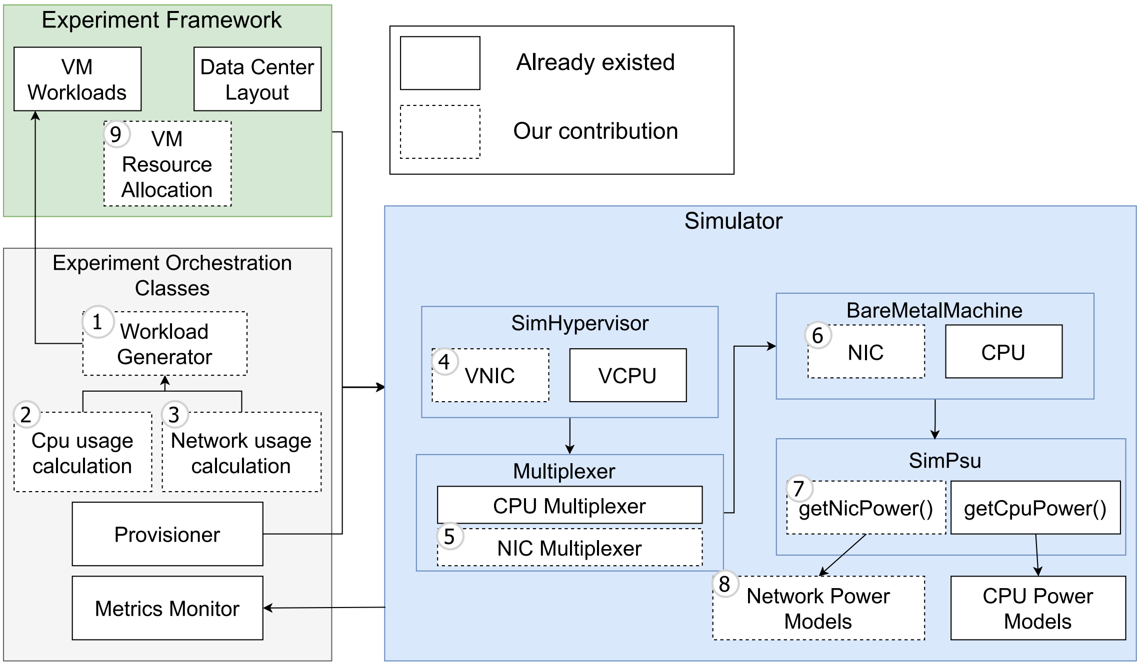

Figure 4.1: Simplified representation of the components we incorporated into OpenDC and their relationships with related components.

(player count per timestamp) with our predefined assumptions (CPU usage per player, network usage per player, and maximum load per VM).

4.2 Incorporation of Network Power Usage in Data Center Simulation

In this section, we detail how we augmented the OpenDC data center simulator with the inclusion of network components, as well as the integration of network usage into workload modeling.

In its original implementation, OpenDC acknowledged the presence of network components. However, it did not consider the impact of these components on the total power consumption of a physical machine. Given the significant role that networking plays in a data center's operation, especially in the context of online gaming servers, we identified the necessity to extend the simulator to include this detail.

In our refined model, we have integrated the notion of Virtual Network Interface Controllers (VNICs) for VMs. The physical hosts are equipped with NICs. We ensure that the power consumption of both the VNICs in SimHypervisor 4 and the NICs 6 is incorporated into the overall power usage calculation of the physical hosts and VMs respectively.

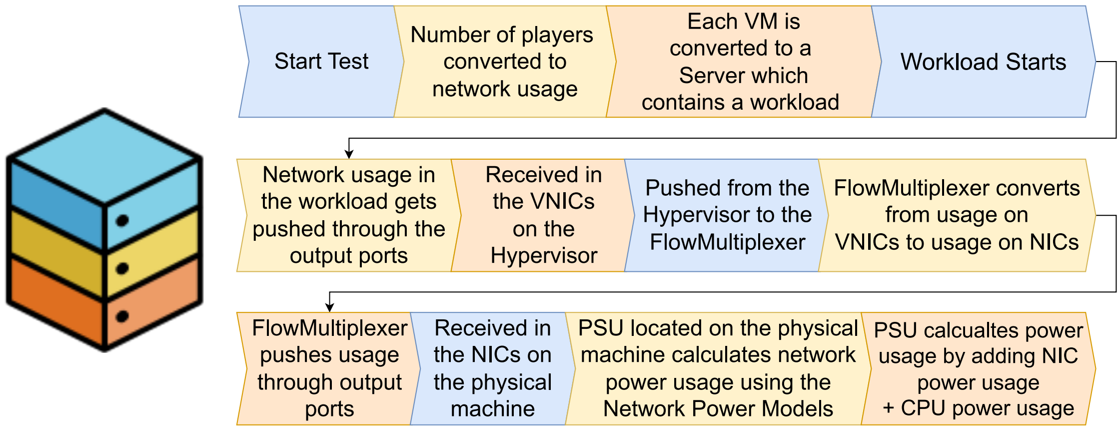

Figure 4.2: Simplified flow of how network usage gets converted to power usage.

A NIC Multiplexer (5) was added to the SimHypervisor, and is responsible for converting from VNIC usage to NIC power draw.

4.2.1 Addition of Network Usage to Workload

In parallel with adding network components, we have included network usage as a crucial factor in our workload construction. Each workload now comes with a network usage parameter.

Figure 4.2 provides a simplified flow of how network usage gets converted to power usage, trickling down from the workload to the physical host where consumption is calculated. This development enriches our workloads by taking into account the variable network demand of a gaming server, thereby providing a more realistic representation. Importantly, the network usage parameter also directly impacts the power consumption of the NICs. The network usage data from the workloads feed into the power consumption models of the NICs, represented by the getNicPower() function in SimPsu (7), in turn affecting the total power consumption of the physical hosts.

4.2.2 Network Power Models

The Network Power Models in our extension (8), are built upon the foundational work by He te al.'s CPU Power Models (17). These models offer various calculations to compute the power consumption of the NIC in a physical host in a data center. Our network models draw on the network usage figures supplied by our workload models. This addition enables us to obtain a more fine-tuned calculation of NIC consumption

This new element is a foundation stone for our energy consumption model, and, with more precise data, can be further refined and enhanced in future research endeavors.

4.3 Dynamic VM Resource Allocation

Traditionally in OpenDC, the number and capacity of VMs are statically defined, specified through input files in CSV or Parquet formats. While this approach is effective, it requires precise prior knowledge about the VMs' resource needs. However, such specific real-world data about VM allocation in gaming server contexts is often not readily available.

In order to address this challenge, we have introduced a novel function that generates VMs dynamically based on the player count 9. This allows us to bypass the need for a static file detailing VM resource specifications.

This approach introduces an improvement to the versatility and usability of OpenDC in the absence of specific VM specification data. By offering a way to simulate various server loads based on player count.

4.4 Flexibility for Emerging Energy Models

Given the nature of energy models as approximations, we rely on the most accurate models currently available. However, as the development of these models is an ever-evolving area of research, we anticipate that more refined models will emerge over time. With this in mind, our goal is to create a system that easily integrates these improved models as they become available. Ensuring that our approach is not only current but also adaptable to the future landscape of energy research is crucial.

Our commitment to adaptability in the face of evolving energy models is mirrored in the design framework of our system. We've opted for a modular approach, where each segment of the system can be upgraded or replaced on its own, negating the need for a comprehensive system revamp. This architectural choice brings several benefits:

- **Adaptable Workload Construction:** As we gain deeper insights into gaming dynamics and demands, our strategy for workload formulation can be refined.
- **Upgradable Network Power Models:** The current models, while founded on simple implementations, have been designed to be adaptable. They can be readily updated or expanded as newer and better methodologies emerge.

- **Precision in VM Resource Allocation:** As we accumulate more research data and insights, our system's allocation of VM resources can be fine-tuned, aiming for even greater accuracy in energy consumption predictions.

Thanks to our system's modular design, it's ready to adapt and grow with both tech improvements and new research findings.

5

Evaluation

5.1 Experimental Overview

In this chapter, we delve into the evaluation of the modifications made to the OpenDC simulator. We construct experiments that aim to answer key questions about how these improvements affect energy consumption within a data center under varied conditions.

1. **What is the impact of incorporating network usage into the power consumption calculation on the overall energy consumption of the data center?**

   This experiment is designed to provide a measure of how our modification (the inclusion of network usage in the computation of power consumption) influences the overall energy consumption in a data center. This will enable us to compare scenarios where network usage is considered to those where it is not. Through this comparison, we aim to quantify the contribution of network usage to the overall power consumption, highlighting the significance of this enhancement in the simulator. This aspect will be further addressed in section 5.4.

2. **How do different data rates affect the total power consumption?**

   This experiment is designed to assess the relationship between varying data rates and their subsequent impact on a data center's energy consumption. Data rates, understood as the speed at which data is transferred, can significantly influence the load levels of various data center components. Different genres of online games possess unique network demands, leading to fluctuations in data rates. By examining scenarios across a spectrum of data rates, we aim to uncover how diverse network demands can affect the overall energy consumption of data centers. Results and further discussion on this experiment will be elaborated upon in a subsequent section 5.5.

3. **How does changing the maximum player limit per VM influence the data center's CPU utilization and overall energy consumption?**

   Our third experiment revolves around the manipulation of the player limit per VM. By adjusting this variable, we create a variety of load scenarios, giving us insights into the relationship between VM workload, CPU utilization, and energy consumption. We will address this in section 5.6.

4. **How do different game genres influence data center energy consumption?**

   This experiment aims to investigate the impact of various game genres - specifically Turn-Based Strategy (TBS), Massively Multiplayer Online Role-Playing Game (MMORPG), and First-Person Shooter (FPS) — on the energy consumption of data centers. Each game genre has unique computational demands, and this experiment seeks to quantify how these demands translate to energy usage in a data center environment. A complete analysis and discussion of the results will be presented in a subsequent section 5.7.

5.2 Experimental Design

This section outlines the setup for our experiments, detailing the hardware and software configuration, key variables, constraints, and metrics used to measure the outcomes.

5.2.1 Physical Hosts Capabilities

Our experimental setup utilizes a simulated data center layout modeled after specifications of Azure's D-Series VMs, specifically the Dav4 and Dasv4 units. These VMs are general-purpose compute units featuring the AMD EPYC™ 7452 processor, which houses 32 cores and supports up to 96 virtual CPUs (vCPUs) (22, 23). The choice to model our hosts' capabilities after these specifications was made to provide a realistic yet manageable testing environment, given the absence of publicly available, real-world gaming data center layouts.

5.2.2 Experiment Workloads

To generate workloads for our VMs, we utilize player count data from MineTrack traces (24). Each timestamp provides the total count of players connected to the server at that instant. From this pool, we assign players to VMs. For each VM, we randomly select a number of players, ranging from one up to a predetermined maximum per VM. This procedure results in a dynamic distribution of players across VMs at each timestamp, simulating real-world scenarios of players logging in and out or switching between virtual servers.

CPU usage is calculated as a percentage of the VM's total CPU capacity, using game genre formulas. The usage percentage rises in proportion to the number of players on the VM and reaches its maximum capacity when the VM is hosting the maximum number of players.

Network usage is influenced by three data rate levels: Linear, Square, and Cubic. These levels follow our data rate formulas. Similar to CPU usage, network usage also scales proportionally with the number of players on the VM, within the bounds of the designated data rate level.

It should be noted that, due to a lack of precise data, these computations for CPU and network usage are based on our assumptions. They are intended to provide a reasonable approximation of VM workloads, acknowledging the often unpredictable nature of player behavior.

5.2.3 VM Capacity

In our experimental setup, each VM is assigned a certain number of vCPUs. This allocation of vCPUs varies according to the requirements of the specific experiment conducted.

The CPU capacity of each VM is determined based on the product of the maximum number of players it can host and the usage per player. Similarly, network capacity is calculated considering the maximum number of players and the data rate level.

Throughout the experiments, each VM remains operational from the beginning to the end. Like our other setup decisions, these capacity specifications are derived from assumptions made due to the lack of precise real-world data.

5.3 Overview of Findings

Our experiments sought to understand key aspects of data center energy consumption, specifically focusing on the impact of network usage, the influence of different player limits per VM across varying vCPU configurations, and the energy implications of different online game genres.

In Section 5.4, it was observed that incorporating network usage into power consumption calculations elevates the total energy consumption of the data center by approximately 10%. This finding is expected, especially when previous studies have indicated a similar contribution from network components to server power consumption, approximating the energy consumed by the NIC in a physical host.

Moving to Section 5.5, our analysis showed that as data rate formulas become more intensive, the energy consumption of the data center rises accordingly. The disparity in energy consumption between each formula hovered around 2%. This was anticipated since the bulk of the energy consumption originates from the operation of hosts and VMs. The study further highlighted that the duration of the experiment, confined to 24 hours, didn't allow the differences stemming from distinct data rate formulas to manifest substantially.

In Section 5.6, the aim was to discern patterns in energy and CPU usage based on the distribution of players across VMs. Overall, energy consumption remained relatively stable at approximately 4.35 GJ, irrespective of the manner in which players were distributed across VMs. In the same vein, CPU utilization exhibited only minor fluctuations. Notably, configurations that allocated a higher number of players per VM seemed to be the most efficient both in terms of energy and CPU usage. However, the practical implementation of such a configuration would be contingent upon the specific demands of individual games.

Section 5.7 delves into the energy implications of different game genres. In this segment, the energy consumption patterns for TBS and MMORPG were found to be almost indistinguishable, averaging at about 4.288GJ. This result was in line with expectations due to the experiment's constraints such as a lower number of players (1000-1500) and the 24-hour time frame. These limitations made it less likely for the quadratic elements in MMORPG's formula to have a noticeable impact. In stark contrast, FPS games exhibited a higher energy demand, averaging around 5.62GJ. The exponential formula associated with FPS was significant enough to impact energy usage, even with the experiment's limited scope. This indicates that the mathematical models governing each game genre can have a substantial impact on a data center's overall energy profile, especially in scenarios with lower player counts.

To wrap up, our experiments shed light on the nuances of energy consumption within data centers. The insights gained emphasize the importance of considering all factors, like network usage, when trying to gauge the energy profile of data centers. Moreover, they suggest that while individual VM configurations can affect their own performance metrics, the overall energy footprint of the data center remains largely consistent.

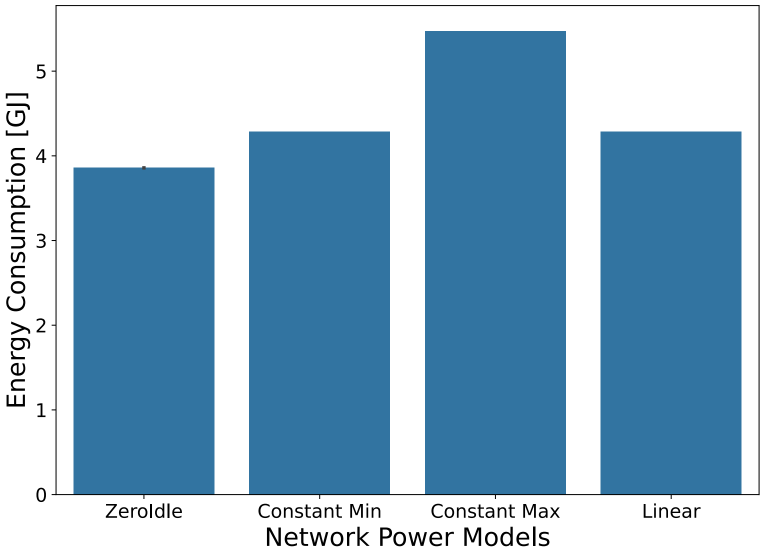

Figure 5.1: Energy Usage for Different Network Power Models.

5.4 Incorporation of Network Components Experiment

Our experiment shows that including network usage into power consumption calculations increases the total energy consumption of a data center by about 10%. This result aligns well with our expectations and is consistent with previous research (25), which suggests that the NIC consumes about 10% of the server's power. Other studies, such as that by Economou et al. (26), have also indicated that NIC consumption might hover around this 10% mark.

Figure 5.1 illustrates the results of our experiment runs wherein network usage was calculated using the linear data rate formula. While other data rate formulas, specifically square and cubic, were tested, they yielded results similar to the linear formula and thus are not separately outlined in the figure.

Our experimental design appropriately estimated the potential for the NIC to consume around 10%, as reflected in the idle and max power values assigned to the CPU and NIC (200W and 350W for CPUs; 24W and 86W for NICs, respectively) based on data from (27). The result from the Linear model, which provides a more nuanced and accurate representation of energy consumption, confirmed the approximations set forth by previous studies. This outcome suggests that, given the capacity of NICs with higher power consumption, network usage contributed as expected to the total power consumption.

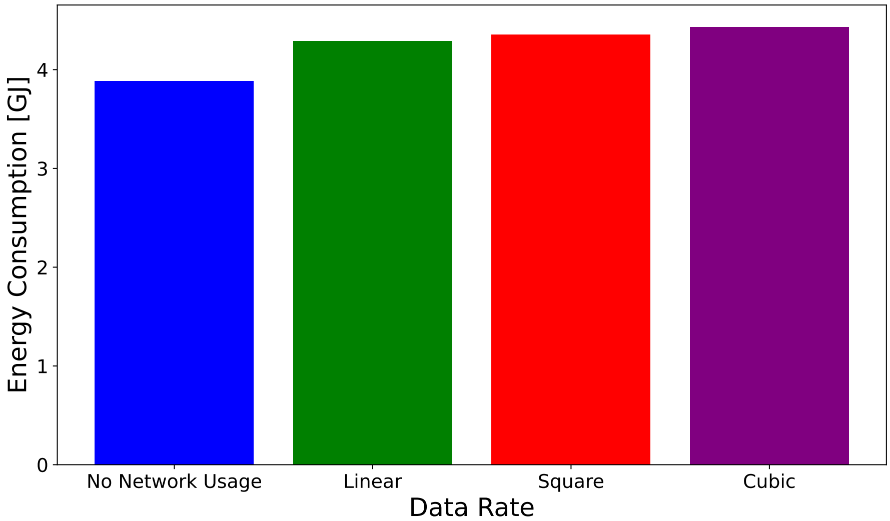

Figure 5.2: Energy Usage Across Different Data Rates.

The models used in the experiment, ZeroIdle, Constant Min, Constant Max, and Linear, each played a unique role in examining power consumption. In contrast to ZeroIdle, which excluded network power, Constant Min and Constant Max offered simplified yet less realistic baselines.

The 10% increase in energy consumption in the Linear model underscores the importance of considering network power consumption when estimating data center energy use. This result offers a more accurate depiction of real-world scenarios and underlines the necessity of including network power in energy consumption models.

5.5 Different Network Data Rates Experiment

Our findings from this experiment, as detailed in Figure 5.2, show that as the data rate formulas intensify, the overall energy consumption in the data center correspondingly rises. The difference in energy consumption between each data rate formula averages to about 2%. Although this result was anticipated, we had initially hoped for a more substantial impact from the different data rate formulas. From our earlier investigations, it's evident that the dominant factor in energy consumption is the operational demands of the hosts and VMs. Thus, minor variations in energy consumption, attributed to the diverse data rate formulas, don't significantly alter the total energy landscape.

Building upon insights from our previous experiment described in Section 5.4, where single-player network usage statistics were derived from (18) and the contribution of net-

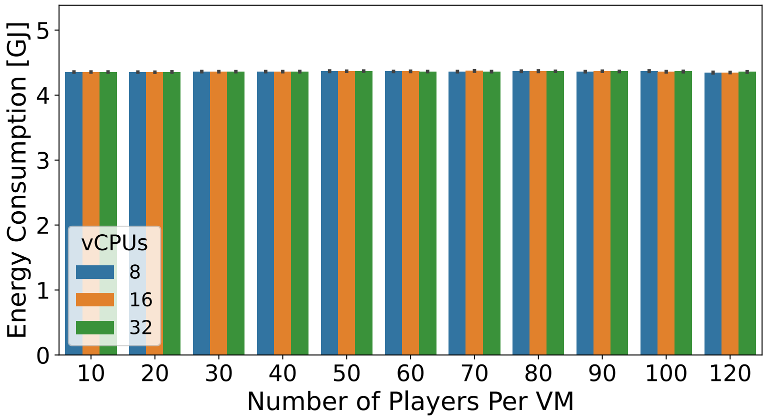

Figure 5.3: Energy Usage Across Different Number of Players and vCPU Configurations Per VM.

work components to overall energy consumption was established to be approximately 10%, we made further observations in the current experiment. Notably, the variance in energy consumption between the different data rate models (linear, square, cubic) was minimal in the previous experiment. However, in this experiment, we observed a 2% difference between each of the data rate models. This observed discrepancy is likely attributable to the different single-player network usage statistics employed in this experiment, which were derived from calculations presented in (28). This led to divergent network consumption rates and hence, differing contributions to overall energy usage in the data center.

One noteworthy aspect possibly influencing these results is the relatively brief duration of our experiment, which lasted only about 24 hours. Within this limited window, the hosts' operational demands overshadowed other factors, such as the nuances introduced by different data rate formulas, thereby making their impact less evident in the overall energy consumption.

5.6 Maximum Number of Players per VM Experiment

Across all examined scenarios, irrespective of the number of players per VM and for all vCPU configurations, the data consistently revealed that total energy consumption remained nearly constant, hovering around 4.35 GJ as evidenced in figure 5.3. Similarly, the CPU utilization showed minimal fluctuation across the settings, as captured in figure 5.4.

To provide clarity on the visuals, in both figures, the x-axis represents the number of players per VM. Each bar stands for a distinct configuration of vCPUs provisioned for each

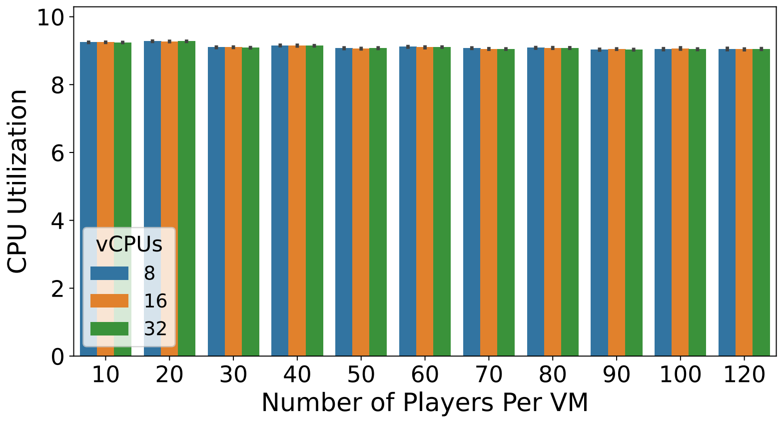

Figure 5.4: CPU Utilization Across Different Number of Players and vCPU Configurations Per VM.

VM. In figure 5.3, the y-axis delineates the total energy consumption, while in figure 5.4, it illustrates the accumulated CPU utilization—accumulated every time the metrics are recorded in the simulator.

This consistency in results aligned with expectations, primarily because every experiment hinged on the same trace. This meant the deployment of an equivalent number of players over a consistent time span. The underpinning logic is that with fewer players designated per VM, we inevitably employ more VMs, each demanding lesser power. However, when more players are designated to a single VM, fewer VMs are active, but their individual power demands spike. This equilibrium ensures that the total consumption remains balanced across differing setups. Notably, VMs with 10 and 20 players did register a slight rise in CPU utilization, but beyond these counts, the utilization remained largely stable.

An integral aspect explaining these observations is rooted in the simulator's design. In our simulator, the penalty associated with spinning up a new VM is notably minimal. Consequently, even when a lower number of players per VM prompts the use of more VMs, there isn't a significant spike in energy consumption. The minimal penalty for initializing VMs provides an understanding of the consistent energy consumption observed, even when the VM counts varied significantly based on player distribution.t

5.7 Different Game Genres Experiment

The results indicate negligible differences in energy consumption between TBS and MMORPG genres, both averaging around 4.288GJ. This was an expected outcome due to the limited

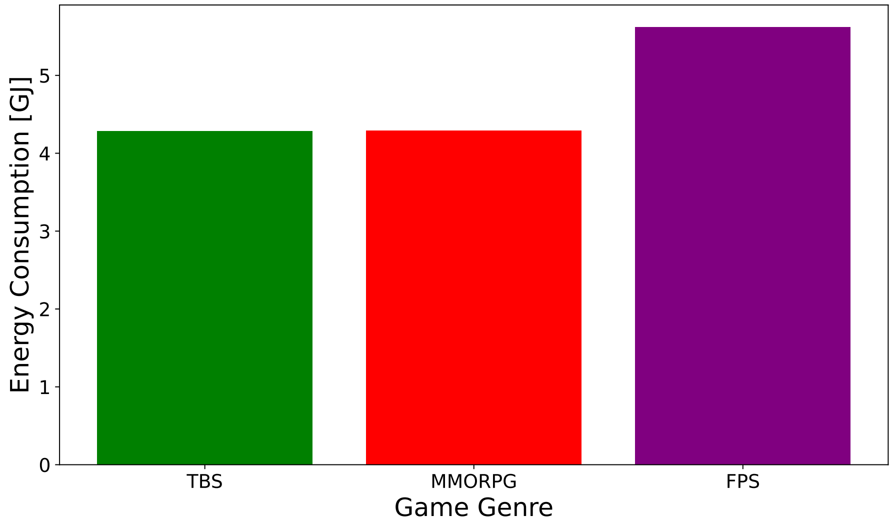

Figure 5.5: Energy Usage Across Distinct Game Genres.

number of players involved in our experiment (ranging from 1000 to 1500) and the low number of players per VM (7). Given these conditions, the quadratic term in the MMORPG formula had little room to significantly influence the energy consumption over the short 24-hour duration of the experiment.

In stark contrast, FPS games displayed a notably higher energy consumption level, averaging around 5.62GJ. This elevated energy usage in FPS games can be attributed to the exponential formula used for this genre, which had a substantial impact on energy consumption even within the short duration and low player count of our experiment.

In this experiment, we intentionally adjusted the CPU idle and max power settings to higher, less realistic values (200W for idle and 800W for max power). This was done to exaggerate the differences in energy consumption across genres and to highlight the impact of low versus high CPU utilization more distinctly.

The results underline the dramatic influence that the mathematical formulations corresponding to different game genres can have on energy consumption, particularly when player counts are low. For future work, longer experimental durations and varied player counts could offer more insights into the energy demands of each genre.

5.8 Limitations and Threats to Validity

While the study provides insights into the correlation between VM configurations, player limits, their impact on CPU utilization and energy consumption, as well as the inclusion of network power consumption in total energy estimates, several limitations and potential threats to the study's validity are worth mentioning.

Firstly, due to the limited availability of real-world data, we had to rely heavily on assumptions to build our research model. These assumptions, while methodically derived, inherently bring some level of uncertainty to our findings. Thus, while our conclusions are grounded in the best available theoretical projections and simulations, there's an acknowledged divergence risk between our results and potential real-world scenarios. It's essential to consider this when interpreting the general applicability of our conclusions.

Secondly, while our research leveraged a data center simulator, it's essential to acknowledge the intrinsic limitations of such tools. Even though OpenDC, the simulator we used, has been validated in previous studies like the "A Trace-Based Validation Study of OpenDC" (29), it remains an approximation of the intricate operations of a real-world data center. Although simulators can capture many nuances of data center behavior, the inevitable disparities between simulated and real environments may affect the direct translatability of our findings. While the aforementioned study adds credibility to OpenDC's capabilities, it's still imperative to remember that a simulator can't fully replicate every nuance of a real-world system.

Thirdly, the temporal scope of our experiment was relatively narrow, limited to a brief time window. The inability to run the experiment over a more extended duration, such as a month or even longer, curtails our grasp on the long-term repercussions of VM configurations, network usage, and different data rates. With a longer observation window, the margins between our results would likely grow, further differentiating the outcomes and potentially amplifying the distinctions in CPU utilization and energy consumption trends.

5.9 Summary

In summary, our research enhances the OpenDC simulator by incorporating network usage in power consumption calculations, adding a layer of precision to its estimations. We also performed an experiment to explore the impact of different VM configurations on CPU utilization and energy consumption, although the results did not provide substantial insights. A key addition to our study, outlined in Section 5.7, demonstrated the considerable energy consumption differences across game genres, especially the higher energy demands of FPS games. This work serves as a stepping stone for more comprehensive future investigations into data center optimization.

6

Related Work

6.1 Dynamic Resource Provisioning

Our research has benefited from several insightful studies, among which Marzolla, Ferretti, and D'Angelo's work (30) offers compelling thoughts on dynamic resource provisioning in the context of large-scale MMOGs. While we appreciated their utilization of a Queueing Network performance model, their thought-provoking approach to handling variable workloads guided us in our resource management strategy for gaming infrastructures.

In the same vein, Donkervliet, Cuijpers, and Iosup's investigation into the scalability of Minecraft-like services with dynamically managed inconsistency was an insightful reference (31). We found their consideration of a wide range of factors, from gameplay design to hardware specifics, a useful parallel for our work. We looked at similar variables, albeit with an emphasis on energy consumption.

6.2 Datacenter Energy Management

Hongyu He's research (17) into the role of data centers in the energy market has been a noteworthy source of inspiration. He's intricate CPU power models provided us with a fresh perspective on network power consumption. While our study took a different route, not implementing machine learning methods and DVFS technology directly, we did draw ideas from his models to help shape our own understanding of network power management.

6.3 Datacenter Simulation and Modeling

In the realm of datacenter simulation, Mastenbroek et al.'s development of OpenDC 2.0 has been a valuable resource (21). Their simulation platform offered us practical insights into datacenter operations that we were able to incorporate into our own models. Their work on unifying the energy resource chain and abstracting various physical resources into a common unit further enriched our understanding and influenced the design of our study.

7

Conclusion

7.1 Answering Research Questions

1. **How to design a model for energy usage in the gaming ecosystem?**

   In Chapter 3, we presented our design of a model aimed at calculating energy consumption in a gaming ecosystem. Our approach, based on a set of assumptions and hardware power models, sought to correlate player activity, server workload, and energy usage.

   Key aspects of our model include support for large-scale online games, adaptability to different types of games, and considerations of data center topology and server specifications. We hypothesized that each VM instance on a server manages several active players, with resource allocation based on the game's requirements and desired service quality.

   To facilitate future adaptation, the model was designed to be flexible and adjustable, capable of incorporating new game types and hardware technology advancements. As the gaming industry continues to evolve and as more data becomes available, our model can be refined to improve its accuracy and effectiveness. Therefore, our research successfully answered the first question, providing a model that could be a valuable tool for exploring energy consumption in gaming data centers.

2. **How to implement such a model into a discrete-event simulator?**

   In Chapter 4, we presented the process of implementing our model into the OpenDC simulator. The procedure involved a few critical steps. Firstly, we incorporated network components and their power usage into the existing data center model. Secondly, we implemented a dynamic VM allocation system, which was responsive to varying player counts, hence effectively accommodating fluctuating server loads in a gaming context. Lastly, we established functions to generate dynamic workloads using game-specific traces, facilitating the simulation of diverse types of games. Collectively, these modifications allowed a more accurate and versatile simulation of energy consumption within gaming data centers.

7.2 Limitations and Future Work

In our current work, we have made strides towards developing a robust model for energy consumption within gaming data centers. However, like any research endeavor, our study has limitations that present opportunities for future work.

Firstly, our model makes certain assumptions for simplification, such as uniformity in power usage for network components and linear scalability of VM resource allocation. These assumptions might not hold true in real-world scenarios, where power usage and resource allocation can be dynamic and non-linear due to myriad factors. More nuanced models can be developed in the future, considering these variables to enhance the accuracy of the simulations.

Secondly, the implementation of our model, while broadly applicable, is still somewhat limited considering the vast number of game genres and unique scenarios that client-server games can encompass. A more detailed examination of the relationship between different game genres and their energy consumption patterns could prove insightful for future work. Moreover, a study of the impact of player behavior on energy consumption, such as peak playing hours or large multiplayer events, could add another layer of complexity to the model.

In terms of hardware, our model could also benefit from comparing the energy efficiency of different server architectures, types of CPUs, and networking equipment. This could lead to a better understanding of the hardware factors influencing energy consumption in gaming data centers.

Finally, our current model could be expanded to include other critical aspects such as cooling systems, which contribute significantly to the total energy consumption of a data center.

With more time, we would have conducted these additional experiments to further refine our model. Nevertheless, our work offers a foundation for these future endeavors. We believe that the continuous refinement of these models will provide valuable insights into the energy consumption characteristics of gaming data centers, thereby guiding the industry towards more sustainable practices.

References

[1] Newzoo. **Newzoo Global Games Market Report 2022 Free Version**, 2022. 1

[2] European Commission, Content Directorate-General for Communications Networks, Technology, F Montevecchi, T Stickler, R Hintemann, and S Hinterholzer. *Energy-efficient cloud computing technologies and policies for an eco-friendly cloud market : final study report.* Publications Office, 2020. 1

[3] Evan Mills, Norman Bourassa, Leo Rainer, Jimmy Mai, Arman Shehabi, and Nathaniel Mills. **Toward Greener Gaming: Estimating National Energy Use and Energy Efficiency Potential**. *The Computer Games Journal*, **8**, 12 2019. 1

[4] Martijn Moraal. **Massive Multiplayer Online Game Architectures**, 2007. 7

[5] Shakirat Sulyman. **Client-Server Model**. *IOSR Journal of Computer Engineering*, **16**:57–71, 01 2014. 7

[6] Praneel Reddy Padala. *Virtualization of Data Centers : study on Server Energy Consumption Performance.* Master's thesis, , Department of Computer Science and Engineering, 2018. 8

[7] Vlad Nae, Radu Prodan, Thomas Fahringer, and Alexandru Iosup. **The impact of virtualization on the performance of Massively Multiplayer Online Games**. pages 1–6, 11 2009. 8

[8] Flexera. **2023 State of the Cloud Report**, 2023. Accessed on: April 11, 2023. 8

[9] Wenfeng Xia, Peng Zhao, Yonggang Wen, and Haiyong Xie. **A Survey on Data Center Networking (DCN): Infrastructure and Operations**. *IEEE Communications Surveys Tutorials*, **PP**:1–1, 11 2016. 9

[10] Léo Grange, Georges Da Costa, and Patricia Stolf. **Green IT scheduling for data center powered with renewable energy**. *Future Generation Computer Systems*, **86**, 03 2018. 9

[11] Alfonso Capozzoli and Giulio Primiceri. **Cooling Systems in Data Centers: State of Art and Emerging Technologies**. *Energy Procedia*, **83**:484–493, 12 2015. 9

[12] Miyuru Dayarathna, Yonggang Wen, and Rui Fan. **Data Center Energy Consumption Modeling: A Survey**. *IEEE Communications Surveys Tutorials*, pages 1–1, 01 2015. 9

[13] Anton Beloglazov, Jemal Abawajy, and Rajkumar Buyya. **Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing**. *Future Generation Computer Systems*, **28**:755 – 768, 05 2012. 9

[14] Katarzyna Mazur and Bogdan Ksiezopolski. **On Data Flow Management: the Multilevel Analysis of Data Center Total Cost**, 2017. 9

[15] Henderson T. R. Riley G. F. **The Ns-3 Network Simulator**. *Modeling and Tools for Network Simulation*, pages 15–34, 2010. 12

[16] Xianhui Che and Barry Ip. **Packet-level traffic analysis of online games from the genre characteristics perspective**. *Journal of Network and Computer Applications*, **35**(1):240–252, 2012. Collaborative Computing and Applications. 14, 15

[17] Hongyu He. **How Can Datacenters Join the Smart Grid to Address the Climate Crisis? Using Simulation to Explore Power and Cost Effects of Direct Participation in the Energy Market**, 08 2021. 15, 21, 35

[18] Kuan-Ta Chen, Polly Huang, Chun-Ying Huang, and Chin-Laung Lei. **Game traffic analysis: an MMORPG perspective**. pages 19–24, 06 2005. 17, 30

[19] N. Degrande, D. De Vleeschauwer, R. E. Kooij, and M. R. H. Mandjes. **Modeling Ping Times in First Person Shooter Games**. In *Proceedings of the 2006 ACM CoNEXT Conference*, CoNEXT '06, New York, NY, USA, 2006. Association for Computing Machinery. 17

[20] Alexandru Iosup, Georgios Andreadis, Vincent Beek, Matthijs Bijman, Erwin Eyk, Mihai Neacsu, Leon Overweel, Sacheendra Talluri, Laurens Versluis, and Maaike Visser. **The OpenDC Vision: Towards Collaborative Datacenter Simulation and Exploration for Everybody**. pages 85–94, 07 2017. 19

[21] Fabian Mastenbroek, Georgios Andreadis, Soufiane Jounaid, Wenchen Lai, Jacob Burley, Jaro Bosch, Erwin van Eyk, Laurens Versluis, Vincent van Beek, and Alexandru Iosup. **OpenDC 2.0: Convenient Modeling and Simulation of Emerging Technologies in Cloud Datacenters**. In *2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, pages 455–464, 2021. 19, 35

[22] Microsoft Azure. **Virtual Machines Pricing**. `https://azure.microsoft.com/en-us/pricing/details/virtual-machines/series/#resources`. Accessed: 07 03, 2023. 26

[23] AMD. **AMD EPYC 7452 Product Information**, 2023. [Online; accessed 13-June-2023]. 26

[24] MineTrack. **MineTrack Data**. `https://data.minetrack.me/`, 2023. Accessed: April 11, 2023. 26

[25] Lauri Minas and Brad Ellison. **The Problem of Power Consumption in Servers**, 2009. accessed: 08/08/2023. 29

[26] Dimitris Economou, Suzanne Rivoire, Christos Kozyrakis, and Partha Ranganathan. **Full-system power analysis and modeling for server environments**. 01 2006. 29

[27] Ripduman Sohan, Andrew Rice, Andrew Moore, and Kieran Mansley. **Characterizing 10 Gbps network interface energy consumption**. pages 268 – 271, 11 2010. 29

[28] Trevor Alstad, Riley Dunkin, Simon Detlor, Brad French, Heath Caswell, Zane Ouimet, Youry Khmelevsky, and Gaétan Hains. **Game network traffic simulation by a custom bot**. 04 2015. 31

[29] Jaro Bosch, Laurens Versluis, and Alexandru Iosup. **A Trace-Based Validation Study of OpenDC**. Technical report, Vrije Universiteit Amsterdam, 12 2020. 34

[30] Moreno Marzolla, Gabriele D'Angelo, and Stefano Ferretti. **Dynamic Resource Provisioning for Cloud-based Gaming Infrastructures**. *ACM Computers in Entertainment*, **10**(3), December 2012. 35

[31] Jesse Donkervliet, Jim Cuijpers, and Alexandru Iosup. **Dyconits: Scaling Minecraft-like Services through Dynamically Managed Inconsistency**. In *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, pages 126–137, 2021. 35

Appendix A

Reproducibility

A.1 Abstract

The following appendix discusses the methods used to obtain the evaluation data presented within Chapter 5, and the necessary steps to reproduce.

A.2 Artifact check-list (meta-information)

- **Program:** Java-based discrete-event simulator using the OpneDC library for energy consumption modeling.
- **Compilation:** Code compiled using Gradle, JDK version 17.0.6.
- **Run-time environment:** Java(TM) SE Runtime Environment (build 17.0.6+9-LTS-190)
- **Hardware:** local machine
- **Metrics:** Energy usage, CPU utilization
- **Output:** Raw, plain text
- **How much disk space required (approximately)?:** 1.5G for OpenDC repository
- **How much time is needed to prepare workflow (approximately)?:** Approximately 20 minutes
- **How much time is needed to complete experiments (approximately)?:** Approximately 15 minutes per experiment
- **Publicly available?:** Yes

A.3 Description

A.3.1 How to access

git clone https://github.com/YuviNir2/opendc.git

A.3.2 Software dependencies

The code for this project was developed using IntelliJ IDEA. This IDE integrates well with Gradle, which is used for compilation. It's recommended to use IntelliJ IDEA for building the project and running the experiments, although it's not strictly necessary if you're comfortable using Gradle from the command line.

A.4 Installation

To set up the development environment, follow these steps:

1. Install IntelliJ IDEA from the official website.
2. Open the cloned project in IntelliJ IDEA.
3. IntelliJ IDEA should automatically recognize the Gradle build file and attempt to download the necessary dependencies. If not, you can manually trigger a Gradle sync from the `View > Tool Windows > Gradle` menu.
4. After the dependencies have been downloaded and the project is successfully built, you can run the experiments from within IntelliJ IDEA. Navigate to the class `IntegrationTest` containing the method `integrationTest()`, right-click and choose `Run 'IntegrationTest.integratio` to verify everything is running smoothly.

A.5 Evaluation and Expected Results

A.5.1 Running an experiment

To run experiments, you must run the `runExperiment()` method found in the `GamingExperiment` class. This method relies on a `configFile` variable to select an appropriate experiment configuration file from the `experiment-configs` folder. Each experiment configuration file contains the following fields:

- `envFileName`: Specifies the data center topology, selected from files in the `env` folder.
- `traceFileName`: Chooses the trace file from the `traces` folder.
- `singlePlayerCpuUsage`: The CPU usage per single player.
- `maxCoresPerVm`: Maximum number of cores per VM.

- `singlePlayerNetworkUsage`: Network usage per single player.
- `maxNicsPerVm`: Maximum number of NICs per VM.
- `maxNumPlayersPerVm`: Maximum number of players per VM.
- `vmMemoryCapacity`: Memory capacity per VM.
- `cpuMaxPower`: Maximum CPU power.
- `cpuIdlePower`: CPU idle power.
- `cpuPowerModel`: Power model for the CPU.
- `nicMaxPower`: Maximum NIC power.
- `nicIdlePower`: NIC idle power.
- `nicPowerModel`: Power model for the NIC.
- `gameType`: Genre/Type of the game being simulated.
- `dataRate`: Data rate level for the network.

Upon running the `runExperiment()` method, the metrics will be both displayed in the console and written to a result text file. This file will be saved in the `results` folder. The naming convention for this result file is as follows:

"env-${envFileName}_trace-${traceFileName}_config-${configName}_${Random.nextInt(1000)}"

These results will provide detailed metrics on energy usage, CPU utilization, and other variables as configured in the experiment.

A.5.2 Validating Our Experiments

To validate the experiments and ensure the consistency of the results, it is crucial to utilize the correct configuration files for each experiment:

- Experiment 1, described in Section 5.4, was run using configuration files that have names beginning with `add_net_usage`.
- Experiment 2, outlined in Section 5.5, used the configuration files that start with `datarate`.

- Experiment 3, detailed in Section 5.6, was conducted using the configuration files that have names beginning with `num_players`.
- Experiment 3, detailed in Section 5.7, was conducted using the configuration files that have names beginning with `gamegenre`.

By following these guidelines, you can validate the experiments and reproduce the results accurately.

A.6 Experiment Customization

Customizing the experiments to suit different objectives is straightforward. Simply add a new configuration file in the `experiment-configs` folder. After adding your custom configuration file, replace the `configFileName` variable value in the source code with the name of your new file. This allows you to define your experimental setup and parameters, thereby enabling you to create and run your own experiments.

A.7 Methodology

Submission, reviewing and badging methodology:

- `https://www.acm.org/publications/policies/artifact-review-badging`
- `http://cTuning.org/ae/submission-20201122.html`
- `http://cTuning.org/ae/reviewing-20201122.html`