Energy and Carbon-Efficient Resource Management in Geographically Distributed Cloud Data Centers

Atefeh Khosravi

Submitted in total fulfilment of the requirements of the degree of

Doctor of Philosophy

School of Computing and Information Systems
THE UNIVERSITY OF MELBOURNE

April 2017

Energy and Carbon-Efficient Resource Management in Geographically Distributed Cloud Data Centers

PhD Candidate: Atefeh Khosravi
Principal Supervisor: Prof. Rajkumar Buyya

Abstract

Cloud computing provides on-demand access to computing resources for users across the world. It offers services on a pay-as-you-go model through data center sites that are scattered across diverse geographies. However, cloud data centers consume huge amount of electricity and leave high amount of carbon footprint in the ecosystem. This makes data centers responsible for 2% of the global CO_2 emission, the same as the aviation industry. Therefore, having energy and carbon-efficient techniques for distributed cloud data centers is inevitable. Cloud providers while efficiently allocating computing resources to users, should also meet their required quality of service.

The main objective of this thesis is to address the problem of energy and carbon-efficient resource management in geographically distributed cloud data centers. It focuses on the techniques for VM placement, investigates the parameters with largest effect on the energy and carbon cost, migration of VMs between data center sites to harvest renewable energy sources, and prediction of renewable energy to maximize its usage. The key contributions of this thesis are as follows:

1. A VM placement algorithm to optimally select the data center and server to reduce energy consumption and carbon footprint with considering energy and carbon related parameters.

2. A dynamic method for the initial placement of VMs in geographically distributed cloud data centers that simultaneously considers energy and carbon cost and maximizes renewable energy utilization at each data center to minimize the total cost.

3. Variations of VM placement methods, which explore the effects of different parameters in minimizing energy and carbon cost for a cloud computing environment.

4. The optimal offline algorithm and two online algorithms, which exploit available renewable energy levels across distributed data center sites for VM migration to minimize total energy cost and maximize renewable energy usage.

5. A prediction model for renewable energy availability at data center sites to incorporate into online VM migration algorithm and maximize renewable energy usage.

Declaration

This is to certify that

1. the thesis comprises only my original work towards the PhD,
2. due acknowledgement has been made in the text to all other material used,
3. the thesis is less than 100,000 words in length, exclusive of tables, maps, bibliographies and appendices.

Atefeh Khosravi, April 2017

Acknowledgements

PhD is a rewarding but challenging journey, which would not be possible without the help of many people. First and foremost, I would like to express my sincere gratitude to my supervisor, Professor Rajkumar Buyya, who has given me the opportunity to undertake a PhD in his group. I would like to thank him for continuous guidance, support, patience, and encouragement throughout all rough and enjoyable moments of my PhD endeavor.

I would like to express my deepest appreciation and gratitude to the members of my PhD committee, Professor James Bailey for his generous and kind guidance and Dr. Rodrigo Calheiros for providing insightful comments and support.

I am grateful to Amazon Web Services (AWS) for giving me the opportunity, both during my Internship and afterwards, to perform exciting work and build my career. I am thankful to my manager and all my colleagues and friends at AWS for their support and for the trust they have shown in me.

I would like to thank all past and current members of the CLOUDS Laboratory, at the University of Melbourne. My deepest gratitude goes to Dr. Adel Nadjaran Toosi for always being so generous with his time and giving me constructive comments to improve my work. I thank Sareh Fotuhi, Yaser Mansouri, Amir Vahid Dastjerdi, Deepak Poola, Safiollah Heidari, Mohsen Amini Salehi, Nikolay Grozev, Anton Beloglazov, Farzad Khodadadi, Maria Rodriguez, Chenhao Qu, Yali Zhao, Jungmin Jay Son, Bowen Zhou, Deborah Magalhes, Tiago Justino, Guilherme Rodrigues, LinlinWu, and Mohammed Alrokayan for their friendship and support.

I would also like to express special thanks to my collaborators, Dr. Lachlan Andrew (Monash University, Australia) and Dr. Saurabh Kumar Garg (University of Tasmania,

Australia) for their guidance and constructive comments.

I acknowledge the University of Melbourne, Australian Federal Government, Australian Research Council (ARC), Google, and CLOUDS laboratory for granting scholarships and travel supports that enabled me to do the research in this thesis and attend international conference.

I would like to thank all my friends in Australia who helped me through tough times and always encouraging me to stay positive and keep working to reach my goal.

Finally, I would like to thank my parents, my sister Mina, and my brother Hamid for their unconditional and loving support to overcome all the difficulties encountered in this journey successfully.

Atefeh Khosravi
April 2017

Preface

This thesis research has been carried out in the Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne. The main contributions of the thesis are discussed in Chapters 2- 6 and are based on the following publications:

- **Atefeh Khosravi** and Rajkumar Buyya, *"Energy and Carbon Footprint-Aware Management of Geo-Distributed Cloud Data Centers: A Taxonomy, State of the Art, and Future Directions"*, Advancing Cloud Database Systems and Capacity Planning With Dynamic Applications, N. Kamila (editor), Pages: 27-46, IGI Global, Hershey, PA, USA, 2017.

- **Atefeh Khosravi**, Saurabh Kumar Garg, and Rajkumar Buyya, "*Energy and Carbon-Efficient Placement of Virtual Machines in Distributed Cloud Data Centers*", Proceedings of the 19^{th} International European Conference on Parallel and Distributed Computing (Euro-Par), Pages 317-328, Aachen, Germany, 2013.

- **Atefeh Khosravi**, Lachlan L. H. Andrew, and Rajkumar Buyya. *"Dynamic VM Placement Method for Minimizing Energy and Carbon Cost in Geographically Distributed Cloud Data Centers"*, IEEE Transactions on Sustainable Computing (T-SUSC), Volume 2, Number 2, Pages: 183-196, IEEE Computer Society Press, USA, 2017.

- **Atefeh Khosravi**, Adel Nadjaran Toosi, and Rajkumar Buyya, *"Online Virtual Machine Migration for Renewable Energy Usage Maximization in Geographically Distributed Cloud Data Centers"*, Concurrency and Computation: Practice and Experience (CCPE), Wiley Press, New York, USA, DOI:10.1002/cpe.4125, 2017.

- **Atefeh Khosravi** and Rajkumar Buyya, *"Short-Term Prediction Model to Maximize Renewable Energy Usage in Cloud Data Centers"*, Sustainable Cloud and Energy Services: Principles and Practice, W. Rivera (editor), Springer International Publishing AG, 2017 (in press, accepted in April 2017).

Contents

List of Figures

List of Tables

Chapter 1
Introduction

CLOUD computing is a paradigm focused on the realization and long held dream of delivering computing as a utility [1]. It enables businesses and developers access to hardware resources and infrastructure anytime and anywhere they want. Nowadays, the number of individuals and organizations shifting their workload to cloud data centers is growing more than ever. Cloud services are delivered by data center sites each containing tens of thousands of servers, which are distributed across geographical locations. The geographical diversity of computing resources brings several benefits, such as high availability, effective disaster recovery, uniform access to users in different regions, and access to different energy sources.

Over the recent years the use of services offered by cloud computing systems has been increased and different definitions for cloud computing have been proposed. According to the definition by the National Institute of Standards and Technology (NIST) [2]: "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction".

Cloud addresses the issue of under provisioning of resources for a running service and lose the potential users at the peak times or even over provisioning of resources that leads to wastage of capital costs. This definition highlights a major feature for cloud computing that is called elasticity of resources. By delivering computing as a utility to users and providing the resources based on the users' request, the users will be charged on a pay-as-you-go manner, such as other utility pricing models (e.g., electricity and

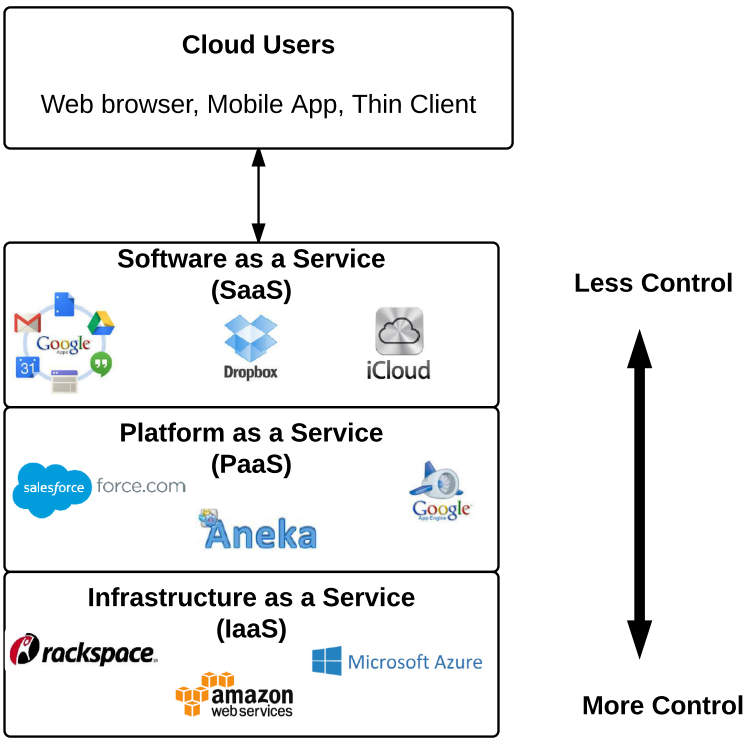


Figure 1.1: Cloud computing services.

water). In other words, users need not pay any upfront cost and the billing will be based on the usage (e.g. hourly) of the cloud resources.

Cloud delivers three main services to users as shown in Figure 1.1 and discussed in the following.

- Software as a Service: At the highest level there is Software as a Service (SaaS). SaaS service model, which is an old idea of cloud computing delivers on-demand software to users. Google Apps [3] and Salesforce [4] are examples of services offered in SaaS model. In this model, the control, support, and maintenance of the hardware, platform, and software of the cloud environment is shifted from the end-user to the cloud provider.

- Platform as a Service: Platform as a Service (PaaS) provides computing platform with pre-installed operating system, in order to enable the developers create their own software. By using PaaS, the developers need not concern about the underlying hardware and the operating system. Users can have scalable resources any-

time and anywhere. Google App Engine [5], Microsoft Azure[6], and Manjrasoft Aneka [7] are examples of PaaS environment.

- Infrastructure as a Service: Infrastructure as a Service (IaaS) located at the lowest layer of the cloud service stack offers computing physical resources such as servers, storage, hardware, networking, and virtual machines (VMs) to users. In this model, users have control over the operating system, storage, and applications while they need not manage the underlying infrastructure. Amazon EC2 [8], Google Cloud [9], and Rackspace [10] are some of the well-known IaaS providers.

Cloud computing like any other technology has its own challenges. Further in this thesis, we investigate one of the biggest challenges a cloud provider faces, which is the high energy consumption and carbon footprint.

1.1 Energy Consumption and Carbon Footprint Challenges in Cloud

Services by cloud computing are delivered by data centers that are distributed across the world, which can host small numbers to thousands of servers. A major issue with these data centers is that they consume a large amount of energy. According to a report from NRDC [11], US data centers power consumption estimation alone in 2013 was 91 billion kilowatt-hours of electricity. This is equivalent to two years' power consumption of New York City's households and is estimated to increase to 140 billion kilowatt hours by 2020, which is responsible for emission of nearly 150 million tons of carbon pollution.

To overcome the problem of high energy consumption and environmental concerns due to the high CO_2 emission of energy sources, there are possible solutions such as improving the data center's efficiency or replacing the brown energy sources with clean energy sources. By making data centers energy efficient and aware of energy sources, cloud providers are able to reduce the energy consumption and carbon footprint significantly [12].

Many cloud providers often maintain geographically distributed data center sites, similar to popular cloud providers (e.g., Amazon, Google, and Microsoft). Having sev-

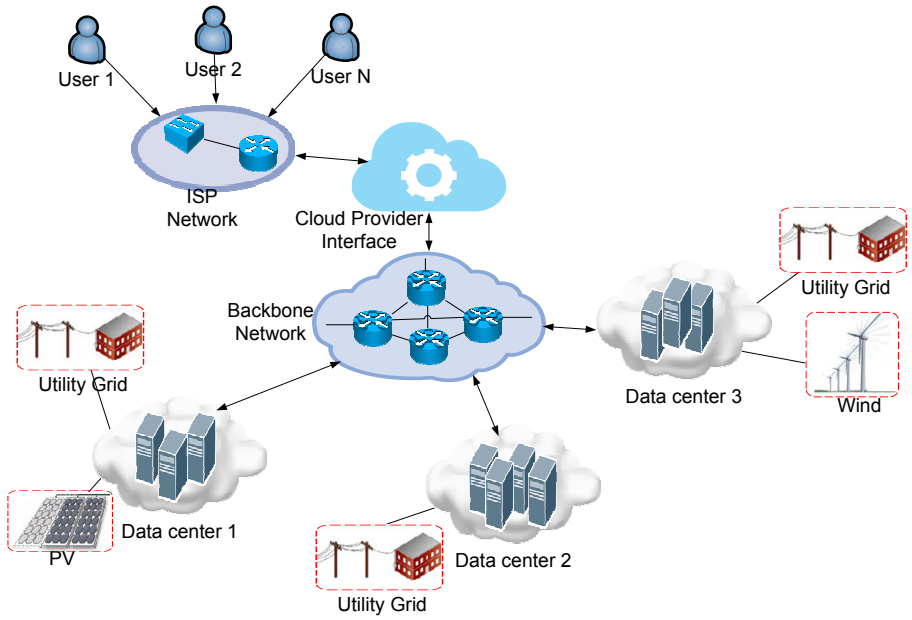


Figure 1.2: A simple system model used in this thesis.

eral sites not only increases the availability, it also gives the cloud provider the option of choosing the destination site based on different criteria upon the reception of the user request (VM requests in this thesis). Figure 1.2 depicts a high-level view of a cloud provider with distributed cloud data centers, each with different energy sources, to clarify our motivation and targeted model in this thesis.

By the arrival of each VM request from users, there are different challenges a cloud provider faces to select physical resources to instantiate the VM request. Moreover, later on in the system, effectively migrating the VMs to another data center could lead to significant improvements in energy consumption and carbon footprint. Thus, wisely taking into account parameters that affect resource scheduling for the VMs result in less energy consumption and less carbon footprint.

This thesis presents solutions to energy and carbon-efficient resource management in distributed cloud data centers. It is evaluated by theoretical analysis, development of algorithms, and extensive simulations using workload traces from Lublin-Feitelson workload model [13], Google cluster workload [14], real world meteorological traces from NREL [15], energy and carbon related rates from US Department of Energy [16] and US Energy Information Administration [17].

1.2 Research Problems and Objectives

This thesis tackles the research challenges in relation to energy and carbon-efficient resource management in geographically distributed cloud data centers. The targeted cloud computing system in this thesis is an IaaS provider offering VMs. In particular, the following research problems are investigated:

- **How to map VMs to physical resources?** Determining the best placement of the new VM requests to the available physical resources is a complex task. By the arrival of a new VM, performing an efficient selection of the destination data center and server within the data center, considering energy and carbon related parameters, has high impact on the energy consumption and carbon footprint of the system.

- **What are the important parameters on energy and carbon cost?** Determining the parameters with the highest effect on energy and carbon cost are important for a cloud computing environment. Moreover, identifying parameters that their effect changes over time and is dependent to the current state of the system is crucial.

- **How to design dynamic VM placement algorithms?** Considering the parameters that affect the total energy and carbon cost is important to design dynamic VM placement algorithms, evaluate their performance and identify the ones with the greatest impact on the total amount of renewable and brown energy consumption, carbon footprint, and cost.

- **When to migrate the VMs?** Considering the intermittent nature and limited amount of available renewable energy sources, migrating VMs to the nearby data centers with excess renewable energy helps to reduce and even eliminate brown energy usage. A crucial decision that must be made is determining the best time to migrate the VMs to minimize energy costs and maximize renewable energy usage.

- **Where to place the migrated VMs?** Making decision on the best placement of the migrated VM to another data center is another key aspect that influences the total energy cost.

To tackle the challenges associated with the above research problems, the following objectives have been identified:

- Explore, analyze, and classify the research in the area of energy and carbon-efficient resource management in geographically distributed cloud data centers to gain an understanding of the existing techniques and gaps in this area.
- Propose a VM placement algorithm to minimize energy consumption and carbon footprint, considering energy source and power usage effectiveness of data centers.
- Propose dynamic VM placement method and variations of it to provide insight to parameters with highest effect on energy and carbon cost with the objective of maximizing renewable energy usage while minimizing energy and carbon cost.
- Design optimal offline algorithm and online algorithms and conduct competitive analysis of online algorithms to understand their performance compared to the optimal offline algorithm for VM migration to minimize energy cost and maximize renewable energy usage.
- Propose a novel prediction model to maximize renewable energy usage.

1.3 Thesis Contributions

The main contributions of this thesis can be broadly categorized into: 1) A survey and taxonomy of the area, 2) A novel VM placement algorithm to optimally select the data center and server to reduce energy consumption and carbon footprint, 3) A novel optimization model and VM placement to minimize total energy and carbon cost and maximize renewable energy usage, 4) An optimal offline algorithm, online algorithm, and a future-aware online algorithm to migrate the VMs across data centers to minimize cost and maximize renewable energy usage, 5) A prediction model of renewable energy in data centers to maximize renewable energy usage. The **key contributions** of the thesis are as follows:

1. A taxonomy and survey of the state-of-the-art in energy and carbon-efficient resource management in distributed cloud data centers.

2. Energy and carbon-efficient VM placement in distributed cloud data centers:
    - An energy and carbon-efficient cloud architecture, based on distributed cloud data centers.
    - An efficient VM placement algorithm that integrates energy efficiency and carbon footprint parameters.
    - A comprehensive comparison on carbon footprint and power consumption for different VM placement algorithms with respect to users' quality of service (number of rejected VMs).
3. A dynamic VM placement and cost optimization model in geographically distributed cloud data centers:
    - A new method for the initial placement of VMs in geographically distributed cloud data centers that simultaneously considers the cost of 1) overhead energy 2) servers' energy and 3) carbon footprint.
    - A novel VM placement method that maximizes renewable energy utilization at each data center to minimize the total cost.
    - Efficient two-stage VM placement approaches that respond to dynamic PUEs.
    - Extensive study of the variations of the proposed VM placement method, which explores the effects of different parameters in minimizing energy and carbon cost for a cloud computing environment.
4. Cost minimization and renewable energy usage maximization through VM migration across cloud data centers: offline and online algorithm:
    - Formulation of the offline cost optimization problem for VM migration, across geographically distributed cloud data centers, with respect to the availability of renewable energy.
    - Proof and competitive ratio analysis of the optimal online deterministic algorithm with no future knowledge against the optimal offline algorithm.
    - Design of an online VM migration solution with limited future knowledge regarding the solar/wind power availability.

- Evaluation of the proposed algorithms through extensive simulations using real-world renewable energy (solar and wind) traces and workload traces of a Google cluster.

5. Prediction model to maximize renewable energy usage in cloud data centers:

   - A short-term prediction model of renewable energy availability at data centers based on Gaussian mixture model.
   - Accuracy validation of the proposed model based on renewable traces from NREL [15] and the workload data from Amazon Web Services biggest region in US East (N. Virginia).

1.4 Thesis Organization

The core chapters of this thesis are derived from a set of publications during the PhD candidature. The thesis structure is depicted in Figure 1.3 and the rest of the thesis is organized as follows:

- Chapter 2 presents a taxonomy and survey of energy and carbon-efficient resource management techniques in cloud computing environment. This chapter is derived from:
  - **Atefeh Khosravi** and Rajkumar Buyya, *"Energy and Carbon Footprint-Aware Management of Geo-Distributed Cloud Data Centers: A Taxonomy, State of the Art, and Future Directions"*, Advancing Cloud Database Systems and Capacity Planning With Dynamic Applications, N. Kamila (editor), Pages: 27-46, IGI Global, Hershey, PA, USA, 2017.
- Chapter 3 proposes a VM placement algorithm to increase the environmental sustainability considering data centers energy sources and power usage effectiveness (PUE). This chapter is derived from:
  - **Atefeh Khosravi**, Saurabh Kumar Garg, and Rajkumar Buyya, "*Energy and Carbon-Efficient Placement of Virtual Machines in Distributed Cloud Data Centers*",

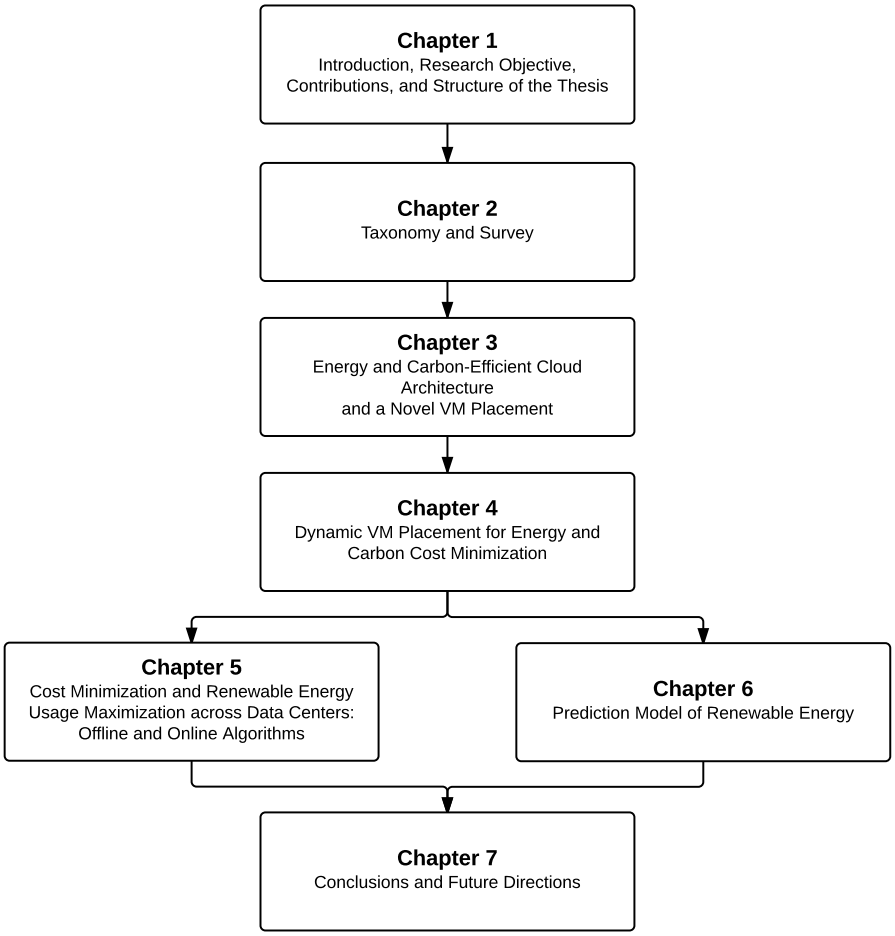


Figure 1.3: Thesis organization.

Proceedings of the 19^{th} International European Conference on Parallel and Distributed Computing (Euro-Par), Pages 317-328, Aachen, Germany, 2013.

- Chapter 4 investigates energy and carbon cost optimization parameters. It proposes VM placement method to maximize renewable energy utilization and minimize the total cost. The chapter also presents efficient two-stage VM placement approaches that respond to dynamic PUEs. This chapter is derived from:

  – **Atefeh Khosravi**, Lachlan L. H. Andrew, and Rajkumar Buyya. *"Dynamic VM Placement Method for Minimizing Energy and Carbon Cost in Geographically Distributed Cloud Data Centers"*, IEEE Transactions on Sustainable Computing (T-SUSC), Volume 2, Number 2, Pages: 183-196, IEEE Computer Society Press,

USA, 2017.

- Chapter 5 describes energy cost minimization and renewable energy maximization across geographically distributed cloud data centers based on offline and online algorithms. This chapter is derived from:

  – **Atefeh Khosravi**, Adel Nadjaran Toosi, and Rajkumar Buyya, *"Online Virtual Machine Migration for Renewable Energy Usage Maximization in Geographically Distributed Cloud Data Centers"*, Concurrency and Computation: Practice and Experience (CCPE), Wiley Press, New York, USA, DOI:10.1002/cpe.4125, 2017.

- Chapter 6 presents a prediction model of renewable energy at data center sites to maximize renewable energy usage for cloud providers. This chapter is derived from:

  – **Atefeh Khosravi** and Rajkumar Buyya, *"Short-Term Prediction Model to Maximize Renewable Energy Usage in Cloud Data Centers"*, Sustainable Cloud and Energy Services: Principles and Practice, W. Rivera (editor), Springer International Publishing AG, 2017 (in press, accepted in April 2017).

- Chapter 7 concludes the thesis with a summary of the key findings and discussion of future research directions.

Chapter 2
A Taxonomy and Survey

Cloud computing provides on-demand access to computing resources for users across the world. It offers services on a pay-as-you-go model through data center sites that are scattered across diverse geographies. However, cloud data centers consume huge amount of electricity and leave high amount of carbon footprint in the ecosystem. This makes data centers responsible for 2% of the global CO_2 emission. Therefore, having energy and carbon-efficient techniques for resource management in distributed cloud data centers is inevitable. In this chapter, we present a taxonomy and classify the existing research works based on their target system, objective, and the technique they use for resource management in achieving a green cloud computing environment. We discuss how each work addresses the issue of energy and carbon-efficiency.

2.1 Introduction

IN recent years the use of services that utilize cloud computing systems has increased greatly. Cloud computing consists of virtualized computing resources inter-connected through a network, including private networks and the Internet. It delivers service, platform, and infrastructure services to users through virtual machines (VMs) deployed on the physical servers. Virtualization technology maximizes the use of hardware infrastructure and physical resources. Hardware resources are the servers located within the data centers. Data centers are distributed across the world to provide on-demand access for different businesses. Due to the distributed nature of cloud data centers, many

This chapter is derived from the publication: Atefeh Khosravi and Rajkumar Buyya, "*Energy and Carbon Footprint-Aware Management of Geo-Distributed Cloud Data Centers: A Taxonomy, State of the Art, and Future Directions*", Advancing Cloud Database Systems and Capacity Planning With Dynamic Applications, N. Kamila (editor), Pages: 27-46, IGI Global, Hershey, PA, USA, 2017.

enterprises are able to deploy their applications, such as different services, storage, and database, in cloud environment. By the increase of demand for different services, the number of data centers increases as well; which results in significant increase in energy consumption. According to [18] energy usage by data centers increased by 16% from the year 2000 to year 2005. Energy consumption of data centers almost doubled during these five years, 0.5% and 1% of total world energy consumption in 2000 and 2005, respectively.

Hence, during the recent years there has been a great work on reducing power and energy consumption of data centers and cloud computing systems. Recently, considering data centers carbon-efficiency and techniques that investigate cloud data centers energy sources, carbon footprint rate, and energy ratings have attracted lots of attention as well. The main reasons for considering carbon-efficient techniques are increase in global CO_2 and keeping the global temperature rise below 2 °C before the year 2020 [19].

In the rest of the chapter, we provide an in-depth analysis of the works on energy and carbon-efficient resource management approaches in cloud data centers, based on the taxonomy showed in Figure 2.1. We explore each category and survey the works that have been done in these areas. A summary of all the works is given in Table 2.1.

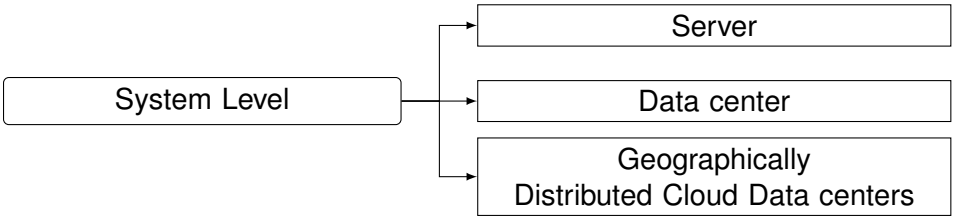


Figure 2.1: Taxonomy of energy and carbon-efficient cloud computing data centers.

2.2 Energy Efficiency in Servers

Servers are the physical machines that run the services requested by users on a network. Servers are placed in a rack and any number of racks can be used to build a data center. Servers along with cooling systems and other electrical devices in the data centers consume 1.1-1.5% of the global electricity usage [20]. Hence, power and energy management of servers by the increase in users' demand for computing resources is irrefutable. Figure

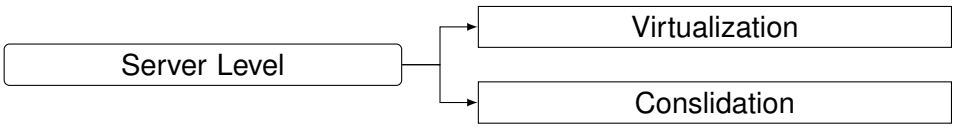


Figure 2.2: Server level energy and carbon-efficient techniques.

2.2 shows a classification of techniques that are used in data center servers to reduce energy consumption. Virtualization and consolidation are two well-known strategies that make the data center servers energy-aware. These are two powerful tools that are applied in cloud data center servers in order to reduce energy consumption and accordingly carbon footprint.

2.2.1 Virtualization

Virtualization technology is the main feature of data center servers that leads to less energy consumption [21]. By having virtualized servers and resources, and using virtualization technology several VMs can be built on one physical resource. Three types of virtualization that are widely used in data centers are hardware, software, and operating system virtual machines. The VMs run on the servers share the hardware components, that helps the operators to maximize servers utilization and benefit from the unused capacity. By maximizing servers utilization, huge savings in cost and energy consumption of data centers will be made. Decrease in data centers costs and energy consumption is not the only advantage of using virtualization technology. As the average life expectancy of a server is between three to five years, data and applications need to be consolidated and migrated to another server. Virtualization helps these two techniques to be done faster and with less cost and energy.

2.2.2 Consolidation

Server consolidation technique benefits from emerging of multi-core CPUs and virtualization technology. Its aim is to make efficient usage of computing resources to reduce data centers cost and energy consumption [22]. Consolidation is used when the utilization of servers is less than the cost associated to run the workloads (energy cost to run

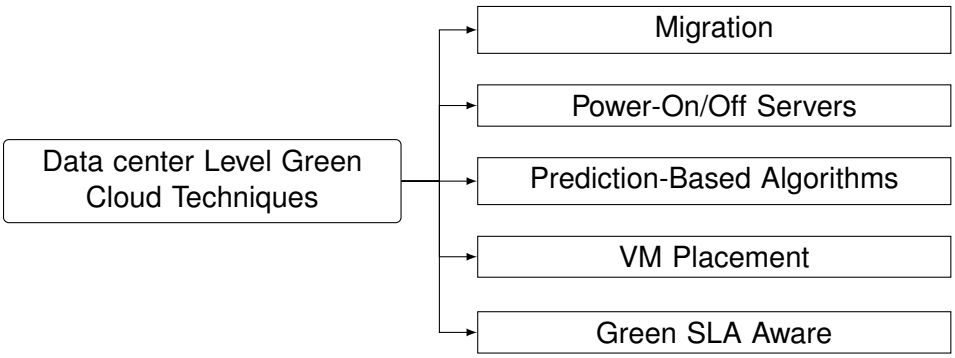


Figure 2.3: Data center level energy and carbon-efficient techniques.

servers and cooling cost for data center servers). By using consolidation, servers can combine several number of running VMs and workloads from different servers and allocate them on a certain number of physical servers. Therefore, they can power-off or change the performance-level of the rest of physical servers and reduce the energy consumption, cost, and carbon footprint.

2.3 Energy Efficiency in Data centers

This section gives an overview of the researches that have been done at data center level to improve carbon and energy-efficiency of cloud data centers. An extensive taxonomy and survey of these techniques is done by Beloglazov et al. [23]. Most of the works within a data center focuses on reducing energy consumption, which can indirectly result in carbon footprint reduction as well. Figure 2.3 classifies different approaches that have been taken for single data center. Some approaches use server level techniques (virtualization and consolidation) to migrate the current workload (user applications or virtual machines) and turn-off unused servers. Moreover, a provider could use the incoming workload pattern to place user request in the best suited cluster and server (and virtual machine for user applications) with less increase in energy consumption and carbon footprint.

2.3.1 Migration

Using virtualization, data center workloads migrate between servers. VM migration is the process of moving a running virtual machine from its current physical machine to another physical machine. Migration should be done in a way that all the changes be transparent to the user and the only change that user may encounter is a small increase in latency for the running VM or application.

Migration allows a virtual machine to be moved to another physical server so that the source physical server could be switched off or be moved to a power saving mode in order to reduce the energy consumption. VM migration in cloud data centers could be done off-line or live [24]. There has been a great amount of work done in this area try to identify the VMs on the servers with low utilization that could be migrated, so that the provider can put the unused servers in idle or power-off state.

2.3.2 Power-on and off Servers

When in an idle state data centers consume around half the power of their peak utilization and power state [25]. There are technologies that try to design data center servers so that they just consume power in the presence of load, otherwise they go to a power saving mode. Work that is done by Lin. et al. [26] uses a dynamic right-sizing on-line algorithm to predict the number of active servers that is needed by the arriving workload to the data center. Based on the experiments that are done in [26] dynamic right-sizing algorithm can achieve significant energy savings in the data center. We should consider that this requires servers to have different power modes and be able to transit to different states while still keeping the previous state. Moving the system to different power consumption modes is a challenging problem and requires dynamic on-line policies for resource management.

Green Open Cloud (GOC) is an architecture which is proposed by Lefevre et al. [27] on top of the current resource management strategies. The aim of this architecture is to switch-off unused servers, predict the incoming requests, and then switch-on required servers on the arrival of new requests. GOC proposes green policies to customers in the

way that they can have advance resource reservation and based on this knowledge cloud provider could estimate how many servers, and when they should be switched-on. Using this framework and strategy, they were able to save a considerable amount of energy on cloud's servers.

2.3.3 Prediction-Based Algorithms

Aksanli et al. [28] used the data from solar and wind power installations in San Diego [29] and National Renewable Energy Laboratory (NREL) [30], respectively to develop a prediction-based scheduling algorithm to serve two different types of workloads, web-services and batch-jobs. The main goal of this model was to increase the efficiency of the green energy usage in data centers. Based on the experiments of the proposed model, the number of tasks that were done by the green energy resources increased and the number of works that were terminated because of the lack of enough green energy resources decreased. This model uses a single queue per server for web services which are time sensitive applications, and for the batch-jobs it uses the Hadoop which is the general form of Map-Reduce framework.

GreenSlot scheduler [31] also proposes a scheduling and prediction mechanism to efficiently use the green energy sources. Goiri et al. [31] consider solar as the main source of energy and smart grid, known as brown energy, as the backup power source for the data center. The main objective of GreenSlot is to predict the availability of solar energy two days in advance so that it can maximize the use of green energy and reduce the costs associated with using brown energy. GreenSlot uses the suspension mechanism when there is not enough green energy available and based on the availability of enough solar energy it resumes the jobs. According to the experimental results that are presented in comparison with other conventional scheduling mechanisms, like backfilling scheduler [32], GreenSlot scheduler can significantly increase the use of green energy for running batch-jobs and decrease the brown energy costs, which leads to less carbon footprint and moving towards a sustainable environment. Unlike web-service jobs which are time-sensitive batch-jobs are compute intensive and the deadline is not critical as web-service jobs, so the suspension will not affect the user quality of service (QoS) parameters.

2.3.4 VM Placement

Users send their requests to the cloud Infrastructure as a Service (IaaS) providers in the form of VMs. Goudarzi et al. [33] presented a VM placement heuristic algorithm to place the VMs in physical servers in a way to reduce data centers energy consumption. The algorithm receives the VM requests and splits each VM into several copies and places them on servers. Each copy of VM gets the same amount of physical memory but with different CPUs. The total summation of assigned CPUs for copies of a VM request will be equal to the required CPU by the VM at the time of arrival to the data center. The proposed algorithm, which is known as MERA (Multi-dimensional Energy-efficient Resource Allocation), receives the VM requests and after a certain time epoch places the VMs on the servers and calculates the consumed energy. Then, it splits the VMs and places the copies on servers and recalculates the energy consumption. Based on the calculated energies the algorithm makes decision whether to split and replicate VMs or not. This algorithm tries to increase the servers utilization while decreasing the energy consumption without considering the physical characteristics and energy related parameters of servers and data centers. Moreover, it does not perform the VM placement dynamically. The algorithm receives a group of VMs and after a certain time epoch performs VM placement. In addition, inter-communication between replicated VMs could lead to bottleneck and high energy consumption. Finally, in the placement all VMs are treated the same. As all the replicated VMs get the same amount of physical memory, whilst for memory-intensive VMs this could result to shortage in resources and it is better to make balance between CPU intensive and memory intensive VM requests.

The work done by Xu et al. [34] addresses the problem of data centers VM placement with the objective to simultaneously minimize resource wastage, power consumption, and maximum temperature of the servers. They used a genetic algorithm on the global controller of the data center to perform the VM placement. The global controller receives the VM requests and then based on a multi-objective VM placement algorithm assigns each VM to a server. This algorithm, same as the previously discussed work, performs VM placement after receiving all the VM requests, which is not in a dynamic manner. Moreover, the algorithm makes balance between power consumption and temperature.

Therefore, it uses more servers to distribute the load and avoid hotspots in the data center. This might cause more carbon footprint as more servers will be used and more electricity will be consumed.

2.3.5 Green SLA Aware

Due to the high energy consumption by cloud data centers and climate concerns, cloud providers do not just rely on the electricity coming from brown energy sources. They have their own on-site green energy sources or draw it from a nearby power plant. Moreover, enterprises and individuals demand for quantifiable green cloud services. Haque et al. [35] propose a new class of cloud services that provides a specific service level agreement for users to meet the required percentage of green energy used to run their workloads. They undertake a new power infrastructure in which each rack can be powered from brown or green energy sources. The optimization policies have the objective of increasing the provider's profit by admitting the incoming jobs, with Green SLA requirements. If cloud provider cannot meet the requested percentage of green energy to run the job should pay penalty to the user, which means decrease in the total gained profit of running jobs. The type of green energy used by Haque et al. [35] in the data center is solar energy and they predict the availability and amount of solar energy based on the method proposed in [36]. The experiments carried in their work are based on comparison with greedy heuristics, and they show that optimization based policies outperform the greedy ones. Furthermore, among optimization based policies cloud provider can decide whether wants to increase the number of admitted jobs or violate less Green SLAs.

In the calculated total cost to run the admitted jobs in the work by Haque et al. [35], it is not clear that whether it is the cost to run the servers or the total cost in the data center, including overhead energy cost as well. This is important because overhead energy is dependent on the data center power usage effectiveness (PUE) and this varies by the change in the data center total utilization and ambient temperature [37,38]. Therefore, the calculated value for profit in the optimization based policies would vary based on the two aforementioned parameters for different jobs with different configuration requirements and also time of the day.

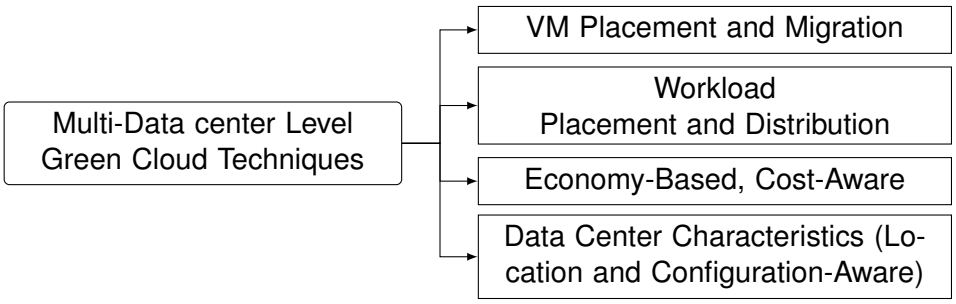


Figure 2.4: Multi data center level energy and carbon-efficient techniques.

2.4 Energy Efficiency in Geographically Distributed Data centers

Applying different policies to switch-off and on servers and placing user requests within a data center could lead to reduce in energy consumption. But still these are not enough to solve the problem of high energy consumption and carbon footprint by cloud data centers.

By increasing the use of cloud computing services that leads to increase in energy consumption and carbon footprint in the environment, some cloud providers decided to use green energy as a secondary power plant. Therefore, the need to have a scheduling policy to select the data center site to run the user request based on the energy source is necessary. Moreover, data center selection based on considering different data centers energy efficiency, as it has a direct effect on total carbon footprint, reduces energy consumption and carbon dioxide in the ecosystem. This section explores different energy and carbon-efficient approaches have been taken across distributed cloud data centers. Some of the applied techniques are the same as single data center level, but with considering factors to select the data center site before cluster and server selection. Figure 2.4 shows the taxonomy of different approaches taken at multi data center level with different optimization objectives, such as minimizing cost, energy consumption, carbon emission, and maximizing renewable energy consumption.

2.4.1 VM Placement and Migration

Research works in this area consider initial placement of a VM and further monitoring of the running VM to meet the optimization objective. Virtual machine (VM) placement in a geographically distributed data center environment requires selection of a data center and a server within the data center based on the optimization objective and data centers characteristics. Moreover, after the VM placement considering the future state of the host data center and other data centers, cloud provider can perform VM live migration to move the VM to another data center with preferable parameters. There are a few research works that consider these two techniques.

Chen et al. [39] developed a model for optimal VM placement considering a cloud provider with distributed data center sites connected through leased/dedicated lines. They introduce a cost-aware VM placement problem with the objective of reducing operational cost as a function of electricity costs to run the VMs and inter-data center communication costs. For this purpose, they take advantage of variable electricity costs at multiple locations and wide-area network (WAN) communication cost to place the VMs using a meta-heuristic algorithm. Similarly, Qureshi et al.[40] try to minimize electricity cost of running the VMs by initially placing the VMs into data centers with low spot market prices. They take advantage of spatial and temporal variations of electricity price at different locations.

Akoush et al.[41] propose an architecture known as Free Lunch to maximize renewable energy consumption. They consider having data center sites in different geographical locations in such a way to complement each other in terms of access to renewable energy (solar and wind) by being located in different hemisphere and time zone. The architecture considers pausing VMs execution in the absence of renewable energy or migrating the VMs to another data center site with excess renewable energy. The proposed architecture provides a good insight to harness renewable energy by having geographically distributed data center sites with dedicated network. However, this model has technical challenges and limitations dealing with VM availability, storage synchronization, VM placement and migration that have been pointed out in their work.

2.4.2 Workload Placement and Distribution

A large body of literature recently focused on reducing energy consumption and energy costs by load placement and distribution across geographically distributed data centers.

Le et al. [42] proposed a framework to reduce cost and brown energy consumption of cloud computing systems by distributing user requests across data center sites. This is the first research that considers load distribution across data center sites with respect to their energy source and cost. The framework is composed of a front-end that receives user requests and based on a distribution policy forwards the requests to the data center site with less cost and more available green energy sources. The request distribution policy sorts the data center sites based on the percent of the load that could be completed within a time period and minimum cost to run the requests. The evaluation results show that by knowing data centers' electricity price (constant price, dynamic, or on/off-peak prices) and base/idle energy consumption of the servers', significant improvements in cost reduction will be made. Moreover, being aware of the energy sources (green or brown) in the data centers could lead to less brown energy usage with a slight increase in the total cost.

Zhang et al. [43] use the idea of distributing the load among a network of geographically distributed data centers to maximize renewable energy usage. They proposed a novel middleware, known as GreenWare, that dynamically conducts user requests to a network of data centers with the objective of maximizing the percentage of renewable energy usage, subject to the cloud service provider cost budget. Experiment results show GreenWare could significantly increase the usage of renewable energies, solar and wind with intermittent nature, whilst still meeting the cost budget limitation of the cloud provider.

Following the idea of reducing brown energy consumption in data center sites, Liu et al. [44] proposed the geographical load balancing (GLB) algorithm. The algorithm takes advantage of diversity of data center sites to route requests to the places with access to renewable, solar and wind, energy sources. Considering the unpredictable nature of renewable energy, specially wind, GLB algorithm finds the optimal percentage of wind/solar energies to reduce the brown energy consumption and carbon footprint.

Moreover, the authors consider the role of storage of renewable energies, when they are not available in data centers in reducing brown energy usage. Based on the experiments, by using even small-scale storage in the data centers, the need for brown energy will decrease and in some cases even will be eliminated. A question that might rise with Liu et al. [44] work is the carbon footprint caused by the batteries in a long-term period, since renewable energy storage in the data center sites is done through reserving them in the form of batteries. Lin et al. [45] extended the GLB algorithm to reduce the total cost along with reducing the total brown energy consumption for geographically distributed data centers. They compared their proposed algorithm with two prediction-based algorithms with a look-ahead window, known as receding horizon control (RHC) a classical control policy and an extension of RHC known as averaging fixed horizon control (AFHC) [46]. The analytical modelling and the simulations carried, based on real workload traces, show that GLB algorithm can reduce the energy cost by slightly increase in network delay. Moreover, it can eliminate the use of brown energy sources by routing user requests to the sites where wind/solar energy is available.

Garg et al. [47] proposed an environment-conscious meta-scheduler for high performance computing (HPC) applications in a distributed cloud data center system. The meta-scheduler consists of two phases, mapping the applications to the data center and scheduling within a data center. They treat the mapping and scheduling of applications as an NP-hard problem with the objective to reduce carbon emission and increase the cloud provider profit at the same time. They run different experiments in order to find the near optimal solution for this dual objective problem. The parameters taken into account in the simulations and scheduling algorithms are data centers' carbon footprint rate, electricity price, and data center's efficiency. The simulations carried for high urgent applications (with short deadlines) and different job arrival rates help the cloud providers to decide for each application which scheduling algorithms should be used in a way to meet the objective of reducing the carbon emission or maximizing the profit. Moreover, they proposed a lower bound and an upper bound for the carbon emission and profit, respectively. Another work done by Garg et al. [48] addresses the issue of energy efficiency of ICT industry, specially data centers. The main focus of this work is to reduce the carbon

footprint of running workloads on data centers by proposing a novel carbon-aware green cloud architecture. This architecture consists of two directories, which imposes the use of green energy by data centers while meeting users and providers' requirements. In this framework, cloud providers should register their offered services in the aforementioned directories, and the users should submit their requests to the data centers through the Green Broker. The scheduling mechanism used in the broker, Carbon Efficient Green Policy (CEGP), chooses the cloud provider based on the least carbon footprint while considering users QoS parameters. The performance evaluation results of the proposed framework and policy in comparison with a traditional scheduling approach shows that CEGP can achieve a considerable reduction in energy consumption and carbon footprint in the ecosystem. However, this algorithm does not work dynamically. It receives all the job requests and based on the jobs deadline assigns them to the data center with the least carbon footprint. Moreover, it only considers high performance computing applications (non-interactive workloads) with predefined deadlines at the time of submission.

Chen et al. [49] use the idea of geographically distributed data centers to increase usage of green energy and reduce brown energy consumption in data centers. They proposed a workload scheduling algorithm, called MinBrown, that considers green energy availability in different data centers with different time zones, cooling energy consumption for data centers based on outside temperature and data center utilization, incoming workload changes during time, and deadline of the jobs. The workload used to run the simulation is HPC jobs with sufficient slack time to allow advanced scheduling. The algorithm copies all the data in all the data centers and based on the least consumed brown energy executes the task. Based on the simulation results, the MinBrown algorithm reduces brown energy consumption in comparison to other competitive algorithms. The idea of replicated data in distributed data center sites itself results to high energy consumption that is not considered in Chen et al. [49] work. Moreover, assignment of the jobs and tasks are based on the availability of green energy, that does not consider communication between tasks of the same job and jobs of the same workload. Finally, the scheduler does not consider an efficient resource assignment within a data center in a way to reduce the need for future consolidation of the running jobs.

The idea of federation of cloud providers can be useful for relocation of computational workload among different providers in a way to increase the use of sustainable energy. Celesti et al. [50] take advantage of a federated cloud scenario to reduce energy costs and CO_2 emissions. They consider cloud providers' data centers are partially powered by renewable energies along with getting the required electrical energy from electrical grids. The main contribution of their work is based on the approach of moving the workload towards the cloud data center with most available sustainable energy. This is inspired by the fact that if a provider generates more green energy than its need, it would be difficult to store the exceeded amount in batteries or put it in public grids; therefore, the easiest way is to relocate the workload to the site with the excess renewable energy. The architecture is based on an Energy Manager, that is known as CLoud-Enabled Virtual Environment (CLEVER). By applying CLEVER-based scenario, the VM allocation would be based on the energy and temperature driven policies. The energy manager in the architecture receives different data centers' information, such as temperature, sun radiation, energy grid fare, photovoltaic energy, cost, and data centers' PUE and number of available slots or physical resources, and based on this data assigns VMs to the site with the most sustainable energy and least cost.

Celesti et al. [50] work increases the use of sustainable energies and it is based on the availability of the photovoltaic (PV) energy. When a site has a high value for the PV energy, the outside temperature would be higher and this will increase the need for more energy for the cooling, and as a result higher PUE value. Relying only on the amount of used PV in the system is not enough for a green and sustainable system. Cloud providers should consider the whole picture and take into account all the parameters that affect the total CO_2 emission. Moreover, Celesti et al. [50] assume that each new VM request would be replicated in all the federated providers. Considering the consumed energy for this replication and the effect of network distance are also important that should be considered by the time of system design.

Xu et al. [51] take advantage of diversity in data centers location to route the incoming workload with the objective of reducing the energy consumption and cost. They studied the effect of ambient temperature on the total energy consumed by cooling system,

which is 30% to 50% of the total energy consumption of data centers [52,53]. Energy consumption often is modelled as a constant factor, which is an over-simplification of what is happening in reality. Xu et al. [51] considered partial PUE (power usage effectiveness) to participate cooling systems' energy along with the servers' total energy consumption. Through using partial PUE data centers can route the workload to the sites that use outside air cooling and reduce considerable amount of energy consumption. Moreover, they took advantage of having two types of incoming requests to manage the resources and reduce the energy consumption. The proposed model does not only depend on the energy consumed by interactive workload form users, instead it reduces energy costs by allocating capacity to the batch workloads, which are delay tolerant and can be run at the back-end of the data centers. The proposed joint optimization approach could reduce cooling energy and overall energy cost of data centers.

However, the proposed partial PUE only considers the energy consumed by cooling system as the total overhead energy in the data center. Based on the introduced definition by Xu et al. [51], PUE is mainly dependent on the ambient temperature, while IT load of the data center is the second important factor affecting the PUE [37]. Finally, source of the energy used to generate the electricity and its carbon footprint is not considered. This is important because as mentioned earlier reducing energy cost does not necessarily lead to reduce in the carbon footprint in the environment.

2.4.3 Economy-Based, Cost-Aware

Cost associated with energy usage in large data centers is a major concern for the cloud providers. Large data centers consume megawatts of electricity, which leads to huge operational costs. Work done by Ren et al. [54] takes advantage of different electricity prices in different geographical locations and over time to schedule batch jobs on the servers in scattered data centers. Their proposed online optimal algorithm, known as GreFar, uses servers' energy efficiency information and locations with low electricity prices to schedule the arrived batch jobs from different organizations. GreFar's key objective is to reduce energy cost, while assuring fairness considerations and delay constraints. The scheduling is based on a provably-efficient online algorithm, that schedules the jobs according to

the current job queue lengths. Based on the simulation results, GreFar online algorithm can reduce system cost, in terms of a combination of energy cost and fairness, in comparison to the offline algorithm that has knowledge of system's future state. The algorithm's main contribution is to serve the jobs when the electricity price is low or there are energy-efficient servers in the system. To accomplish this objective, it queues jobs and suspends low priority jobs whenever the electricity price goes up or there are not enough efficient servers in the system. This approach is not applicable for interactive jobs and web requests that are time sensitive and need to be served immediately from the queue and also cannot be suspended. Moreover, the cloud provider does not consider the cost of the transmission network and its energy consumption at the time of data center selection to submit the job request.

Le et al. [55] take advantage of capping the brown energy consumption to reduce the cost of serving Internet services in data centers. They proposed an optimization-based framework to distribute requests among distributed data centers, with the objective to reduce costs, while meeting users' service level agreement (SLA). The main parameters that affect the site selection by the framework are different electricity prices (on-peak and off-peak loads), different data centers location with different time zones, data centers with access to green energy sources, which enables the data center to have a mixture of brown and green energy. The front-end of the framework performs the site selection and optimization problem for the arrived requests periodically, in contrast to heuristic algorithms, which are greedy and select the best destination for each request that arrives [40]. The optimization framework uses load prediction by Auto-Regressive Integrated Moving Average (ARIMA) modeling [56] and simulated annealing (SA) [57] to predict the load for the next epoch (one week) and schedule the requests. This approach helps the front-end to decide about the power mixes at each data center for the next week, unless a significant change occurs in the system and predictions. Le et al. [55] use simulation and real system experiments with real traces to evaluate their proposed framework and optimization policy. The evaluation results show that by taking optimization policy and using workload prediction, diversity in electricity price, taking benefit of brown energy caps, and use of green energy sources significant savings in cost related to the execution

of Internet services in distributed data centers would be made. The framework assumes that all the received requests from the users are homogeneous. While in the real systems this is not the case and having heterogeneous requests and distributing them in a way to reduce resource wastage is very difficult and itself results to huge energy consumption and accordingly high costs. Moreover, it focuses on the electricity prices in different locations without considering the carbon footprint rate of the sources. Since some brown energy sources, which are cheap and lead to reducing the system overall cost, may lead to huge amount of carbon dioxide in the ecosystem.

The other work by Le et al. [58] investigates different parameters that affect the electricity costs for geographically distributed data centers with the focus on IaaS services that run HPC workloads. According to their proposed cost computation framework for the data centers, there are two important parameters that affect the total cost, energy consumed to run the service and the cost for the peak power demand. The provider can reduce the consumed energy by selecting the sites with off-peak period electricity, lower outside temperature, and lower data center load, so that the energy used for cooling would be low. Because as the data center temperature rises, the provider needs to use chillers to reduce temperature, which increases the energy consumption dramatically. In order to show this relation, they used a simulation model for the data center cooling system. Based on the simulation model, increase in the outside temperature and data center load forces the providers to use the chillers in order to keep the data center cool. This simulation has been carried with real workload traces from the Parallel Workloads Archive [59]. Le et al. [58] compared their two proposed algorithms, cost-aware and cost-aware with migration, with baseline policies. Based on the results, considering above mentioned factors can reduce the energy cost of data centers. Moreover, predicting the need to use the chillers for system cooling and considering the transient cooling prevents the data center from overheating and would not let spikes in the temperature.

Le et al. [58] conducted sensitivity analysis to investigate the effect of parameters, such as predicting the run-time of the jobs, the time to migrate the jobs, outside temperature, price of the energy in a region, and size of the data center on the total cost of the data center. According to the simulation results, in order to maximize the cost-saving all the

electricity-related parameters should be considered in job placement in the system. One of the shortcomings of this work, similar to the previously discussed work, is not considering the source of electricity. As some brown energy sources with high carbon footprint might be cheaper and more desirable to run the services. Moreover, as the temperature changes during the day and the consumed energy for cooling changes consequently; PUE should be modelled as a dynamic parameter instead of having a constant value per data center. Considering network distance and the energy consumption of intra and inter-data centers will also affect the total cost.

Work by Buchbinder et al. [60] has also the objective of reducing energy cost for a cloud provider with multi data center sites but with a different approach. They perform on-line migration of running batch jobs among data center sites, taking advantage of dynamic energy pricing and power availability at different locations, while considering the network bandwidth costs among data centers and future changes in electricity price. The total cost in their model, is the cost of energy to run the jobs at the destined data center plus the bandwidth cost to migrate the data. To attain an optimal algorithm with lower complexity comparing the optimal off-line solution, Buchbinder et al. [60] proposed an efficient on-line algorithm (EOA) with higher performance comparing to the greedy heuristics that ignore the future outcomes. The calculated cost in their work is based on the data centers' operational cost, which focuses on the energy consumption by servers and transport network. However, a considerable part of the energy consumed by a data center is related to the overhead energy, such as cooling systems. Moreover, the objective of reducing the energy cost and routing the jobs to the data center with lowest cost without considering the energy source might lead to increase in the carbon footprint in the environment. The migration of running jobs in this work is in the context of batch jobs, which are delay tolerant in comparison to user interactive requests such as web requests that are delay sensitive. Therefore, the applicability of this algorithm should be investigated for other workloads and user requests in a cloud computing environment. Similarly, work by Luo et al. [61] leverages both the spatial and temporal variation of electricity price to route the incoming requests between geographically distributed data centers targeting energy cost minimization.

2.4.4 Data Center Characteristics (Location and Configuration-Aware)

There are several works try to make data centers energy and carbon-efficient by reducing the number of active servers or run the virtual machines and applications on the physical machines with the least energy consumption and carbon footprint rate. However, geographical location of the data center has a direct impact on the amount of consumed energy that leads to CO_2 emission in the ecosystem. Work done by Goiri et al. [38] considers intelligent placement of data centers for Internet services. Their goal is to find the best location for data center site to minimize the overall cost and respect users' response time, consistency, and availability. They classified the parameters that affect data centers overall cost into location dependent and data center characteristics data.

The location dependent data specifies the data center's distance to the network backbones, power plants, and the CO_2 emission of the power plant. Moreover, it includes the electricity, land, and water price. The last and one of the most important factors related to the location is the outside temperature. Since, when the temperature goes high the need for cooling increases as well. Cooling system is an important parameter in the data centers, which its energy consumption increases as outside temperature increases. Indeed, high temperature leads to need for more chillers and more chillers increases data center's total energy consumption. This situation eventually leads to higher PUE and energy consumption, which indirectly increases carbon footprint. Goiri et al. [38] propose a framework to find the most optimum location for the data center to minimize the total costs. Explicit decrease in data center's cost, leads to indirect decrease in energy consumption and carbon footprint.

In order to increase the use of renewable energies, Berral et al. [62] propose a framework to find the best location to site the data centers and renewable power plants, solar and wind in their work. In the meantime, their objective is reducing total cost for building these infrastructures to support cloud HPC services with different amounts of renewable usage. Berral et al. divided the costs of building green cloud services into capital (CAPEX) and operational (OPEX) costs and CAPEX itself is divided to costs dependent and the costs that are independent to the number of servers to be hosted. Independent CAPEX costs are cost of bringing brown energy to the data center and connecting to the

backbone network. Land cost, building green power plants, cooling infrastructure, batteries, networking equipment, and servers are part of the dependent CAPEX costs. Costs incurred during the life cycle of the data center, such as network bandwidth and amount of brown energy usage are part of the OPEX. Brown energy consumption is the total energy needed by the servers and overhead parts, such as cooling and networking, minus energy derived from renewables. To calculate the overhead energy, Berral et al. [62] use PUE as a parameter related to the location temperature. It should be noted that temperature is not the only parameter that affects PUE, data center load is also an important parameter that changes PUE value [37].

In order to take the most of the generated renewable energy in different data centers, Berral et al. [62] compare different approaches such as net metering, which is directing the excess renewable energy into the grid and mix it with brown energy, using batteries and having storage for renewables or not having any storage and migrating the load to the sites with available solar or wind. One of the shortcomings of their work is neglecting the network delay and amount of energy consumed due to VM migration, as the data centers are scattered at different geographical locations. Moreover, all the data in this system are replicated at all the sites, which itself imposes overhead and increases energy consumption.

Table 2.1: Summary of various techniques for energy and carbon-efficient resource management in cloud data centers

Project Name	**Goal**	**Architecture**	**Technique**	**Carbon-Aware**
Dynamic right-sizing on-line algorithm, Lin. et al. [26]	Minimize energy consumption and total cost	Single Data Center	Online prediction algorithms for the number of required servers for the incoming workload	No
Green open cloud framework, Lefevre et al. [27]	Minimize energy consumption	Single Data Center	Predict the number of switched-on servers through providing in-advance reservation for users	No
Prediction-based Algorithms, Aksanli et al. [28]	Maximize renewable energy usage and minimize number of job cancellation	Single Data Center	Use prediction-based algorithms to run the tasks (mainly batch jobs) in the presence of renewable energies	Yes

Table 2.1: Summary of various techniques for energy and carbon-efficient resource management in cloud data centers (continued)

Project Name	Goal	Architecture	Technique	Carbon-Aware
GreenSlot scheduler, Goiri et al. [31]	Maximize renewable energy usage and minimize cost of using brown energies	Single Data Center	Prediction-based algorithms for the availability of solar energy and suspending the batch jobs in the absence of green energy	Yes
Multi-dimensional energy-efficient resource allocation (MERA) algorithm, Goudarzi et al. [33]	Minimize energy consumption and maximize servers' utilization	Single Data Center	VM placement heuristic to split the VMs and place them on a server with the least energy consumption	No
Multi-objective VM placement, Xu et al. [34]	Minimize power consumption, resource wastage, and the maximum temperature on the servers	Single Data Center	Data center global controller places the VMs based on a multi-objective algorithm to provide balance between power consumption and temperature	No
Green SLA service class, Haque et al. [35]	Explicit SLA to guarantee a minimum of renewable energy usage to run the workload	Single Data Center	Power distribution infrastructure to support the service and optimization based policies to maximize cloud provider's profit while meeting user's green SLA requirements	Yes
Cost-aware VM placement problem (CAVP), Chen et al. [39]	Minimize the operating cost	Distributed Data centers	VM Placement using meta-heuristic algorithms, considering different electricity prices and WAN communication cost	No
Energy model for request mapping, Qureshi et al. [40]	Minimize electricity cost	Distributed Data centers	Request routing to data centers with lower energy price using geographical and temporal variations	No
Free Lunch architecture, Akoush et al.[41]	Maximize renewable energy consumption	Distributed Data centers	VM migration and execution between data center sites considering renewable energy availability	Yes
Framework for load distribution across data centers, Le et al. [42]	Minimize brown energy consumption and cost	Distributed Data Centers	User request is submitted to the data center with access to the green energy source and least electricity price	Yes
Geographical load balancing (GLB) algorithm, Liu et al. [44]	Minimize brown energy consumption	Distributed Data Centers	Use the optimal mix of renewable energies (solar/wind) and energy storage in data centers to eliminate brown energy consumption	Yes

Table 2.1: Summary of various techniques for energy and carbon-efficient resource management in cloud data centers (continued)

Project Name	Goal	Architecture	Technique	Carbon-Aware
Online global load balancing algorithms, Lin et al. [45]	Minimize brown energy consumption and cost	Distributed Data Centers	Route requests to the data centers with available renewable energy using online algorithms	Yes
GreenWare middleware, Zhang et al. [43]	Maximize renewable energy usage	Distributed Data Centers	Submit the requests to the data center site with available renewable energy, while meeting provider's budget cost constraint	Yes
Environment-conscious meta-scheduler, Garg et al. [47]	Minimize carbon emission and maximize cloud provider profit	Distributed Data Centers	Near-optimal scheduling policies to send HPC applications to the data center with the least carbon emission and maximum profit, considering application deadline	Yes
Carbon-aware green cloud architecture, Garg et al.[48]	Minimize energy consumption and carbon footprint	Distributed Data Centers	Submit the user requests to the data center with the least carbon footprint, considering user deadline	Yes
MinBrown workload scheduling algorithm, Chen et al. [49]	Minimize brown energy consumption	Distributed Data Centers	Copy the data in all the data centers, then based on the request deadline and the data center with least brown energy consumption executes the request	Yes
Federated CLEVER-based cloud environment, Celesti et al. [50]	Minimize brown energy consumption and cost	Distributed Data Centers	Allocate the VM request to the cloud data center with the highest amount of photovoltaic energy and lowest cost	Yes
Temperature-aware workload management, Xu et al. [51]	Minimize cooling energy and energy cost	Distributed Data Centers	Joint optimization of reducing cooling energy by routing requests to the site with lower ambient temperature and dynamic resource allocation of batch workloads due to their elastic nature	No
Provably-efficient on-line algorithm (GreFar), Ren et al. [54]	Minimize energy cost	Distributed Data Centers	Use servers' energy efficiency information and places with low electricity prices to schedule batch jobs and if necessary suspending the jobs	No

Table 2.1: Summary of various techniques for energy and carbon-efficient resource management in cloud data centers (continued)

Project Name	Goal	Architecture	Technique	Carbon-Aware
Optimization-based framework, Le et al. [55]	Minimize cost and brown energy consumption	Distributed Data Centers	Distribute the Internet services to the data centers considering different electricity prices, data center location with different time zones, and access to green energy sources	No
Dynamic load distribution policies and cooling strategies, Le et al. [58]	Minimize cost	Distributed Data Centers	Intelligent placement of the VM requests to the data centers considering data centers geographical location, time zone, energy price, peak power charges, and cooling system energy consumption	No
Online job-migration, Buchbinder et al. [60]	Minimize cost	Distributed Data Centers	On-line migration of running jobs to the data center with lowest energy price, while considering transport network costs	No
Spatio-temporal load balancing, Luo et al. [61]	Minimize cost	Distributed Data Centers	Route the incoming requests to the data centers considering spatial and temporal variation of electricity price	No
Data centers' intelligent placement, Goiri et al. [38]	Minimize cost, energy consumption, and carbon footprint	Distributed Data Centers	Find the best location for data center, considering location dependent and data center characteristics data	Yes
GreenNebula, a prototype for VM placement that follows-the-renewables, Berral et al. [62]	Minimizing data center and renewable power plant building costs	Distributed Data Centers	Find the best geographical location to build data centers and renewable power plants and migrate the VMs, whenever necessary, to use a certain amount of renewables (solar or wind)	Yes

2.5 Summary

In this chapter, we studied the research works in the area of energy and carbon footprint-aware resource management in cloud data centers. We first had an overview on the existing techniques in green cloud resource management with the focus on a single server and a single data center and the limitations facing these techniques, specially not being

able to harvest renewable energy sources at different locations. We then focused more specifically on the works considering geographically distributed cloud data centers, as nowadays most of the big cloud providers have data centers in different geographical locations for disaster recovery management, higher availability, and providing better quality of experience to users.

A large body of literature have focused on reducing the energy used within a single or multiple data centers without considering the energy sources and power usage effectiveness (PUE). We proposed a VM placement algorithm in Chapter 3 to increase the environmental sustainability that considers data centers carbon footprint rates and PUEs.

Moreover, some of the research works achieve energy efficient resource management through minimizing cost and the cost of brown energy usage, which indirectly could lead to less carbon footprint in the ecosystem. In Chapter 4, we investigated the parameters that have the biggest effect on the energy and carbon footprint costs. We proposed a new method that simultaneously considers the cost of overhead energy, servers' energy, and carbon footprint. The proposed VM placement method maximizes renewable energy utilization at each data center to minimize the total cost. We also presented efficient two-stage VM placement approaches that respond to dynamic PUEs.

In Chapter 5, we explored how much energy cost savings can be made knowing the future level of renewable energy in the data centers. We took advantage of migrating VMs to the data centers with excess renewable energy. We proposed two online deterministic algorithms, one with no future knowledge called deterministic and one with limited knowledge of the future renewable availability called future-aware. We studied the algorithms performance against the optimal offline algorithm with full knowledge of the future level of renewable energy. A short-term prediction model is proposed in Chapter 6 that helps the future-aware online deterministic algorithm to make informed decisions and migrate the VMs between data center sites in the absence of the renewable energy.

Chapter 3

Energy and Carbon-Efficient Placement of Virtual Machines

Due to the increasing use of cloud computing services and the amount of energy used by data centers, there is a growing interest in reducing energy consumption and carbon footprint of data centers. Cloud data centers use virtualization technology to host multiple virtual machines (VMs) on a single physical server. By applying efficient VM placement algorithms, cloud providers are able to enhance energy efficiency and reduce carbon footprint. Previous works have focused on reducing the energy used within a single or multiple data centers without considering their energy sources and Power Usage Effectiveness (PUE). In contrast, this chapter proposes a novel VM placement algorithm to increase the environmental sustainability by taking into account distributed data centers with different carbon footprint rates and PUEs. Simulation results show that the proposed algorithm reduces the CO_2 emission and power consumption, while it maintains the same level of quality of service compared to other competitive algorithms.

3.1 Introduction

THE information and communication technology (ICT) industry consumes an increasing amount of energy and most of it is consumed by data centers [63]. A major consequence of this amount of energy consumption by data centers is a significant increase in ecosystem carbon level. According to Gartner, the ICT industry produces 2% of global CO_2 emission, which places it on par with the aviation industry [64]. Therefore,

This chapter is derived from the publication: Atefeh Khosravi, Saurabh Kumar Garg, and Rajkumar Buyya, "*Energy and Carbon-Efficient Placement of Virtual Machines in Distributed Cloud Data Centers*", Proceedings of the 19^{th} International European Conference on Parallel and Distributed Computing (Euro-Par), Pages 317-328, Aachen, Germany, 2013.

reducing even a small fraction of the energy consumption in ICT, results in considerable savings in financial and carbon emission of the ecosystem.

Cloud computing offers a wide range of services and applications to its users. Three main services that clouds provide are infrastructure, platform, and software as a service. Infrastructure as a service (IaaS) allows users to run their applications in form of virtual machines (VMs) on a shared infrastructure. Cloud data centers take advantage of virtualization technology [21] to share a physical server's resources among multiple VMs. Each VM has its own characteristics and depending on the resource usage, it consumes energy and leaves carbon footprint. By the arrival of each VM request, the cloud manager selects the physical resource to instantiate the request. VM placement in cloud computing system is a complex task and if cannot be done effectively, it leads to high energy usage and high carbon footprint.

Thus, wisely taking into account parameters that affect VM placement and physical server selection results in less energy consumption and less carbon footprint. Distributed cloud data centers, alongside bringing high availability and disaster recovery, provide the opportunity to have different energy sources. *Carbon footprint rate* of energy sources is an important parameter, since data centers use electricity driven by these sources to run VMs. By having different energy sources in different data center sites or within a data center site, cloud providers should increase the use of more clean and off-grid renewable energies [65]. *Power usage effectiveness (PUE)* is coined by the Green Grid consortium [66] and indicates the energy efficiency of a data center. PUE is a ratio of total power consumed by the data center to its power consumed by IT devices. Providers can consider PUE as a parameter to perform VM placement among different data center sites. *Proportional power* is another parameter that can be taken into account for VM placement. Server proportional power has a cubic relation with CPU frequency [67]. Therefore, considering the increase in CPU frequency, which is related to increase in CPU utilization upon new request arrival, will have a great impact on the amount of energy consumption in data centers.

This chapter proposes a VM placement algorithm by considering distributed cloud data centers with the objective of minimizing carbon footprint. Our proposed cloud com-

puting system, Energy and Carbon-Efficient (ECE) cloud architecture, benefits from distributed cloud data centers with different carbon footprint rates, PUE value, and different physical servers' proportional power. ECE cloud architecture places VM requests in the best suited data center site and physical server. The main contributions of this chapter are: an energy and carbon-efficient cloud architecture, based on distributed cloud data centers; an efficient VM placement algorithm that integrates energy efficiency and carbon footprint parameters; a comprehensive comparison on carbon footprint and power consumption for different VM placement algorithms with respect to quality of service (number of rejected VMs).

The reminder of the chapter is organized as follows. In Section 3.2 the related work is discussed. Section 3.3 presents the proposed cloud architecture with its components, VM placement algorithm, and formulates the objective. Finally, the performance evaluation results and the experimental environment are presented in Section 3.4 followed by a summary in Section 3.5.

3.2 Related Work

There is a growing body of literature that aims to reduce the amount of carbon dioxide of cloud services in data centers. Most of the works in this area focus on reducing the energy consumption in a single data center or considering the data center hardware aspects [68] [23]. Well-known technologies that data centers benefit from by applying virtualization technology [21] are VMs migration [24] and consolidation [22]. The main problem with migration and consolidation is that they are complex and, due to the need for resuming and suspending VMs cause overhead to the system [69]. Moreover, these technologies act reactive whereas applying preventive technologies are more efficient.

As idle servers consume almost half of the power when they are in the peak power state [25], work by Lin et al. [26] uses a dynamic right-sizing on-line algorithm to predict the number of active servers that are needed for the arriving workload to the data center. Based on their experiments, dynamic right-sizing can achieve significant energy savings in the data center, but it requires servers to have different power levels and be able to

transit to different states. A similar work done by Lefevre et al. [27] proposes Green Open Cloud (GOC) architecture, with advance resource reservation for users to improve the prediction of the arrival requests.

The above mentioned technologies are adopted within a data center and intend to reduce the energy consumption, whilst they do not particularly consider carbon emission. Reducing data center energy consumption does not necessarily lead to reduce in carbon footprint. Works by Aksanli et al. [28] and Goiri et al. [31] consider the availability of both non-polluting (green) and polluting (brown) energy sources in a single data center. They use prediction-based scheduling algorithms to increase usage of green energy sources.

Liu et al. [44] consider reducing the carbon footprint of data centers by considering multiple data center sites. They proposed an algorithm to efficiently use the renewable energies, such as wind and solar, in different places. This algorithm uses the idea of geographic diversity of data center sites and unpredictability of renewable energies to find the optimal percentage of wind/solar energies in order to reduce the brown energy consumption. Garg et al. [48] also consider reducing carbon footprint of cloud data center sites. They proposed a novel carbon-aware green cloud architecture, which uses two directories for cloud providers to register their offered services.

Our work is different from the previous works, since we address the problem of increase in carbon footprint of the cloud data centers by performing efficient VM placement. Our proposed method accommodates VM requests by considering distributed data center sites of a cloud provider, with various energy sources and carbon footprint rates. Moreover, we consider data centers' PUE, physical servers' proportional power usage, and user VM requests of different types. Finally, we present an energy and carbon-efficient algorithm that uses two level decision making for VM placement.

3.3 System Model

In this section, Energy and Carbon-Efficient (ECE) cloud architecture is described. This architecture assures system quality of service, while minimizing the cloud carbon footprint by applying an energy and carbon-efficient heuristic for VM placement.

3.3.1 ECE Cloud Architecture

The proposed architecture is represented in Figure 3.1. The system consists of the following components and symbols used in this chapter are presented in Table 3.1:

Table 3.1: Description of Symbols.

Symbol	Description	Symbol	Description
d	Number of Data Center Sites	P	Proportional Power
c	Number of Clusters at each Data Center	P_{fixed}	Server Power Consumption in Idle State
h	Number of Hosts at each Cluster	P_f	Server Power Consumption at Frequency f
cf	Data Center/Cluster Carbon Footprint Rate	f_u	CPU Operating Frequency at Utilization u
CF	Cloud Total Carbon Footprint	$type$	Virtual Machine Instance Type
ht	Virtual Machine Holding Time	$core$, pu	CPU Cores and Total Processing Unit
ram,$storage$	RAM and Storage	bw	Network Bandwidth

Cloud Users: Cloud Users send their VM requests based on predefined requirements to the cloud provider. Virtual machine types and configurations are inspired by Amazon Elastic Compute Cloud (EC2) [70]. The expected requirements for each VM are specified by its predefined configurations in terms of required number of cores, processing unit of each core, storage, RAM, and network bandwidth. In addition, holding time of a VM depends on the application runs on that VM. We consider two types of applications in this chapter: bag-of-tasks and web-requests. Every requested VM by users has the following requirements: $(ht, type)$, where each $type$ consists of the following components: $\{cores, pu, ram, storage, bw\}$. Cloud computing system load at time t, according to the running VMs, is represented as:

$$load = \sum_{i=1}^{d} \sum_{j=1}^{c} \sum_{k=1}^{h} vm_{(i,j,k,t)}.$$

Cloud Provider: A cloud provider has several geographically distributed data center sites. Each data center is composed of several clusters with various heterogeneous physical servers. Physical servers are characterised by CPU cores, CPU processing unit,

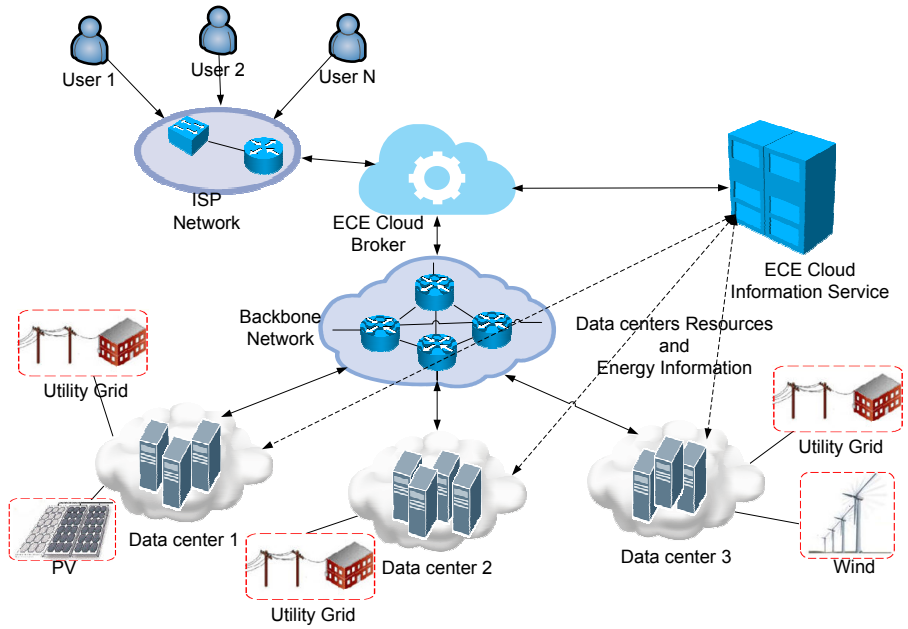


Figure 3.1: Energy and Carbon-Efficient (ECE) Cloud Architecture.

amount of RAM, storage, and network bandwidth. In addition to the physical servers configuration, each data center has its own energy-related parameters, shown by PUE and proportional power. Moreover, each data center can have one or more energy sources with different carbon footprint rates.

ECE Cloud Information Service: Each data center site registers its characteristics in the ECE cloud information service (ECE-CIS) and they keep their information updated. This information includes available physical resources and energy related parameters; such as data center PUE, energy source(s), carbon footprint rate, and physical servers' current utilization and power consumption. Cloud broker uses this information to perform ECE VM placement in cloud computing environment.

ECE Cloud Broker: ECE cloud Broker is the cloud provider's interface with cloud users. It receives user requests and schedules them based on their predefined requirements. Despite users request scheduling, broker should also ensure energy efficient data centers with minimum carbon footprint for cloud providers.

Resources on the cloud provider are physical servers in the clusters within each data center. The broker receives the current status of data centers' physical resources and their

energy information from ECE-CIS, and based on this information, assigns the VM to a physical server in a data center site. Based on [71], in today's Internet and core networks design, average number of hops a packet passes from source to destination is between 12-14 hops. Therefore, we can have data center site selection without considering network distance; especially for sites that are located in a region, such as different states in USA, as we considered in this chapter.

3.3.2 Placement Decision

As stated before, the broker makes the placement decision based on the data centers' power usage effectiveness (PUE), energy sources carbon footprint rate, and proportional power.

The PUE indicates the energy efficiency of a data center and is a metric to compare different data center designs in terms of electricity consumption. Data center's PUE is calculated as:

$$PUE_i = \frac{Datacenter_i TotalPowerConsumption}{Datacenter_i ITDevicesPowerConsumption},$$

where the total power consumption is sum of power drawn by cooling, lightening, and IT equipment. PUE is a value larger than 1 ($PUE \geq 1$). PUE of 1.0 means 100% of the data center's electricity goes to the IT part and is ideal for any data center, but is unattainable pragmatically. In other words, the smaller the PUE, the more energy efficient the data center.

Data center proportional power is the next important metric in physical server selection. According to the experiments by Lien et al. [67] server's power consumption depends on the system base power and the CPU frequency, and the CPU frequency itself depends on the CPU utilization. The data center proportional power, also known as dynamic power, is calculated as: $P = P_{fixed} + P_f \times f_u^3$. The power consumption for a VM on physical server k in cluster j of data center i at time t is modeled as: $P(vm_{(i,j,k,t)})$.

According to the above mentioned metrics the objective is to minimize total carbon

footprint of the cloud provider, CF, for time interval $[1, T]$, and is computed as follows:

$$CF = \sum_{t=1}^{T} \sum_{i=1}^{d} (PUE_i \times \sum_{j=1}^{c} (cf_j \times \sum_{k=1}^{h} (P(vm_{(i,j,k,t)}) \times ht))),$$

subject to following constraints:

$$\sum_{i=1}^{d} \sum_{j=1}^{c} \sum_{k=1}^{h} vm^{core}_{(i,j,k)} \leq host^{core}_{(i,j,k)}, \qquad \sum_{i=1}^{d} \sum_{j=1}^{c} \sum_{k=1}^{h} vm^{pu}_{(i,j,k)} \leq host^{pu}_{(i,j,k)},$$

$$\sum_{i=1}^{d} \sum_{j=1}^{c} \sum_{k=1}^{h} vm^{ram}_{(i,j,k)} \leq host^{ram}_{(i,j,k)}, \qquad \sum_{i=1}^{d} \sum_{j=1}^{c} \sum_{k=1}^{h} vm^{storage}_{(i,j,k)} \leq host^{storage}_{(i,j,k)}.$$

The above mentioned constraints ensure that allocated resources to the VMs on a physical server do not exceed the total capacity of the server.

3.3.3 Energy and Carbon-Efficient (ECE) Heuristic for VM Placement

By the arrival of each VM request the broker has $(d \times c \times h)$ different VM placement options. The VM placement problem can be seen as a bin-packing problem with different bin sizes (physical servers). Therefore, we propose the Energy and Carbon-Efficient (ECE) VM placement algorithm (Algorithm 1), which is a derivation of the best-fit heuristic to place the VMs in the data center, cluster, and host with the minimum carbon footprint, PUE, and minimum increase in physical server's power consumption.

The broker receives a VM request and selects the best physical server for the VM. Its objective is to minimize the data centers' carbon footprint and accordingly power consumption. Therefore, ECE placement algorithm gets data centers' resources and energy status from ECE-CIS, upon the arrival of a new VM request (Line 2). According to the received information, ECE adds the clusters of all the data centers into an aggregated cluster list (Lines 3-4), and sorts the new list based on the minimum $(PUE \times cf)$ (Line 5). By receiving the data centers and clusters status, ECE calculates the amount of power consumption that will be added to each host after initiating the new VM (Lines 8-10). Afterwards, ECE sorts the hosts list based on the estimated $\triangle P$ (Line 11), and if the host has enough resources for the VM (Line 13), it submits the VM to the selected data center,

Algorithm 1: Energy and Carbon-Efficient (ECE) VM Placement Algorithm

```
   Input: datacenerList, clusterList, hostList
   Output: destination
1  while vmRequest do
2      Get data centers' Information from ECE-CIS;
3      foreach data center in data centerList do
4          Add clusterList into aggregateClusterList;
5      Sort aggregateClusterList in an ascending order of (PUE × cf);
6      foreach cluster in aggregateClusterList do
7          foreach host in hostList do
8              P1 ← Get current hostDynamicPower;
9              P2 ← Calculate hostDynamicPower after initiating the vm;
10             △P ← P2 − P1;
11         Sort hostList in an ascending order of △P;
12         foreach host in hostList do
13             if host is suitable for vm then
14                 destination ← (data center, cluster, host);
15                 return destination;
16     destination ← null; //rejection of request;
17     return destination;
```

cluster, and host.

In order to show the time complexity of Algorithm 1, we consider v VM requests. Line 3-4 take $O(d)$, and the sort function at Line 5 can be done in $O(dc\log(dc))$. Lines 7-9 need $O(h)$ time, and the sort function for hosts at Line 11 needs $O(h\log(h))$ to be done. Lines 12-15 take $O(h)$, in the worst case. Thus, the total running time of the algorithm is $O(v(d + dc\log(dc) + dc(h + h\log(h) + h)))$. Since there are small number of data center sites and clusters (dc) for a cloud provider, the complexity of this algorithm is dominated by the number of VM requests and hosts sort function. The total time complexity of the algorithm is $O(vdch\log(h)))$.

3.4 Performance Evaluation

We use the CloudSim toolkit [72] to evaluate the cloud computing virtualized environment. We have extended CloudSim to enable energy and carbon-efficient VM placement

Table 3.2: Data Centers Characteristics.

Data Center Site	**PUE**	**Carbon Footprint Rate** (Tons/MWh)
DC1 -Oregan, USA	1.56	0.124, 0.147
DC2 -California, USA	1.7	0.350, 0.658
DC3 -Virginia, USA	1.9	0.466, 0.782
DC4 -Dallas, USA	2.1	0.678, 0.730

simulations. Apart from being aware of data center's PUE, carbon footprint rate, and dynamic power, the extended package has the ability to simulate dynamic VM requests with different instance types.

In order to evaluate the proposed algorithm, we modeled an IaaS provider with 4 data center sites, and each site with 90 heterogeneous physical servers. Each data center has a unique PUE value and 2 clusters with different carbon footprint rates. Table 3.2 shows data centers' PUE value and carbon footprint rate for different group of clusters. The PUE value is based on the work by Greenberg et al. [73], and is in the range [1.56, 2.1]. Data centers' carbon footprint rates, the last column of Table 3.2, are derived based on the information from US Department of Energy, Appendix F, Electricity Emission Factors [16]. Carbon footprint rate is based on the average carbon dioxide emission of total electric sector generation for specified state-based regions and include transmission and distribution losses incurred in delivering electricity to the point of use. In the simulation, we use 5 different physical servers whose characteristics are given in Table 3.3. Moreover, we use 2 different power models for servers in order to support hardware heterogeneity. According to the linear relationship between CPU utilization and frequency, and the cubic relation between CPU frequency and system proportional power, the following is the 2 power models for the platforms:

$$\text{CPU Frequency(in GHz): } \{f(u) : (1.4, 1.57, 1.74, 1.91, 2.08, 2.25, 2.42, 2.6, 2.77, 2.94, 3.11)\}$$

$$\text{Power Model1(in Watt): } \{P_f : (60, 63, 66.8, 71.3, 76.8, 83.2, 90.7, 100, 111.5, 125.4, 140.7)\}$$

$$\text{Power Model2(in Watt): } \{P_f : (41.6, 46.7, 52.3, 57.9, 65.4, 73, 80.7, 89.5, 99.6, 105, 113)\}$$

VM characteristics are inspired by Amazon EC2 instance types given in Table 3.4. The

Table 3.3: Platform Types Characteristics.

Platform Type	**Number of Cores**	**Core Speed** (GHz)	**Memory** (GB)	**Storage** (GB)	**Network Band-width** (Mbps)	**Bits**	**Power Model**
Platform1	2	2	16	2000	1000	B32	PowerModel1
Platform2	4	4	32	6000	1000	B64	PowerModel1
Platform3	8	4	32	7000	2000	B64	PowerModel2
Platform4	8	8	64	7000	4000	B64	PowerModel2
Platform5	8	16	128	9000	4000	B64	PowerModel2

Table 3.4: VM Types and Simulated User Types; (Bag-of-Task Users (BT) and Web-Request Users (WR)).

VM Type		**Number of Cores**	**Core Speed** (GHz)	**Memory** (MB)	**Storage** (GB)	**Network Band-width** (Mbps)	**Bits**	**Probability and UserType**
Standard Instances	M1Small	1	1	1740	160	500	B32	0.25-BT
	M1Large	2	4	7680	850	500	B64	0.12-WR 0.25-BT
	M1XLarge	4	8	15360	1690	1000	B64	0.08-WR
High Memory Instances	M2XLarge	2	6.5	17510	420	1000	B64	0.12-WR
	M22XLarge	4	13	35020	850	1000	B64	0.08-WR
High CPU Instances	C1Medium	2	5	1740	320	500	B32	0.1-BT

physical resources to the VMs are allocated based on the VM resource requirements and all the VMs are considered to perform at the maximum utilization during their lifetime. The VM type and the number of VMs requested by users depend on the user type (bag-of-tasks or web-requests), and are based on the related probabilities. The VM type related probability is shown in the last column of Table 3.4 and is derived from the work by Mills et al. [74].

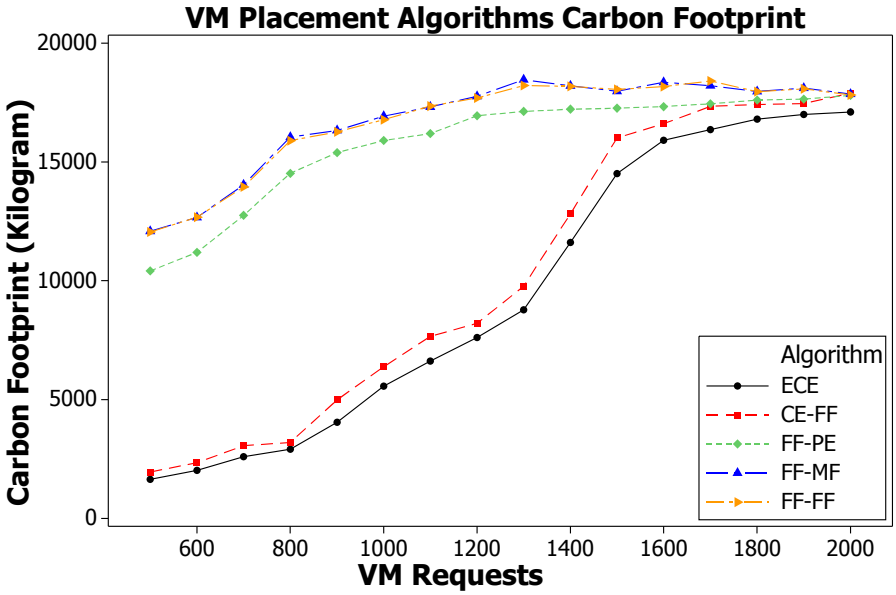
In order to generate the workload, we need VM requests arrival rate and holding time. The Lublin-Feitelson [13] workload model is employed to generate the bag-of-tasks VM requests. We take benefit of Lublin to set arrival request parameters, including simulation duration, number of requests, requests arrival time, and request holding time. We consider each generated request in Lublin as a VM request. In order to generate VMs

with longer holding time, we increased the first parameter of the Gamma distribution and left other Lublin parameters with their default value. To generate the web-requests, we use the same arrival time model of bag-of-tasks requests, and for the holding time we use a hyper gamma distribution with expectation value 73 and variance 165. For both workloads, we omit 5% of created requests at the start (warm-up period) and end (cool-down period) of the simulation to get a steady environment. We apply 240-hour long workload with different number of requests. Finally, for the purpose of accuracy, each experiment is repeated 30 times and the mean is reported for measured values for experimental results.

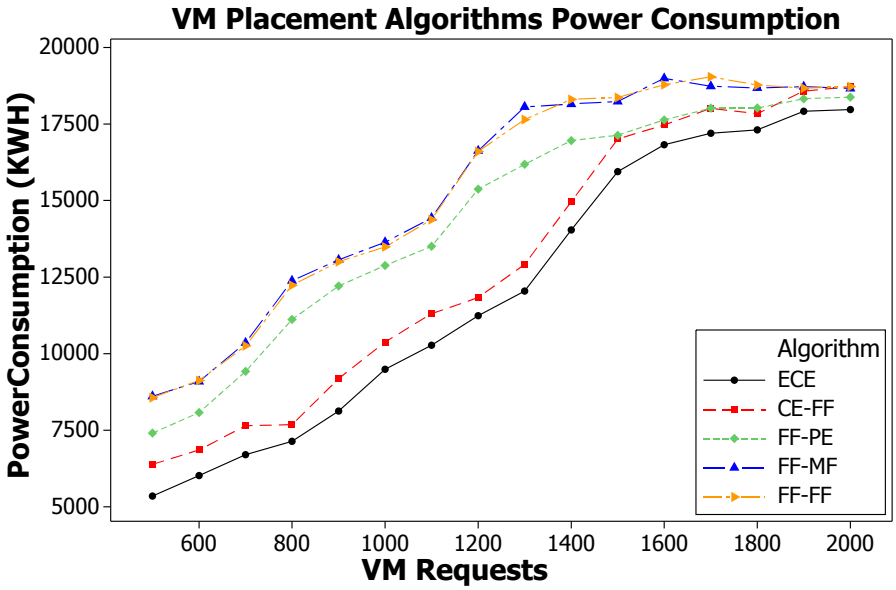
3.4.1 Results

We use the described workload data to compare the proposed VM placement algorithm with respect to carbon and power efficiency with four competing algorithms. The first algorithm is a version of ECE, that its data center and cluster selection is same as ECE, and uses first-fit bin-packing for host selection. We refer to this algorithm as Carbon-Efficient First-Fit (CE-FF). The other group of algorithms are three bin-packing heuristics that use first-fit heuristic for data center/cluster selection, without considering carbon footprint parameters. First-Fit Power-Efficient (FF-PE) uses power-efficient policy for host selection (same as ECE host selection). First-Fit Most-Full First (FF-MF) selects the physical server with least available resources. Finally, the last algorithm uses first-fit heuristic for data center, cluster, and host selection (FF-FF).

Figure 3.2a illustrates the carbon footprint of ECE in comparison with other placement algorithms under different number of VM requests. Based on the experiments, as the number of VMs increases, the system utilization increases as well to the point that system performs with highest utilization and reaches to the saturation point. Therefore, increase in system load leads to increase in the total carbon footprint in data centers. Based on the Figure 3.2a, ECE in comparison to CE-FF (carbon-efficient) and other heuristics (non carbon-efficient) reduces carbon footprint with an average of 10% and 45%, respectively. The same behaviour can be seen for the data centers' power consumption in Figure 3.2b. This figure shows total power consumed by each server within each data center



(a) Carbon Footprint.



(b) Power Consumption.

Figure 3.2: Comparison of ECE Algorithm with other VM Placement Algorithms.

to support the VMs, as it has been discussed in Section 3.3.2. The ECE algorithm has lower power consumption in comparison to the other algorithms and consumes on average 8% and 20% less power than CE-FF and other heuristics placement algorithms, respectively. Considering the differences between algorithms behaviour in both figures,

we can infer that just considering power-efficient parameters is not enough to reduce the data centers' carbon footprint. However, taking into consideration data centers' energy and carbon rate parameters, at the same time, leads to significant reduction in terms of cloud computing system carbon footprint and consumed power.

Table 3.5: SLA Violation for Different VM Placement Algorithms.

VM Placement Algorithm	**SLA Violation Under Different VM Requests**					
	1000	**1200**	**1400**	**1600**	**1800**	**2000**
ECE	0.0%	0.05%	0.4%	2.9%	8.6%	13.0%
CE-FF	0.0%	0.0%	0.3%	0.7%	6.0%	11.4%
FF-PE	0.0%	0.0%	0.3%	2.5%	9.4%	15.3%
FF-MF	0.0%	0.0%	0.2%	2.5%	9.7%	15.3%
FF-FF	0.0%	0.0%	0.1%	2.6%	9.7%	15.3%

Table 3.5 shows the SLA violation under different system loads for different VM placement algorithms. It shows that, the SLA violation (number of rejected VMs) under low system load for ECE is slightly higher than the other algorithms. However, by increasing system load, ECE will have lower SLA violation. Overall, all the VM placement algorithms have close values for violation, while ECE considerably reduces carbon footprint and power consumption.

3.5 Summary

In this chapter, the problem of VM placement to reduce cloud computing energy consumption and carbon footprint is investigated. We used ECE cloud information service (ECE-CIS), as part of next generation cloud computing environment. ECE-CIS obtains energy and carbon related information from data centers and enables the broker to carry out carbon and power-efficient VM placement. We introduced the energy and carbon-efficient (ECE) VM placement algorithm, and compared it with a carbon-efficient algorithm (CE-FF) and three other heuristic algorithms (FF-PE, FF-MF, FF-FF). We performed the simulations by extending CloudSim and used different VM instance types with different holding times for the system workload. Based on the experiment results, ECE can on average save up to 10% and 45% carbon footprint in the ecosystem in comparison to CE-FF and three other heuristics, respectively, while keeping SLA violation level as the

same. Moreover, ECE reduces the power consumption in data centers by an average of 8% and 20% in comparison to CE-FF and other three algorithms, respectively; which illustrates the importance of considering data centers' carbon footprint rate and PUE to reduce cloud computing carbon footprint.

In the next chapter, we study how energy efficient resource management can be achieved through minimizing cost and the cost of brown energy usage. Chapter 4 investigates parameters that have the biggest effect on the energy and carbon footprint cost. It proposes VM placement method to maximize renewable energy utilization and minimize the total cost. It also presents efficient two-stage VM placement approaches that respond to dynamic PUEs.

Chapter 4

Dynamic VM Placement Method for Minimizing Energy and Carbon Cost

Cloud data centers consume a large amount of energy that leads to a high carbon footprint. Taking into account a carbon tax imposed on the emitted carbon makes energy and carbon cost play a major role in data centers' operational costs. To address this challenge, we investigate parameters that have the biggest effect on energy and carbon footprint cost to propose more efficient VM placement approaches. We formulate the total energy cost as a function of the energy consumed by servers plus overhead energy, which is computed through power usage effectiveness (PUE) metric as a function of IT load and outside temperature. Furthermore, we consider that data center sites have access to renewable energy sources. This helps to reduce their reliance on "brown" electricity delivered by off-site providers, which is typically drawn from polluting sources. We then propose multiple VM placement approaches to evaluate their performance and identify the parameters with the greatest impact on the total renewable and brown energy consumption, carbon footprint, and cost. The results show that the approach which considers dynamic PUE, renewable energy sources, and changes in the total energy consumption outperforms the others while still meeting cloud users' service level agreements.

4.1 Introduction

CLOUD computing is considered a big step towards the long held dream of delivering computing as a utility to users [75]. The cloud enables access to hardware resources, infrastructure, and software anytime and anywhere on a pay-as-you-go model.

This chapter is derived from the publication: Atefeh Khosravi, Lachlan L. H. Andrew, and Rajkumar Buyya. *"Dynamic VM Placement Method for Minimizing Energy and Carbon Cost in Geographically Distributed Cloud Data Centers"*, IEEE Transactions on Sustainable Computing (T-SUSC), Volume 2, Number 2, Pages: 183-196, IEEE Computer Society Press, USA, 2017.

Services by cloud computing are delivered by data centers that are distributed across the world, which can host small numbers to thousands of servers. A major issue with these data centers is that they consume a large amount of energy. According to a report from NRDC [11], US data centers power consumption estimation alone in 2013 was 91 billion kilowatt-hours of electricity. This is equivalent to two years' power consumption of New York City's households and is estimated to increase to 140 billion kilowatt hours by 2020, which is responsible for emission of nearly 150 million tons of carbon pollution.

The high energy consumption by data centers incurs high costs to cloud providers, since energy related costs are the most significant cost for a data center [76]. Furthermore, to enforce the environmental sustainability, some countries set carbon tax on the emitted CO_2 [77]. Therefore, monitoring the amount of energy consumed by a data center and the source of the energy, which directly affects the carbon footprint and carbon tax, helps cloud providers to reduce the energy and carbon cost as a major sector of their total cost.

In this chapter, we investigate parameters that affect the total cost associated with the energy consumption and carbon footprint for a cloud provider. Here, we only consider the cost of these two parameters, unless otherwise mentioned. A cloud provider often maintains geographically distributed data center sites, similar to popular cloud providers (e.g., Amazon, Google, and Microsoft). Having several sites not only increases the availability, it also gives the cloud provider the option of choosing the destination site based on different criteria upon the reception of the user request (virtual machine requests in this chapter). There are different challenges a cloud provider faces to make the decision regarding VM placement and scheduling. In this chapter, we study the selection process among several data center sites. Each data center can get its electricity from different electricity providers, we refer to this as off-site brown energy sources, or even can draw the required electricity from on-site renewable ("green") energy sources, such as solar and wind. Having data center sites that can get their power from renewable sources partially or completely helps the provider decrease its dependency on the electricity drawn from off-site grids, which is costly and less clean. Secondly, off-site brown energy at different locations have different carbon intensities and carbon taxes. Therefore, by the change of the availability of renewable energy during the day and in the case that they are not

available, cloud provider can select the cleanest source of electricity with less carbon tax. The third advantage of having different energy sources at different locations is changes in electricity price, as we consider variable energy pricing during times of the day, i.e., on-peak and off-peak prices.

The last and one of the most important parameters that affects data centers energy consumption, carbon footprint, and their associated cost is the overhead power, e.g., power supplies, cooling, lightning, and UPS. The metric used to demonstrate the overhead is known as Power Usage Effectiveness (PUE) that is defined by The Green Grid consortium [78]. PUE is equal to the data center's total power consumption, which is the input power that goes to the data center, divided by the IT devices power consumption (PUE=TotalPower/ITDevicesPower). If PUE is equal to 1 it means that the data center is perfectly efficient, which is not practically attainable. An increase in PUE indicates more waste of power to support IT devices in the data center. Although state of the art cloud-scale data centers can achieve a PUE of 1.1 [79] or 1.2 [80], cloud providers often collocate with smaller data centers, which can still have PUEs up to 2 [81]. To increase a data center's efficiency, we should identify variables that have the highest impact on the increase of the system's overhead power. The main variable that affects efficiency and PUE value is IT load. When the IT load is increased, CPUs perform in higher frequencies and servers consume more power. This leads to increase in data center's overall load and inside temperature; accordingly the need arises for more power for the cooling of the infrastructure. The second important variable that affects PUE is the outside temperature, which has a great effect on the cooling system power consumption. As outside temperature increases, the data center needs to use the chillers along with the computer room air handler (CRAH), which leads to a significant increase in the power consumption and PUE value. We exploit a model for PUE as a function of IT load and outside temperature and perform VM placement based on dynamic changes of PUE.

The **key contribution** of this chapter is a new method for the initial placement of VMs in geographically distributed cloud data centers that simultaneously considers the cost of 1) overhead energy 2) servers' energy and 3) carbon footprint. Moreover, the proposed VM placement method maximizes renewable energy utilization at each data center to

minimize the total cost. Finally, we present efficient two-stage VM placement approaches that respond to dynamic PUEs. We also present variations of our method, which explore the effects of different parameters in minimizing energy and carbon cost for a cloud computing environment. To achieve this, we have carried out the following:

- Developed an analytical model of the total cost incurred by the energy consumption and carbon footprint for the data centers.
- Modeled PUE as a function of IT load and outside temperature to incorporate overhead energy consumption, e.g., power supplies, cooling, lightning, and UPS, along with the energy consumed by the servers.
- Used different carbon intensities and carbon taxes for energy sources at each data center site.
- Analyzed the effect of distributing load among data center sites with access to intermittent renewable energy sources.

The reminder of the chapter is organized as follows: In Section 4.2, the related work is discussed. Section 4.3 discusses the system model, parameters, objective function and constraints. The proposed VM placement approaches are discussed in Section 4.4. The experimental environment and the performance analysis of the proposed VM placement approaches are presented in Section 4.5. Finally, a summary is depicted in Section 4.6.

4.2 Related Work

Over the last few years, there have been extensive studies on reducing energy consumption of cloud data centers. Recently, there has been much interest in reducing data center carbon footprints and energy consumption due to the environmental concerns (specifically around global warming), social pressure, and the prospect of a carbon tax. Most of the early work focuses on making a single server energy efficient by considering hardware aspects and using techniques such as CPU DVFS (dynamic voltage and frequency scaling) [68, 82]. Moreover, virtualization [21] as the foundation of cloud computing systems, enables consolidation [22] and VM migration [24]. There is ongoing research on

the later techniques, but the main issue is that they are reactive and require resume and suspension of VMs which cause overhead to the system [69]. Therefore, these techniques should be applied only when they are cost-effective. Lin et al. [26] and Shen et al. [83] used a pro-active technique, known as dynamic right-sizing, to predict the number of active servers needed to host the incoming workload. Since idle servers consume almost half of the peak power [25], this technique could reduce the energy consumption significantly. Lefevre et al. [27] proposed an advanced resource reservation architecture to have a better prediction of the incoming requests by users. The above-mentioned techniques are in the scope of a single data center and they only consider the aspect of reducing energy consumption; which does not necessarily lead to a reduction in the carbon footprint. Aksanli et al. [28] use predication-based algorithms to maximize the usage of renewable energy sources and in the meantime minimize the number of canceled jobs.

One of the first works to reduce cost and brown energy consumption by load distribution among several data center sites, is that of Le et al. [42]. Their work is based on considering the electricity price and energy source (green or brown) to calculate the number of requests each data center can host within a specific time period and budget. However, they do not differentiate among sites that have brown energy sources with different carbon emission rates. Further, the incoming workload is based on SaaS (Software as a Service) requests for Internet services with short processing times, usually in milliseconds. Liu et al. [44] consider geographical load balancing to minimize brown energy consumption as well. They use an optimal mix of renewable energy (solar and wind) along with energy storage in data centers to eliminate brown energy consumption. Lin et al. [45] extended the previous work to find the best estimate combination for solar and wind energy while having net-zero brown energy usage. The MinBrown workload scheduling algorithm is proposed by Chen et al. [49] to minimize brown energy consumption. This algorithm forwards the incoming request to all data centers, then based on the request deadline and brown energy consumption schedules request for execution. Celesti et al. [50] proposed a federated CLEVER-based cloud environment; which is based on allocation of VM requests to the cloud data center with the highest amount of solar energy and lowest cost. Le et al. [55] proposed an optimization-based

framework to minimize brown energy consumption and leverage green energy through distribution of the Internet services to the data centers, considering different electricity prices, data center location with different time zones, and access to green energy sources.

Le et al. [58] apply dynamic load distribution policies and cooling strategies to minimize the overall cloud provider's cost but places no cost on carbon emissions. Their work is based on intelligent placement of VM requests on data centers considering the geographical location, time zone, energy price, peak power charges, and cooling system energy consumption. Ren et al. [54] proposed a provably-efficient on-line algorithm (GreFar) with the objective to minimize energy cost. They use servers' energy efficiency information and locations with low electricity prices to schedule batch jobs and, if necessary, suspend the job and resume later. Work by Goiri et al. [38] aims to find the best place for a data center, based on geographical location and data center characteristics to minimize cost, energy consumption, and carbon footprint. Garg et al. [47] proposed an environment-conscious meta-scheduler to minimize carbon emission and maximize cloud provider's profit. They used near-optimal scheduling policies to send HPC (high performance computing) applications to the data center with the least carbon emission and maximum profit, considering applications deadline. They also address the issue of energy consumption and carbon footprint by proposing a novel green cloud architecture [48]. This architecture uses two directories so the cloud providers can register their offered services. A notable work by Buchbinder et al. [60] has the same objective of reducing the energy cost of a cloud provider with multiple data center sites. They perform on-line migration of running batch jobs among data center sites, taking advantage of dynamic energy pricing at different locations, while considering the network bandwidth costs among data centers and future changes in electricity price. Similarly, Giacobbe et al. [84] perform VM migration between cloud data centers participating in a federated environment to push down energy costs. They take advantage of dynamic electricity pricing to migrate the VMs to the data center with lowest energy cost and enough resources. Another work by Giacobbe et al. [85] uses the idea of migrating VMs in a federated cloud environment to reduce carbon footprint. They move the VMs from a high carbon footprint source to a data center with access to solar energy, using a two-step approach.

Our work is different from the discussed studies, since our objective is to minimize the cost associated with both energy consumption and carbon footprint. We consider carbon cost as a function of carbon intensity and carbon tax. Moreover, regarding the energy cost, we consider overhead energy of the data center along with the energy consumed by the servers. For this purpose, we exploit a data center's PUE model as a dynamic function of IT load and outside temperature. Finally, we present efficient and dynamic two-level VM placement approaches. These approaches observe the effect of different parameters on the total green and brown energy consumption, carbon footprint, and their associated cost for the cloud provider with distributed data center sites. In addition to this, the discussed VM placement approaches consider hourly changes in outside temperature, solar energy, and variable energy pricing.

4.3 System Model

In this section, we first present the system architecture, its components, and their role in a cloud computing environment. Then, we will present details on the parameters that affect cloud provider's decision in placing the VM request considering energy consumption, carbon footprint, and their associated cost. Finally, we will present the objective function and relevant constraints of the model. The list of all the symbols used in this chapter are given in Table 4.1.

The targeted system in this study is an IaaS cloud provider offering VM resources to its clients similar to Elastic Compute Cloud (EC2) service by Amazon Web Services [70]. As shown in Figure 4.1, the cloud provider consists of several geographically distributed data centers connected through a carrier network. The main parties involved in a cloud computing system are the cloud provider, cloud broker and cloud users, whose roles are discussed in the following section.

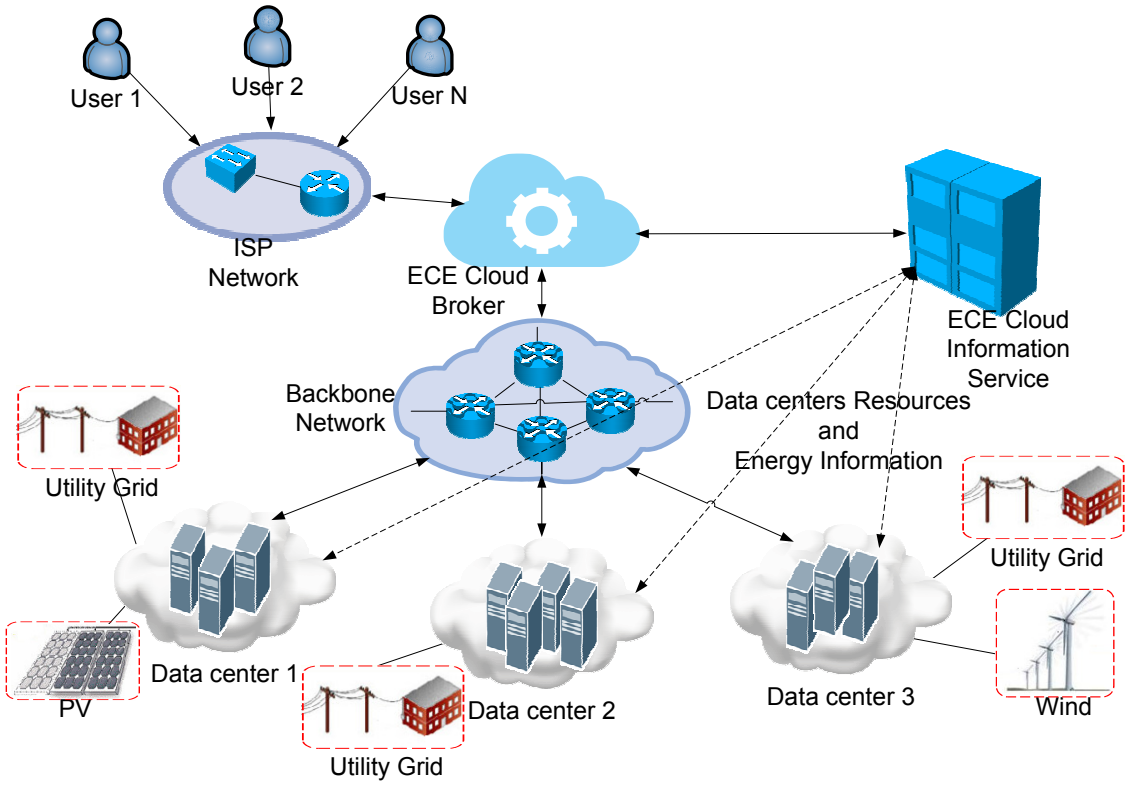


Figure 4.1: System model for geographically distributed green cloud computing environment.

4.3.1 System Components

Cloud Provider

The cloud provider consists of n data center sites, shown as a set $\mathcal{D} = \{d_1, d_2, ..., d_n\}$, distributed in different geographical locations. Each data center site, d, is connected to a backbone network to serve cloud users and uses one or more energy sources to provide electricity for its servers, networking equipment, power systems, and other devices. A data center can just use the electricity from the off-site utility grid, O, or have its own on-site or local green sources (renewable energy), G, such as wind and solar. Moreover, data centers have their local brown energy (e.g., a diesel generator), B, in case of emergencies and outages when both grid and renewable energy are not available. Data center energy sources are shown as the set $\mathcal{E} = \{G, B, O\}$. Moreover, each data center has a set of m servers, $\mathcal{S} = \{s_1, s_2, ..., s_m\}$, with different physical configurations.

Table 4.1: Description of symbols.

Symbol	Description	Symbol	Description
$\mathcal{D}$	Set of data center sites	$\mathcal{S}$	Set of servers in a data center
$\mathcal{E}$	Set of energy sources	$\mathcal{VM}$	Set of VM requests
x_{ij}	Matrix X's element to show VMs to data centers mapping	$y_v^B / y_v^G / y_v^O$	Element v of row vector $Y^B / Y^G / Y^O$, that shows VM v mapping to local brown/local green/off-site grid energy source
z_{vm}	Matrix Z's element to show VM to servers mapping	v^L	VM v holding time
C_T	Total cost of the energy and carbon	$C(v_{ij})$	Cost of running VM i at data center j
C_E	Cost of the energy	C_F	Cost of the carbon footprint
$C_s(v)$	Cost of the server energy to run the VM v	$C_o(v)$	Cost of the overhead energy to run the VM v
$C_E(v)$	Cost of the energy to run the VM v	$E_T(v)$	Total energy to run the VM v
$E_s(v)$	Server energy to run the VM v	$E_o(v)$	Overhead energy to run the VM v
$E_s^B(v) / E_s^G(v) / E_s^O(v)$	Consumed local brown/local green/off-site grid energy to run the VM v on server s	$C_E^B / C_E^G / C_E^O$	Price of the local brown/local green/off-site grid energy
$P_{s,\text{Peak}}$	Server power consumption at peak state	$P_{s,\text{Idle}}$	Server power consumption at idle state
U_{st}	Utilization of server s at time t	$P_s^{U_{st}}$	Server power at time t and utilization U_{st}
$P_s^{U_{s(t+)}}$	Server power consumption by running the new VM and the new utilization	P_o	Overhead power
U_t	Data center utilization at time t	H_t	Data center outside temperature at time t
$E_o^B(v) / E_o^G(v) / E_o^O(v)$	Consumed overhead local brown/local green/off-site grid energy to run the VM v	$C_E^B(v) / C_E^G(v) / C_E^O(v)$	Cost of the consumed local brown/local green/off-site grid energy to run VM v
$R_E^B / R_E^G / R_E^O$	Carbon footprint rate of local brown/local green/off-site grid energy source	$T_F^B / T_F^G / T_F^O$	Carbon tax of local brown/local green/off-site grid energy source
v^P	VM v required processing unit	v^M	VM v required memory
s^P	Server s total processing unit	s^M	Server s total memory

Cloud Broker

A cloud broker is the user-facing side of the cloud provider. It receives users VM requests that need to be routed to a data center site and then be placed on a server. The cloud broker should route requests to data centers in such a way that the energy consumption, carbon footprint, and their total cost for running the incoming workload are minimized. As stated in Chapter 3, the cloud broker uses the information sent from the data center sites to the Energy and Carbon-Efficient Cloud Information Service (ECE-CIS) to perform the VM placement.

Cloud Users

Cloud users submit their VM requests to the cloud broker. A submitted VM request from user i, at time t can be shown as the pair $v_i = (Type, HoldTime)$. VM type is inspired by Amazon EC2 VM instance types [70] and VM hold time depends on the application that will be run on that VM. In practice, the arrival time, type and hold time of a VM is not known by the cloud provider in advance. In our model, we serve all the VMs based on their arrival time on a first-come first-serve basis. Cloud users need to have a quality of experience (QoE) that must be satisfied by the cloud provider. The QoE for the users is defined in terms of acceptance of the submitted VM requests, which means lower rejected number of VMs higher QoE for the users.

4.3.2 System Parameters

Before discussing the system objective and constraints, we first introduce all the system parameters that affect the power consumption, carbon footprint, and their relevant cost.

Data center Power Efficiency

A data center's power efficiency depends on its PUE, which is a metric to quantify the overhead power, e.g., power supplies, cooling, lightning, and UPS, in support of the incoming IT load to the system. According to Rasmussen [37] and Goiri et al. [38], the PUE

is dependent on the data center utilization (IT load) and outside temperature. Therefore, we model PUE as $PUE = f(ITLoad, OutsideTemperature)$.

According to Rasmuseen [37], the most important parameter that affects PUE is the load of the data center and it has a linear relation with outside temperature. They showed a data center's PUE in two graphs, first by changing the IT load (at a constant temperature) and then by the change in the outside temperature (at a constant IT load). By using those two graphs, we interpolate a hyperbola relation between PUE and IT load[1] and a linear relation between PUE and outside temperature. Based on the calculations in Appendix A, we get

$$PUE(U_t, H_t) \simeq 1 + \frac{0.2 + 0.01U_t + 0.01U_tH_t}{U_t}. \tag{4.1}$$

Server Power Model

Each server is capable of hosting a different number of virtual machines depending on its configuration and VMs' sizes. Based on the scheduling policy, the incoming load to each server differs over time and this incoming load determines the power consumption of that server [86]. The relationship between the server power consumption and CPU utilization can be a constant, cubic, or even quadratic [52]. Attempts to make servers energy-efficient aim to make them energy proportional; which means that servers should only consume power in the presence of load [25]. A contemporary server's idle power, $P_{s,\text{Idle}}$, is half of the peak power, $P_{s,\text{Peak}}$. In this work, we use SpecPower benchmark [87] measurements to depict the relationship between server power consumption and server utilization. According to this data, a server's total drawn power increases linearly with the increase in utilization. This means that we let server's utilization be a direct mapping of CPU utilization, U_{st}. A server's power consumption as a function of CPU utilization is modeled as

$$P_s^{U_{st}} = P_{s,\text{Idle}} + (P_{s,\text{Peak}} - P_{s,\text{Idle}})U_{st}. \tag{4.2}$$

[1] IT load and utilization are used interchangeably.

Renewable Energy Sources

Large cloud providers use renewable energy to reduce their dependency on the electricity delivered from the grid as it is costly and less clean [88, 89]. The global amount of electricity derived from renewable sources doubled between 2000 and 2012 [90] and amongst these renewable energy, wind and solar photovoltaics (PV) are the fastest growing ones. Many cloud providers try to partially get their power from renewable energy and have their own on-site solar panels and wind turbines (e.g., Facebook [91], Apple [92], and Green House Data [93]).

Most sources of renewable energy are intermittent meaning that their availability changes uncontrollably and unpredictably over time. Cloud providers can benefit from the difference of renewable energy sources at different data center sites with different time zones at the time of VM scheduling. Several studies consider how to schedule incoming workload to manage the intermittent renewable energy. Some works use the immediate available renewable energy and cancel the running jobs when the amount of solar or wind is too low or they are not available in the system [94]. Other studies consider using prediction models for the availability of this energy to assign the workload when this energy is available and reduce the job cancellation [95]. Adding storage to the data centers, where they can store the renewable energy and use it constantly in the system, is another way to overcome the unpredictability of wind and solar [96]. However, this approach has many problems [36]. For example, 1) batteries incur energy losses due to internal resistance and self-discharge, 2) battery-related costs can dominate the cost of renewable power systems, and 3) batteries use chemicals that are harmful to the environment. Given these problems, the best way to take full advantage of the available green energy is to match the energy demand to the energy supply and maximize renewable energy utilization.

In this chapter, we consider solar energy as the local renewable energy for data center sites. We only take into account day/night differences for this energy. Moreover, we consider that renewable energy has the highest priority amongst all other energy sources and data centers get their power from these sources as long as they are available to have the highest renewable energy utilization.

Energy Price

The major incremental electricity cost of a data center is determined by the amount of energy purchased from the off-site utility grid providers. Since renewable energy has a fixed installation cost and maintenance during time, the incremental cost for using them when they are available is negligible. Moreover, the on-site brown energy (e.g., diesel generators) is only used in the absence of other energy sources. Note that we consider on-site brown energy as part of the model for the sake of comprehensiveness. However, we do not explore its effect in the evaluation part of this chapter and leave it for the interested readers to simply consider it as part of their evaluation. For the electricity driven from the grid, we consider variable energy pricing during times of the day, as having on-peak and off-peak prices. By this approach, having geographically distributed data centers for a cloud provider and variable energy pricing, gives the provider the opportunity to route requests to the data center with lowest energy price. We use C_E^O, C_E^G, and C_E^B for off-site utility grid, on-site green, and on-site brown energy prices respectively, based on cents per kilowatt-hour energy usage (cents/kWh).

Carbon Footprint Rate and Carbon Tax

Depending on the source of the power, carbon intensity could vary significantly. We represent the carbon intensity of the energy sources by R_E^O, R_E^G, and R_E^B for off-site grid, local green, and local brown, respectively based on tons per megawatt-hour used electricity (Tons/MWh). The carbon intensity for green energy (solar and wind) is zero but brown energy, from polluting energy sources, could have different rates depending on the type of the fuel burnt to generate the electricity. As green energy availability varies during the day, one data center could get the off-site grid power from more than one provider with different carbon intensities. Moreover, to reduce the effect of the emitted CO_2 and the green house gases (GHG) on the climate change [97], carbon taxes are levied. We represent carbon tax as T_F^O, T_F^B, and T_F^G for off-site utility grid, on-site brown, and on-site green energy, respectively, as dollar per ton of the emitted carbon footprint (Dollar/Ton). We should note that the carbon intensity and carbon tax for the renewable energy are zero.

4.3.3 System Objective Function and Constraints

In this section, we study the objective function of the proposed system model and its constraints.

Objective Function and Cost Modeling

The objective function is to minimize the cost of running the workload in the system, based on energy consumption and carbon footprint for the cloud provider. Meanwhile, the cloud provider should meet the cloud users' expected QoE.

The cost of running the workload is

$$C_T = \sum_{i \in \mathcal{VM}} \sum_{j \in \mathcal{D}} C(v_{ij}) x_{ij}, \tag{4.3}$$

where x_{ij} is an element of the two-dimensional matrix X and shows VM assignment to the data center site. If the element in this matrix is set to 1 means that v_i is assigned to d_j. Note that the summation is over the VM set, $\mathcal{VM} = \{v_1, v_2, ..., v_k\}$, rather than over time, since in each time epoch a data center can use more than one energy source. This means that at a certain time epoch at the data center, two running VMs could use two different energy sources. The total cost of running VMs on the servers located in geographically distributed data centers in (4.3) is composed of the cost of the energy used in the system plus the cost related to the carbon footprint in the environment due to the used electricity. We break this objective into an energy cost C_E and a carbon footprint cost C_F, as

$$C_T = C_E + C_F. \tag{4.4}$$

The energy and carbon footprint costs calculation is explained as follows.

Energy Cost: The energy cost, C_E, is the total amount of money paid to the grid electricity providers, excluding any carbon tax. In order to compute the total electricity draw in the data center sites, we need to compute the total energy used by the IT devices plus the overhead energy to run each VM. The major component of the consumed energy by the IT devices is the energy used by the servers. Therefore, we use the servers energy

consumption for each VM as the total energy used by IT devices. Based on this, we can formulate the cost for the energy consumption as

$$C_E = \sum_{v \in \mathcal{VM}} (C_s(v) + C_o(v)). \tag{4.5}$$

Depending on the type of the energy used by the server in a data center, the cost for the energy consumption by that server is different. As mentioned earlier, a server could get its energy from three different sources: local brown, local green, and off-site brown. By having three different types of energy sources, we can formulate the cost of server energy consumption as

$$C_s(v) = \sum_{\tau \in \{B,G,O\}} E_s^{\tau}(v) C_E^{\tau}. \tag{4.6}$$

The energy consumption for each VM, $E_s^{\tau}(v)$, based on the energy source is

$$\begin{aligned} E_s^B(v) &= y_v^B E_s(v) \\ E_s^G(v) &= y_v^G E_s(v) \\ E_s^O(v) &= y_v^O E_s(v). \end{aligned} \tag{4.7}$$

Here, elements y_v^B, y_v^G, and y_v^O belong to row vectors Y^B, Y^G, and Y^O, respectively. If the element y_v^{τ} of row vector Y^{τ} is set to 1, means VM v is assigned to that energy source. In order to compute the energy used by each server, we compute the increase in the power consumption due to running the new VM times its holding time. The increase in energy by using server s's $\triangle P_s$ is

$$E_s(v) = \triangle P_s v^L \tag{4.8}$$

where the increase in power consumption,

$$\triangle P_s = (P_s^{U_{s(t+)}} - P_s^{U_{st}}), \tag{4.9}$$

is based on the increase of the server utilization in the next time epoch ($t+$); that is after

the VM has entered service. Based on (4.9) and using the server power model (4.2), we have

$$\triangle P_s = (P_{s,\text{Peak}} - P_{s,\text{Idle}})(U_{s(t+)} - U_{st}). \tag{4.10}$$

The second parameter of the energy cost function in (4.5) is the cost associated with the overhead energy consumed to run the VMs, (4.11). Depending on the type of the energy used (local brown, local green or off-site brown) the energy price would be different.

$$C_o(v) = \sum_{\tau \in \{B,G,O\}} E_o^{\tau}(v) C_E^{\tau}. \tag{4.11}$$

Similar to the energy cost by the servers, we calculate the overhead energy. As stated in (4.12), we use the VM to energy sources mapping matrices to specify the energy source used for overhead to run the incoming VM.

$$\begin{aligned} E_o^B(v) &= y_v^B E_o(v) \\ E_o^G(v) &= y_v^G E_o(v) \\ E_o^O(v) &= y_v^O E_o(v). \end{aligned} \tag{4.12}$$

To compute the overhead energy usage by the VM (4.13), we use the same approach used in (4.8) for calculation of the increase in the power consumption.

$$E_o(v) = \triangle P_o v^L. \tag{4.13}$$

As noted earlier, we use PUE as a metric to compute the overhead power consumption. PUE and overhead power relation is

$$\begin{aligned} PUE(U_t, H_t) &= \frac{P_{Total}}{P_s^{U_{st}}} = \frac{P_o + P_s^{U_{st}}}{P_s^{U_{st}}} \\ P_o &= P_s^{U_{st}}(PUE(U_t, H_t) - 1). \end{aligned} \tag{4.14}$$

By using (4.14), we can rewrite (4.13) as

$$\begin{aligned} E_o(v) &= (P_s^{U_{s(t+)}} - P_s^{U_{st}})(PUE(U_t, H_t) - 1)v^L \\ &= \triangle P_s v^L (PUE(U_t, H_t) - 1). \end{aligned} \tag{4.15}$$

Carbon Footprint Cost: The second term in the objective function, (4.4), is the cost of the carbon footprint contributed to the environment due to the energy consumption. We can formulate it as the product of the cost of the consumed energy, the carbon intensity, and the carbon tax of the relevant energy source. Thus, the carbon footprint cost is defined as

$$C_F = \sum_{v \in \mathcal{VM}} \sum_{\tau \in \{B,G,O\}} C_E^\tau(v) R_E^\tau T_F^\tau. \tag{4.16}$$

By using the row vectors of energy sources to VM requests mapping, we have

$$\begin{aligned} C_E^B(v) &= y_v^B C_E(v) \\ C_E^G(v) &= y_v^G C_E(v) \\ C_E^O(v) &= y_v^O C_E(v). \end{aligned} \tag{4.17}$$

As carbon intensity and carbon tax are zero for renewable energy sources and on-site brown is just used in the absence of the other two energy sources, we can rewrite (4.16) as

$$C_F = \sum_{v \in \mathcal{VM}} C_E^O(v) R_E^O T_F^O. \tag{4.18}$$

Constraints

The objective function *minimize* $C_T = C_E + C_F$ is subject to the following constraints:

- The total allocated capacity to the VM requests running on a server should not exceed the server's capacity in terms of processing unit and memory usage:

$$\begin{aligned} \sum_{v \in \mathcal{VM}} \sum_{m \in \mathcal{S}} v^P z_{vm} \leq s^P, \\ \sum_{v \in \mathcal{VM}} \sum_{m \in \mathcal{S}} v^M z_{vm} \leq s^M, \end{aligned} \tag{4.19}$$

where, z_{vm} is an element of the two-dimensional matrix Z that is 1 if VM v is hosted on server m and 0 otherwise.

- Each running VM on a server should just use one energy source at each time epoch:

$$\forall v \in \mathcal{VM},\ y_v^B + y_v^G + y_v^O = 1. \tag{4.20}$$

- Each element of the assigned energy sources to the VMs matrices should be greater or equal to zero:

$$y_v^B, y_v^G, y_v^O \geq 0. \tag{4.21}$$

- The total amount of local green energy and local brown energy consumed by VMs should not exceed the total available green and brown energy at each data center, respectively:

$$\begin{aligned} \sum_{v \in \mathcal{VM}} (E_s^G(v) + E_o^G(v)) \leq \mathit{Total\ Available\ G}, \\ \sum_{v \in \mathcal{VM}} (E_s^B(v) + E_o^B(v)) \leq \mathit{Total\ Available\ B}. \end{aligned} \tag{4.22}$$

- The total consumed off-site grid energy should not go beyond what the cloud provider receives from the electricity provider:

$$\sum_{v \in \mathcal{VM}} (E_s^O(v) + E_o^O(v)) \leq \mathit{Total\ Assigned\ O}. \tag{4.23}$$

With the definitions in Section 4.3.3, the optimization problem becomes

$$\begin{aligned} \min_{x,y} \quad & C_T \\ \text{s.t.} \quad & (4.19) - (4.23) \end{aligned} \tag{4.24}$$

In addition to the hard constraints, we want to give local green energy the highest priority. If there is not enough green energy available, the cloud provider uses off-site

grid energy; otherwise it should use the local brown energy stored in the data center sites. That is,

$$\text{Priority } E^G > \text{Priority } E^O > \text{Priority } E^B.$$

4.4 VM Placement Approaches

In this section, we propose a dynamic VM placement algorithm to approximate (4.24) and six variations that neglect different components of the cost, to study the effect of different parameters and combinations of them on the amount of green and brown energy usage, carbon footprint, and total energy and carbon cost of the cloud data centers.

4.4.1 Cost and Renewable-Aware with Dynamic PUE (CRA-DP)

Upon the arrival of each VM request, the cloud broker has several choices with multiple data center sites and several servers within each data center, to perform VM placement. We see VM placement as a bin-packing problem with different bin sizes (e.g., physical servers) in terms of: energy price, carbon intensity, carbon tax, outside temperature, available green energy, and data center load. These differences can affect the overall energy consumption, carbon footprint, and their associated cost. Since the nature of a bin-packing problem is NP-hard, the first algorithm we propose is a derivative of the best-fit heuristic.

The CRA-DP algorithm, like that of Chapter 3, first selects the data center and then selects the server within the data center. It selects the data center with the minimum added cost for the cloud provider (minimum $\triangle C_T$), considering available green energy and dynamic PUE. CRA-DP sorts the data center sites in increasing order of the added cost due to the energy consumption and carbon footprint to run the VM for its lifetime. The server selection policy for all the algorithms in this chapter is based on the least increase in the server power consumption, given by (4.9). The pseudocode of the algorithm is presented in Algorithm 2. Note that we do not write the rest of the algorithms pseudocode, since they all are derived from CRA-DP.

Algorithm 2: Cost and Renewable-Aware with Dynamic PUE (CRA-DP) VM Placement Algorithm

Input: *datacenerList*, *hostList*
Output: *destination*

while *vmRequest* **do**
  Get data centers' Information from ECE-CIS;
  foreach *data center in data centerList* **do**
    $avgVmUtil \leftarrow v^P / avg\, s^P$;
    $\triangle E_s(v) \leftarrow v^L \times P_s^{avgVmUtil}$;
    $\triangle E_o(v) \leftarrow \triangle E_s(v) \times PUE(U_{t+}, H_t)$;
    $\triangle E_T(v) \leftarrow \triangle E_s(v) + \triangle E_o(v)$;
    $availGreen \leftarrow$ Get Current *availablegreenenergy*;
    if $availGreen > 0$ **then**
      if $availGreen <= \triangle E_T(v)$ **then**
        $usedGreen \leftarrow availGreen$;
        $availGreen \leftarrow 0$;
      else
        $usedGreen \leftarrow \triangle E_T(v)$;
        $availGreen \leftarrow availGreen - usedGreen$;
    $usedOffSiteEnergy \leftarrow \triangle E_T(v) - usedGreen$;
    $\triangle C_E \leftarrow usedOffsiteEnergy \times C_E^O$;
    $\triangle C_F \leftarrow usedOffsiteEnergy \times R_E^O \times T_F^O$;
    $\triangle C_T \leftarrow \triangle C_E + \triangle C_F$;
    Add *dataCenter* with $\triangle C_T$ into *aggregateDCList*;
  Sort *aggregateDCList* in an ascending order of $\triangle C_T$;
  foreach *dataCenter in aggregateDCList* **do**
    foreach *host in hostList* **do**
      $P_s^{U_{st}} \leftarrow$ Get current *hostDynamicPower*;
      $P_s^{U_{s(t+)}} \leftarrow$ Calculate *hostDynamicPower* after initiating the *vm*;
      $\triangle P_s \leftarrow P_s^{U_{s(t+)}} - P_s^{U_{st}}$;
    Sort *hostList* in an ascending order of $\triangle P_s$;
    foreach *host in hostList* **do**
      if *host is suitable for vm* **then**
        $destination \leftarrow (data\ center, host)$;
        return *destination*;
  $destination \leftarrow null$; //rejection of request;
  return *destination*;

4.4.2 Cost-Aware with Dynamic PUE (CA-DP)

The CA-DP algorithm differs from CRA-DP in that CA-DP does not consider the availability of renewable energy while calculating the $\triangle C_T$ to select the data center site. Note that all the algorithms assume that if a data center site has renewable energy available, all the servers and racks are always powered by green energy, unless there is not enough renewable energy in the system. In this case, they will get their required power from off-site grid energy sources.The pseudocode of this algorithm is the same as the CRA-DP, but it omits Lines 8-15 and at Line 16, the *usedGreen* is set to 0.

4.4.3 Energy and Renewable-Aware with Dynamic PUE (ERA-DP)

The ERA-DP algorithm makes decision based on the increase in the total energy consumption (server energy + overhead energy). It calculates the total energy added to each data center to run the new VM ($\triangle E_T(v) = \triangle E_s(v) + \triangle E_o(v)$) with considering dynamic PUE and amount of the available renewable energy. This algorithm omits Lines 17-19 of the Algorithm 2 and Lines 20 and 21 are based on the $usedOffsiteEnergy$ instead of $\triangle C_T$. The rest of the algorithm is the same as the CRA-DP algorithm.

4.4.4 Energy-Aware with Dynamic PUE (EA-DP)

The EA-DP algorithm is similar to ERA-DP, except after calculating $\triangle E_T(v) = \triangle E_s(v) + \triangle E_o(v)$ for each data center site, it does not consider the availability of renewable energy (*usedGreen* is set to 0).

4.4.5 Energy-Aware with Constant PUE (EA-CP)

This algorithm is a derivation of the EA-DP, except that PUE value does not vary by the change in IT load and outside temperature and it has a constant value. In order to obtain a reasonable constant value for PUE, we calculate its average while performing the CRA-DP algorithm from a low load until data centers get fully utilized. Note that, as considering static value for PUE just multiplies the servers energy consumption by a

constant value $\triangle E_T(v) = \triangle E_s(v)(1 + PUE)$, the results are expected to be the same as when the VM placement is without considering the overhead power and just based on the servers power consumption.

4.4.6 Carbon Footprint-Aware with Dynamic PUE (FA-DP)

This algorithm is a derivation of the ECE algorithm in Chapter 3, which considers the effect of PUE and carbon intensity while here PUE has a dynamic value. It selects the data center with the minimum value of $R_E^\tau \times PUE(U_{t+}, H_t)$ and $\tau \in \{B, G, O\}$.

4.4.7 Energy Price-Aware (EPA)

The energy price-aware (EPA) VM placement algorithm, upon the arrival of each VM request selects the data center site with the cheapest energy price (minimum C_E^τ and $\tau \in \{B, G, O\}$). Since green energy cost is zero, the data center site with the available green energy has the highest priority.

4.5 Performance Evaluation

We evaluate the performance of the proposed approaches to investigate the effect of different parameters on the total cost, brown and green energy consumption, and carbon footprint. Note that all algorithms are evaluated based on the total cost C_T described in Section 4.3.3, even though some algorithms ignore some components of the cost.

4.5.1 Experiment Setup

In order to evaluate the proposed approaches, we target an IaaS cloud computing environment. Since it is difficult to perform large-scale and repeatable evaluation on real infrastructures, we use simulation to conduct our experiments. The CloudSim toolkit [72] is a simulation platform that allows evaluation of virtualized cloud environments. As the core framework of CloudSim does not support energy and carbon-efficient simulation, we use the extended version developed in Chapter 3 that enables these features.

Table 4.2: Data center site characteristics.

Site Characteristics	Dallas	Richmond	San Jose	Portland
Server Power Model	$P_s^{U_t} = 120 + 154U_{st}$			
PUE Model	$PUE(U_t, H_t) = 1 + \frac{0.2 + 0.01U_t + 0.01U_tH_t}{U_t}$			
Carbon Intensity (Tons/MWh)	0.730	0.69	0.35	0.147
Carbon Tax (Dollars/Ton)	24	22	11	48
Energy Price (cents/kWh)	6.1	6.54	10	5.77

Apart from adding the energy and carbon-awareness to the CloudSim core, we add other features such as costs of the consumed energy and emitted carbon, access to renewable energy (solar energy in this chapter), overhead power consumption, and dynamic PUE.

Data centers Configuration

We consider four data center sites located in four cities chosen from different states in the United States at three different time zones. These cities are chosen from the Data centers Map website [98] and they are: Dallas in Texas, Richmond in Virginia, San Jose in California, and Portland in Oregon. Since they are connected to one network backbone, the number of hops a packet traverses from source to destination is between 12 and 14 hops [71]. Therefore, different network distances do not affect site selection. Each data center has 130 heterogeneous physical servers with five different configurations described by four parameters: (Number of Cores, Core Speed (GHz), Memory (GB), Storage (GB)). The five different server types are: Type1 (2, 1.7, 16, 2000), Type2 (4, 1.7, 32, 6000), Type3 (8, 1.7, 32, 7000), Type4 (8, 2.4, 64, 7000), and Type5 (8, 2.4, 128, 9000).

Servers Power Consumption

As discussed in Section 4.3.2, we use the approximate linear relation with the server utilization, as shown in the work by Pellet et al. [52], for the server power model. The power model, stated in Table 4.2, is the linear approximation against SpecPower results for two Dell PowerEdge servers.

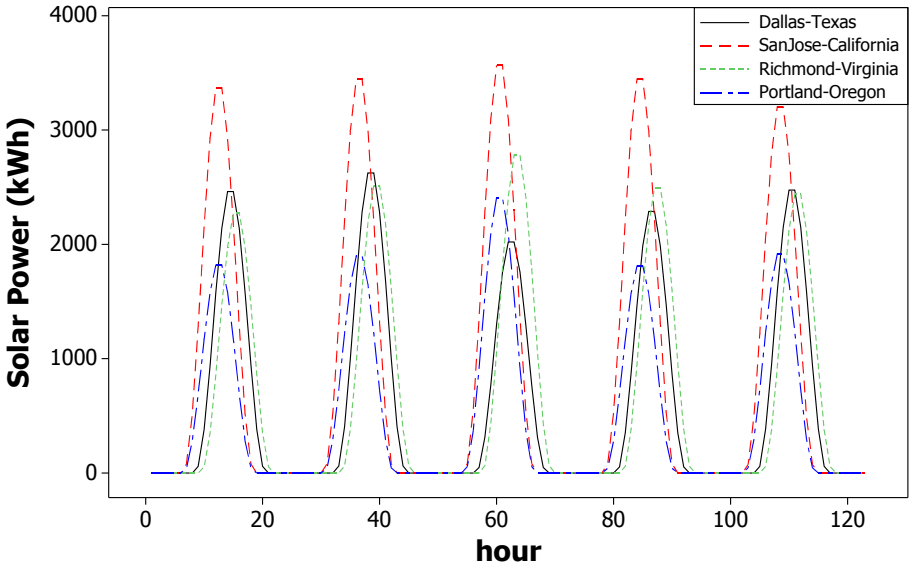


Figure 4.2: Solar Energy for 5 Days.

PUE Model

We use the PUE model described in Section 4.3.2 for all the data centers. We assume that the efficiencies of all the data center sites' infrastructure is the same. The PUE model is shown in Table 4.2.

Solar Energy

We use the data reported in the project undertaken by the NREL [99] to get the solar energy availability in the four aforementioned cities. We use the data of a primary station, solar radiation for flat-plate collectors facing south at a fixed tilt in ($kWh/m^2/day$). We consider five days form May 26th, 2014 to May 30th, 2014 for the simulation time and set the total area for the solar irradiation absorber flat-plates $2684m^2$ from the configuration by Solarbayer [100]. With this information, we can get the daily solar energy in terms of kWh/day. To get the hourly solar traces, we assume that the solar energy for times before 6 a.m. and after 6 p.m. is 0. Moreover, the distribution of the energy between 6 a.m. and 6 p.m. has a raised cosine distribution, with the peak at 12 noon. Knowing the total solar of one day and integrating the raised cosine between 6 a.m. and 6 p.m., we calculate hourly available solar energy in kWh for these four cities as shown in Figure 4.2.

Table 4.3: VM types and simulated user requests; (Bag-of-Task (BT) and Web-Request (WR)).

VM Type		**Number of Cores**	**Core Speed** (GHz)	**Memory** (MB)	**Storage** (GB)	**Probability and UserRequest**
Standard Instances	M1Small	1	1	1740	160	0.25-BT
	M1Large	2	4	7680	850	0.12-WR 0.25-BT
	M1XLarge	4	8	15360	1690	0.08-WR
High Memory Instances	M2XLarge	2	6.5	17510	420	0.12-WR
	M22XLarge	4	13	35020	850	0.08-WR
High CPU Instances	C1Medium	2	5	1740	320	0.1-BT

Carbon Footprint Rate and Carbon Tax

The data centers' carbon intensity ($Tons/MWh$) is obtained from the US Department of Energy, Appendix F, Electricity Emission Factors [16]. We use the data reported by the Carbon Tax Center [101] for the carbon tax, due to the contribution in emitting carbon in the environment, in terms of dollars per ton of CO_2 ($Dollars/Ton$). Values for carbon intensity and carbon tax for the chosen data center sites are reported in Table 4.2.

Energy Price

We consider on-peak and off-peak pricing model for the electricity driven from off-site electricity providers. Energy prices are taken from the US Energy Information Administration [17]. Peak energy price for 4 sites are shown in Table 4.2. Times of the day before 8 a.m. and after 10 p.m. are off-peak times and the energy price will be half of the on-peak times (8 a.m. to 10 p.m.). We assume the on-site solar energy has zero incremental cost, since it has a one time capital cost and regular maintenance independent of use.

Outside Temperature

We derive the hourly temperature of the four data center sites from May 26th, to May 30th 2014 from the Weatherbase portal [102]. Figure 4.3 shows the hourly temperature for the aforementioned sites.

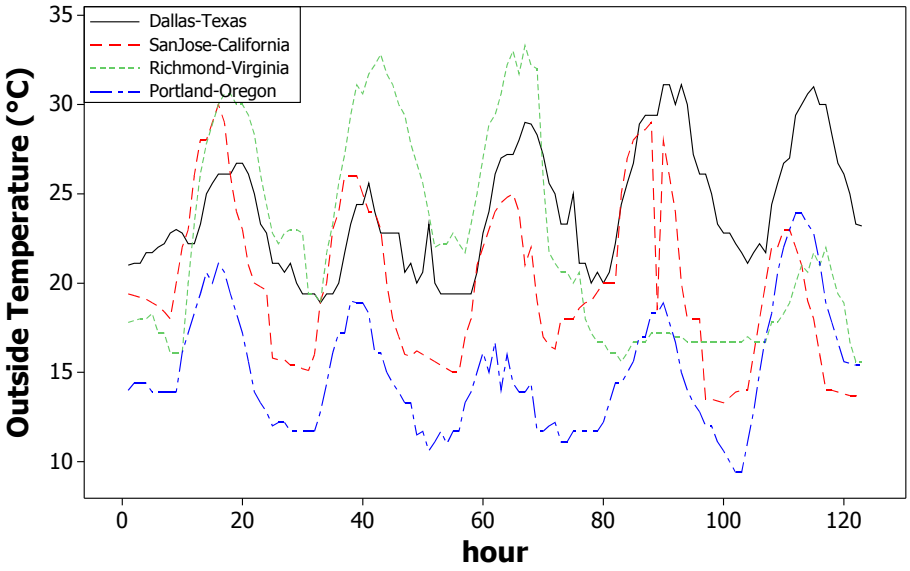


Figure 4.3: Outside Temperature for 5 Days.

Workload Data

The incoming workload to the system is the VM requests from cloud users. Since we only deal with the placement of the VM requests and allocation of their required resources, we do not need to know the type of application running within the instantiated VM. However, we assume that each VM operates at its maximum utilization and uses all the allocated resource. Each VM request has physical characteristics, that are inspired by Amazon EC2 VM instance types. Beside the physical requirements, each VM has a submission time from the user and holding time. For the system workload, we use the same model and workload generator we used in Chapter 3. We generate two types of VMs known as bag-of-tasks and web-requests with the same arrival rate and different holding time pattern (longer holding time for web-requests). The applied workload generator for this purpose is the Lublin-Feitelson [13] workload model. To generate bag-of-tasks, we use the parameters from [13], except that we change the first parameter of the Gamma distribution to 20.4 to get VMs with longer holding times, and we change the holding-time distribution to Hyper Gamma, with mean 73 and variance 165, to generate web-requests. The VM types and the probability of each type submitted from the users are stated in Table 4.3.

We ran the simulation for 5 days (120 hours) and in order to have a steady environment, we omitted 5% of the generated requests from the start and 5% from the end as they are part of the warm-up and cool-down of the system, respectively. (The latter is necessary as the CloudSim simulation finishes when the last VM completes.) Note that we consider each request generated by Lublin as a VM request. Finally, since Lublin takes a random number as input, we repeated each experiment 30 times, and report the mean of the results.

4.5.2 Experiment Results

In the experiments, we measure the total amounts of green and brown energy consumption, carbon footprint, and their associated cost. Moreover, we check the total cost of the cloud computing system under different VM placement policies. Finally, we measure the number of rejected VMs in the system due to insufficient physical resources that leads to the violation of users' QoE in terms of SLA violation. The load varied from 500 VMs, to show how the system behaves when one data center has the physical capacity to host all the requests, up to 1700 VMs, when the system performs at its full utilization and rejects some of the incoming load.

Note that in the experiments, we checked that the results are not skewed and based on this we report their general behavior on the mean value. Moreover, we performed 2-sample t-test to check whether the differences in results are significant or not.

Green Energy Consumption

In this experiment, we measure the amount of green energy consumed by different VM placement policies to run the incoming workload in the system. As Figure 4.4 demonstrates, three algorithms (ERA-DP, CRA-DP, EPA) that consider availability of renewable energy in the placement, have the most green energy consumption, with a slightly higher usage for the ERA-DP algorithm. The EA-CP algorithm has the smallest green energy consumption, as it is not renewable-aware and uses a constant value for PUE. The latter factor leads to not considering data centers' load change and their outside temperature;

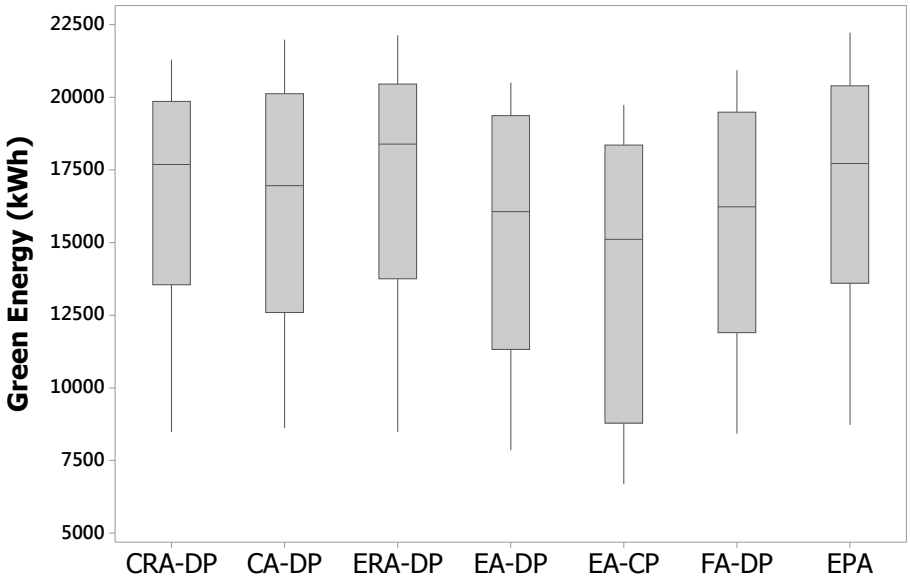


Figure 4.4: Green energy consumption.

therefore it does not lead to an efficient site selection and distribution of load among data centers to get the most of available solar energy at different times of the day. In order to study the effect of considering dynamic PUE versus constant PUE and renewable energy, we run a 2-sample t-test on ERA-DP and EA-CP. We get $p = 0.04$, therefore we conclude that considering dynamic PUE and renewable energy have significant effect on the total green energy consumption. The algorithms (CA-DP, EA-DP, and FA-DP) that are not renewable-aware consume less green energy as well. But the difference with the group of renewable-aware algorithms is not significant ($p > 0.05$), since, as noted earlier, green energy has the highest priority if the data center has access to it.

Brown Energy Consumption

Figure 4.5 shows the amount of brown energy consumption by different VM placement policies. At lower loads, EA-CP consumes significantly more brown energy than the other algorithms, as it is based on a constant value for PUE and distributes the load without considering current load of the data center sites and the outside temperature. The rest of the policies have close behavior. The reason is that they all are based on dynamic PUE, the system load is low and the renewable energy source has the highest

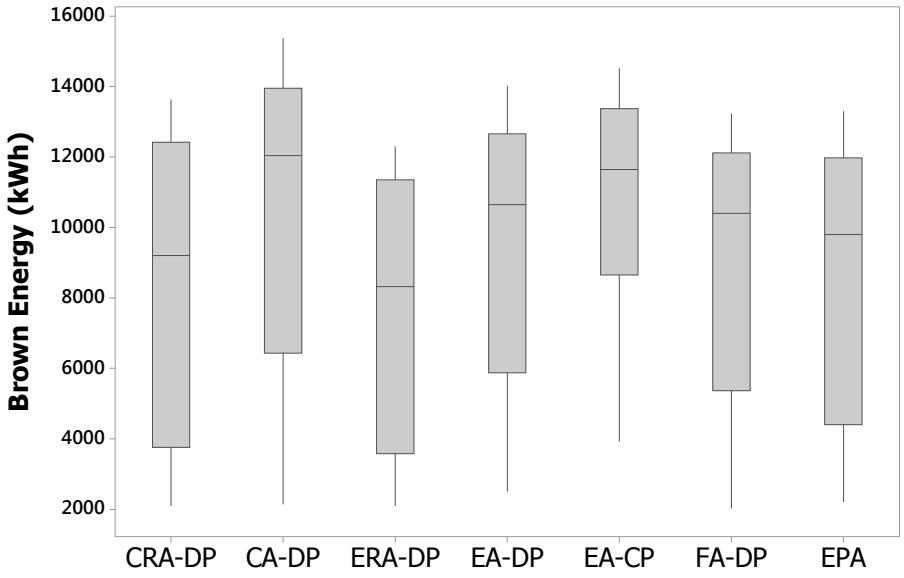


Figure 4.5: Brown energy consumption.

priority. As the system load increases, EA-CP continues consuming more brown energy, just with a slight improvement; since the constant PUE value, that is the average value of PUE gets closer to the real dynamic value. From the results we observe that CA-DP has a sudden increase in the brown energy consumption. Because its placement is based on the increase in the total cost in the system and parameters, such as dynamic energy pricing, that affect the decision making do not have any impact on reducing the total brown consumption.

Overall, ERA-DP policy has the lowest brown energy consumption. It consumes, on average, 8.9% less brown energy in comparison to its competitor, CRA-DP algorithm. These two algorithms along with EPA, that is also based on considering renewable availability, have the lowest brown energy consumption. Moreover, ERA-DP, consumes 31.3% less brown energy on-average than EA-CP and 36.4% less than CA-DP. Based on the 2-sample t-test on ERA-DP and EA-CP, there is significant difference ($p = 0.01$) in the amount of consumed brown energy. Moreover, t-test on ERA-DP and CA-DP shows the significance ($p < 0.05$) of considering increase in the energy consumption rather than increase in the total cost while VM placement is carried out.

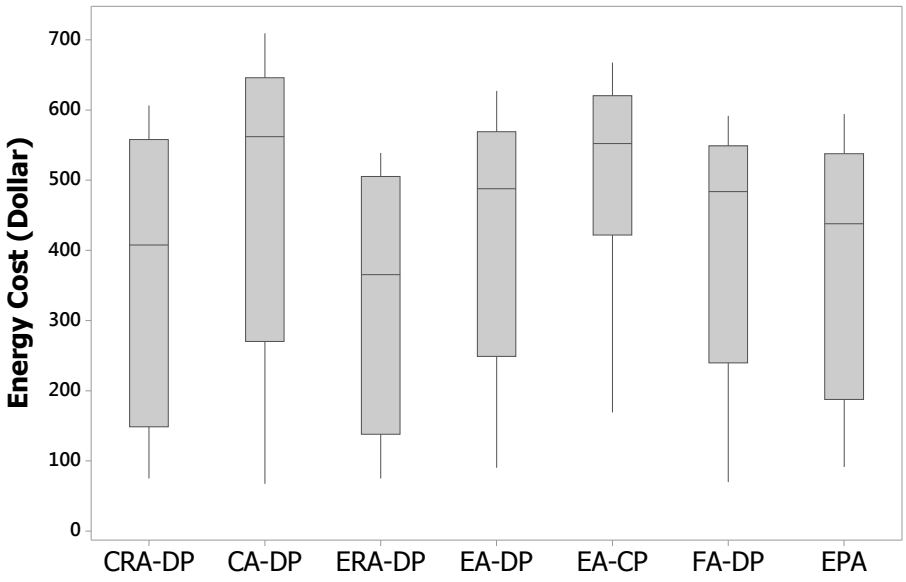


Figure 4.6: Energy cost.

Energy Cost

Energy cost is a function of the amount of brown energy consumption, since the cost of renewable energy is considered zero in this chapter. We observe the same behavior among the algorithms in Figure 4.6 as we witnessed in Figure 4.5. Algorithm ERA-DP reduces the energy cost by an average of 10.03% compared to its competitor algorithm, CRA-DP. Moreover, t-test results show that the energy cost difference between ERA-DP and two other algorithm, EA-CP and CA-DP, is significant with $p = 0.002$ and $p = 0.042$, respectively. This emphasizes the importance of considering dynamic PUE, renewable energy, and increase in energy consumption.

Carbon Footprint

Carbon footprint in the system, likewise energy cost, is a result of the usage of the brown energy sources. Hence, we should expect a similar pattern as Figure 4.5. But we should not expect the same gap from one policy to another, since different energy sources have different carbon intensities. One significant difference in Figure 4.7 is that, at lower workload (VM<800), FA-DP performs significantly better than ERA-DP. The reason is that FA-DP considers sources carbon intensity and dynamic PUE at the same

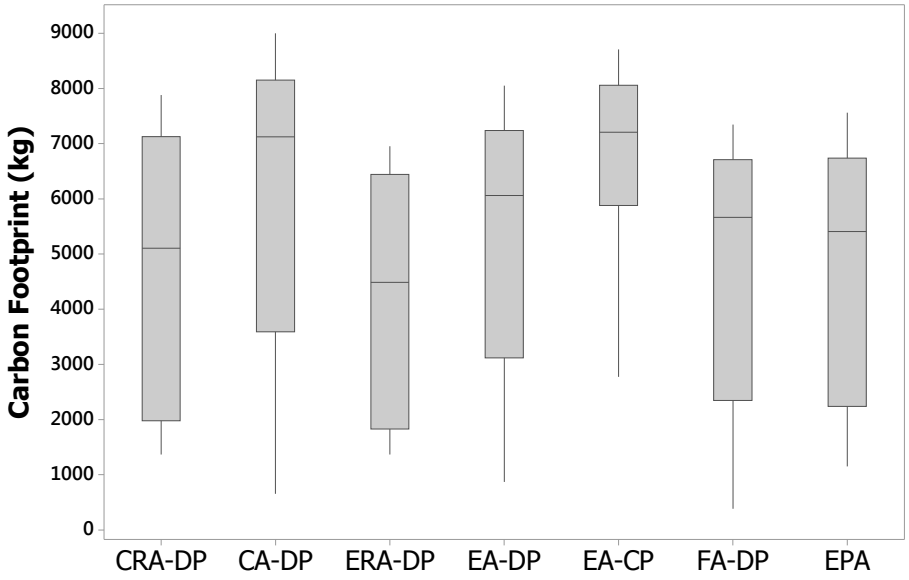


Figure 4.7: Carbon footprint.

time and at lower loads it submits the requests to the data center with the minimum $carbon\ footprint \times PUE$. Though by the increase in the incoming load and the need to use more than one site, this policy does not perform optimal and ERA-DP is the algorithm that has a better performance. Overall, ERA-DP comparing to its close competitor, FA-DP, reduces carbon footprint 10.6% on average. In addition, it reduces carbon footprint on an average of 60% and 42% in comparison to EA-CP and CA-DP, respectively. T-test shows $p < 0.01$ and $p = 0.044$ for ERA-DP versus EA-CP and CA-DP, respectively, which again assures the importance of considering dynamic PUE, renewable energy, and changes in energy consumption.

Carbon Cost

Figure 4.8 shows the cost of the carbon footprint in dollars. Since any increase in the value of a carbon tax is the result of carbon footprint growth, the behavior of different algorithms and their gaps would be the same as the total carbon footprint in Figure 4.7. Still ERA-DP on average has 7.4% less carbon cost comparing to FA-DP. Moreover, it has on average 68% and 45.6% better performance comparing to EA-CP ($p < 0.01$) and CA-DP ($p = 0.33$), respectively.

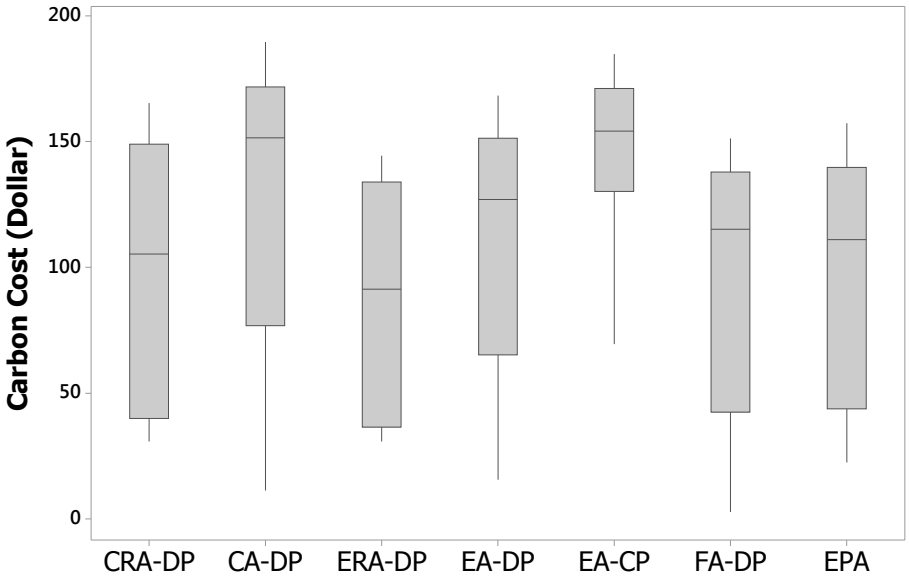


Figure 4.8: Carbon cost.

Total Cost

Figure 4.9 demonstrates an overall view of the effect of different VM placement policies on the total cost related to the energy and carbon footprint. At lower system loads, carbon cost (FA-DP) has a slight effect on the total cost of the system; whilst with the increase in the load, ERA-DP improves the total cost by an average of 19.3% and 10.5% comparing to FA-DP and CRA-DP, respectively. Moreover, ERA-DP significantly improves the total cost by an average of 57.3% and 43.8% in comparison to EA-CP ($p = 0.001$) and CA-DP ($p = 0.04$), respectively.

SLA Violation

The last experiment measures SLA violation rate in order to make sure users' quality of experience is satisfied. SLA is calculated as the number of rejected VMs due to insufficient physical resources in the system. Table 4.4 shows SLA violation rate under increasing workload for different VM placement policies. The table reports violations for loads from 1300, since below this load the violation rate for all the policies is zero and all the incoming load to the system are served. From the table, we observe that all the

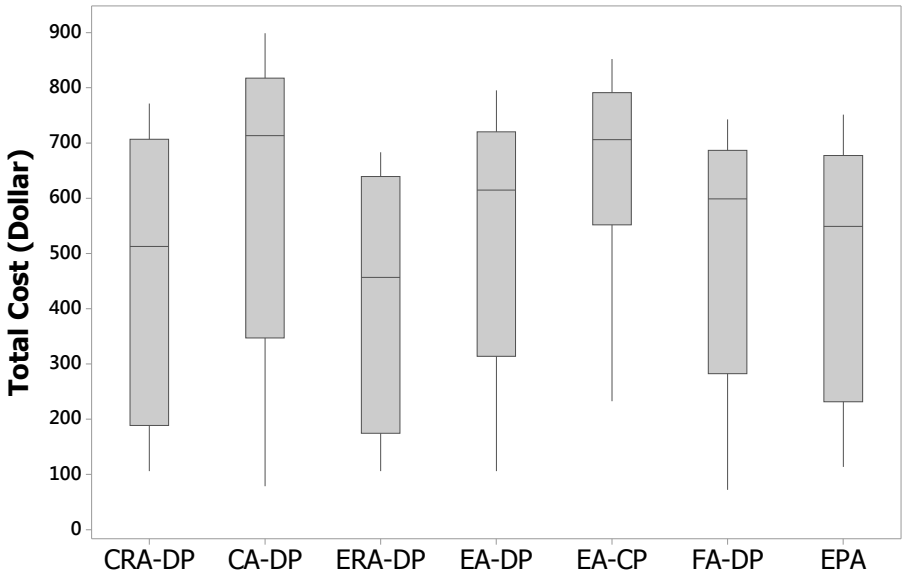


Figure 4.9: Total cost.

placement policies have close SLA violation. Moreover, ERA-DP at two points has the minimum violation rate and in the rest it only has 0.1-0.2% higher violation comparing the minimum reported ones. As a result, we can conclude that ERA-DP performs better in terms of brown energy consumption, carbon footprint, energy and carbon cost. Moreover, it has close, even at some points minimum, values for SLA violation comparing to the competitive algorithms.

Table 4.4: SLA violation for VM placement policies.

Algorithm	**SLA Violation Under Different VM Requests**				
	1300	**1400**	**1500**	**1600**	**1700**
CRA-DP	0.08%	0.3%	**0.9**%	**2.7**%	**4.9**%
CA-DP	**0.0**%	0.3%	1.1%	2.9%	5.2%
ERA-DP	**0.0**%	**0.2**%	1.0%	2.6%	5.1%
EA-DP	0.06%	0.3%	1.1%	2.9%	5.2%
EA-CP	0.05%	0.3%	1.3%	3.1%	5.5%
FA-DP	0.05%	0.3%	1.3%	3.1%	5.5%
EPA	0.14%	0.8%	2.4%	4.5%	6.3%

4.6 Summary

This chapter investigates different parameters that affect energy and carbon cost for a cloud provider with geographically distributed data center sites. First, we consider carbon cost as part of the total cost that enables the provider not only decrease the total cost, but also reduce the CO_2 emission. Moreover, to decrease the energy cost, we consider overhead energy consumption in support of IT devices in the data center. We employ PUE as a metric that affects overhead energy of a data center, which is responsible for almost half of the energy consumption. We exploit a model for PUE as a function of data center's IT load and outside temperature. Further, we consider access to renewable energy sources, besides off-site grid (known as brown) sources.

We have presented and evaluated different energy and carbon-aware dynamic VM placement approaches. In a nutshell, ERA-DP that considers dynamic PUE, availability of renewables, and changes in energy consumption has the highest effect in reducing the total cost of energy and carbon and also reducing brown energy usage; whilst has the same level of SLA compared to the other algorithms. Furthermore, amongst the renewable-aware algorithms (CRA-DP and ERA-DP) and EPA, the later algorithm performs worse. Because EPA prefers the sites with available renewable energy, as it has the lowest price (zero), thus distributes the load between data center sites to get the most of renewables. This leads to use of computing resources of all the data centers and having overhead power as a major killer for the power consumption in all the sites.

In the next chapter, we explore how much energy cost savings can be made knowing the future level of renewable energy in the data centers. We study the effect of VM migration between data center sites to utilize the most of the available renewable energy. We provide competitive-ratio bound of two online algorithms, one with no and one with partial knowledge about the future level of renewable energy, comparing to the optimal off-line with full knowledge of the future level of renewable energy.

Chapter 5

Online Virtual Machine Migration for Renewable Energy Usage Maximization

Energy consumption and its associated costs represent a huge part of cloud providers' operational costs. In this chapter, we explore how much energy cost savings can be made knowing the future level of renewable energy (solar/wind) available in data centers. Since renewable energy sources have intermittent nature, we take advantage of migrating Virtual Machines (VMs) to the nearby data centers with excess renewable energy. In particular, we first devise an optimal offline algorithm with full future knowledge of renewable level in the system. Since in practice, accessing long-term and exact future knowledge of renewable energy level is not feasible, we propose two online deterministic algorithms, one with no future knowledge called deterministic and one with limited knowledge of the future renewable availability called future-aware. We show that the deterministic and future-aware algorithms are $1 + 1/s$ *and* $1 + 1/s - \omega/s.T_m$ *competitive in comparison to the optimal offline algorithm, respectively, where* s *is the network to the brown energy cost,* ω *is the look-ahead window-size, and* T_m *is the migration time. The effectiveness of the proposed algorithms is analyzed through extensive simulation studies using real-world traces of meteorological data and Google cluster workload.*

5.1 Introduction

Data centers as the heart of a cloud computing system are energy intensive. This is due to the high power required to run the IT equipment, power, and cooling infrastructure [63].

This chapter is derived from the publication: Atefeh Khosravi, Adel Nadjaran Toosi, and Rajkumar Buyya, *"Online Virtual Machine Migration for Renewable Energy Usage Maximization in Geographically Distributed Cloud Data Centers"*, Concurrency and Computation: Practice and Experience (CCPE), Wiley Press, New York, USA, DOI:10.1002/cpe.4125, 2017.

Based on the report by Koomey [18], data centers were responsible for 1% of the world's total energy consumption in the year 2005, equivalent to 152 billion kilowatt-hours (kWh) that has been almost doubled from the year 2000. Besides the high energy consumption of data centers, the cost associated with the energy is a big concern as well. According to Hamilton [103], the energy costs are estimated to be around 42% of the data center's operational costs. Furthermore, the issue of high energy consumption by data centers makes them responsible for 2% of the world's total CO_2 emission [104].

To overcome the problem of *high energy consumption* that leads to *high energy costs* for the cloud provider and *environmental concerns* due to the high CO_2 emission of energy sources, there are two possible solutions: 1) improving the data center's efficiency or 2) replacing the brown energy sources with clean energy sources. By making data centers energy efficient and aware of energy sources, cloud providers are able to reduce their costs significantly [12]. Recently, large IT companies started to build their own on-site renewable energy sources, such as Facebook's solar-powered data center in Oregon [105], its newly build wind-powered data center in Texas [106], Amazon [107], Apple [108], Google [109], and Microsoft [110] renewable energy farms. To this end, we consider access to on-site renewable energy sources[1], which is becoming popular for modern data center sites. However, due to the intermittent nature of renewable energy sources, these data centers consider access to off-site electrical grid (also known as brown energy) to power their infrastructure in the absence of renewables. The on-site energy sources considered are solar and wind, the two fastest growing renewables. As discussed earlier, these energy sources are not available all the time. Solar energy is only available during the day and it has its peak during noon, while wind energy fluctuates during the day and does not follow any particular pattern. Yet, cloud providers try to minimize their energy cost through maximizing on-site renewable energy usage. However, maximizing renewable energy usage in one data center site is challenging, because of the intermittent and limited nature of solar and wind energy. One solution to achieve this goal is to migrate the load (VMs) from one data center without currently available renewable energies to a data center with excess renewable energy. Moreover, migrating VMs requires

[1]Renewable and green energy sources are used interchangeably.

the knowledge of the time that migration should take place to avoid brown energy usage.

In this chapter, we are motivated by the following question: "with limited or no priori knowledge of the future level of renewable energies, when should VM migration take place so that the energy cost is minimized and accordingly the overall renewable energy consumption is maximized?" For this, we study cost-minimizing VM migration algorithms targeting a cloud provider with distributed data center sites within a region[2] with access to disparate renewable energy sources. We model the cost minimizing VM migration problem, and determine the cost of offline algorithm, as well as the competitive ratio for the optimal online deterministic algorithm. Moreover, we enhance the online algorithm by adding limited future knowledge of available renewable energy in the system. We evaluate the proposed algorithms through extensive simulation using CloudSim toolkit [72], traces of wind and solar energy undertaken by the National Renewable Energy Laboratory (NREL) [15], and real-world workload traces from Google [14].

The **main contributions** of this chapter are:

1. Formulation of the offline cost optimization problem for VM migration, across geographically distributed cloud data centers, with respect to the availability of renewable energy.

2. Proof and competitive ratio analysis of the optimal online deterministic algorithm with no future knowledge against the optimal offline algorithm.

3. Design of an online VM migration solution with limited future knowledge regarding the solar/wind power availability.

4. Evaluation of the proposed algorithms through extensive simulations using real-world renewable energy (solar and wind) traces and workload traces of a Google cluster.

The remainder of the chapter is organized as follows. The next section discusses the related work. The system model and cost optimization problem are formalized in Sec-

[2]A region is a separate geographic area with multiple and isolated locations known as *availability zones* connected through dedicated low latency links. This is the same definition used by Amazon EC2 architecture [70].

tion 5.3. Section 5.4 presents the optimal offline solution followed, in Section 5.5, by introducing the online deterministic and future-aware online algorithms. Evaluation results are presented in Section 5.6 and a summary is outlined in Section 5.7.

5.2 Related Work

The context of energy-efficient resource management has gained considerable attention over the last few years. Moreover, along with the objective of energy consumption optimization, the problem of reducing carbon footprint has been an ongoing research due to environmental concerns, rise in global warming, social and governmental pressure, (impose of carbon tax), and more importantly increase in the usage of renewable energy sources to power data center sites by cloud providers [93]. Most of the early works on energy efficiency focus on a single server and intra-data center optimization techniques [21, 22, 24, 82]. An extensive taxonomy and survey by Beloglazov et al. [23] discusses different techniques on energy-efficient data centers. Similar to our work, Beloglazov and Buyya [111] formulated cost for the single VM migration and dynamic VM consolidation problems within a single data center environment. They conducted competitive-ratio analysis to characterize the performance of optimal online algorithms against the optimal offline competitor. On the contrary, we focus on energy cost minimization by applying VM migration between data center sites considering access to renewable energy sources and limiting brown energy usage. While they consider service level agreement violation cost due to server over-subscription, we consider inter-data center network cost and additional brown energy usage.

Following the high energy consumption by data centers, increase in their operational costs, and the issue of carbon footprint encouraged cloud providers to have their own on-site renewable energy sources and power their data centers completely or partially through clean energy sources [92, 105]. Kong and Liu [112] investigated research works towards green-energy-aware power management for single and multi data centers. Recently, there has been a large body of literature considering reducing energy costs targeting inter-data center sites. They achieve this goal either by considering spatial (different

electricity prices in different geographical locations) or temporal changes (different electricity prices during different times of the day) of the electricity derived from off-site grid or by maximizing renewable energy usage, which leads to minimizing brown energy consumption as well.

One of the earliest studies that targets reducing the costs associated with brown energy consumption is done by Le et al. [42]. They consider the amount of load each data center can accommodate based on its electricity price and energy source, whether it is brown or green energy and within a specific time period and budget. A similar work by Liu et al. [44] considers geographical load balancing to minimize brown energy consumption through an optimal mix of renewable energy sources (solar and wind) as well as storage of these renewables in data centers. An extension to that work has been done by Lin et al. [45] to explore the optimal combination of brown and green (solar/wind) energy sources aiming a net-zero brown energy system. To tackle the same problem, Toosi and Buyya [113] proposed a fuzzy logic-based load balancing algorithm that needs no knowledge of future. All these works consider routing of incoming load to the data centers based on their initial renewable/brown state by the time of users' requests submission. Whilst, we consider VM migration between data center sites, due to the limited and intermittent nature of renewable energy sources.

Rao et al. [114] aimed at minimizing total cost by considering electricity pricing data to route delay-constraint applications. Ren et al. [54] proposed online algorithms to route jobs to the data centers with low electricity prices or suspend jobs and resume them later, if necessary. Buchbinder et al. [60] has the same objective of reducing energy cost for a cloud provider. They take advantage of dynamic electricity pricing to migrate running batch jobs to the data center with lower electricity price. Comparatively, we focus on VM migration and taking advantage of available renewable energy sources in data centers.

Towards reducing energy cost and limiting brown energy consumption, Chen et al. [49] proposed scheduling algorithms to forward incoming jobs to the data centers considering energy source at the data center and requests' deadline to process the incoming requests for further execution. Celesti et al. [50] proposed a framework to allocate VM requests to the data centers with the highest level of solar energy and lowest cost. Le et al. [55] used

Table 5.1: Comparison of proposed work with existing literature.

	Energy cost minimization	**Brown energy minimization**	**Renewable energy maximization**	**Migration**	**Competitive-ratio analysis**
Le et al. [42]	✓	✓	✓		
Liu et al. [44]		✓	✓		
Lin et al. [45]	✓	✓	✓		✓
Rao et al. [114]	✓				
Le et al. [115]	✓	✓	✓		
Ren et al. [54]	✓				✓
Buchbinder et al. [60]	✓			✓	✓
Chen et al. [49]		✓	✓		
Celesti et al. [50]	✓	✓	✓		
Le et al. [55]	✓	✓	✓		
Luo et al. [61]	✓				
Toosi and Buyya [113]	✓	✓	✓		
Our work	✓	✓	✓	✓	✓

the same idea of assigning incoming requests to the data center considering green energy sources and electricity price in order to minimize brown energy consumption. Luo et al. [61] leverage both the spatial and temporal variation of electricity price to route the incoming requests between geographically distributed data centers targeting energy cost minimization.

The comparison of the existing literature with our proposed work is summarized in Table 5.1. Our work is different from the discussed studies, since we consider VM migration between data centers to maximize renewable energy (solar/wind) consumption. The targeted system here has several data centers located within a region (geographically near locations). We analyze the VM migration problem by calculating the optimal offline cost and computing the competitive ratio for an online deterministic algorithm, without any future knowledge of renewable energy level, and a future-aware online algorithm with a look-ahead window and limited knowledge, up to a window-size, of future level of solar and wind energy.

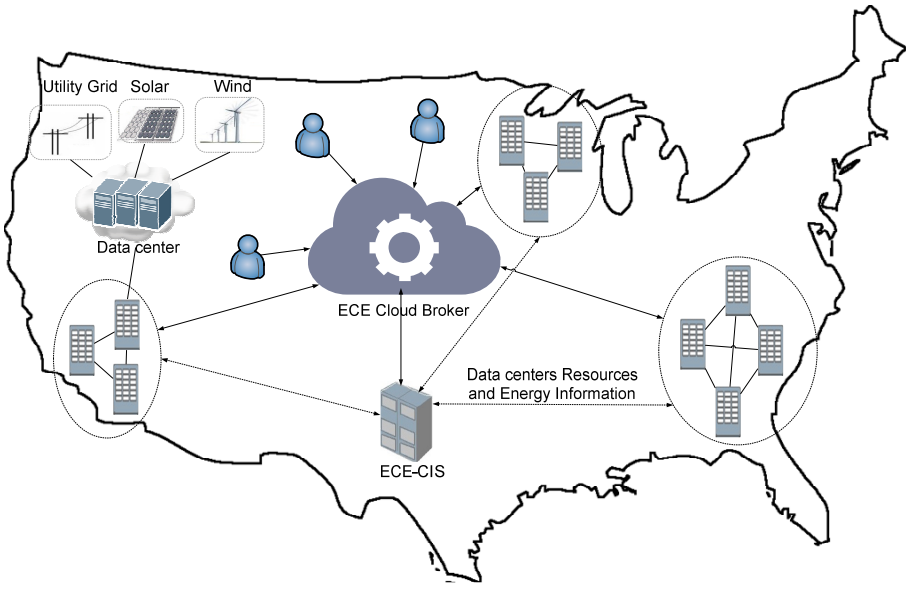


Figure 5.1: System model. ECE-CIS, Energy and Carbon Efficient Cloud Information Service.

5.3 System Specification and Problem Definition

5.3.1 System Model

The targeted system in this chapter is an IaaS cloud provider offering VM resources to its clients similar to *Elastic Compute Cloud* (EC2) service by Amazon Web Services [70]. The cloud provider, as shown in Figure 5.1, consists of several geographically distributed regions. Each region is isolated from other regions and consists of several availability zones. The availability zones in the regions are data centers connected through low latency links. Hereafter, whenever we talk about data centers, we refer to the availability zones within one region. We only consider VM migration between data center sites belonging to the same region, as the network cost and delay associated with that is acceptable [70]. To the best of knowledge, we are the first considering VM migration between cloud data centers to maximize renewable energy utilization.

A cloud user (hereafter called user) at the time of submitting a VM provisioning request can choose the availability zone he/she wants to run the VM in or leaves the availability zone selection up to the cloud provider. Users submit VM provisioning requests

through a cloud interface called a cloud broker (hereafter called broker). This connects them to the cloud provider and enables the users to monitor and follow the status of their requests. Broker, as discussed in Chapter 3, is a major component of the provider. It is responsible for receiving VM requests, performing the VM placement and migrating the currently running VM to another data center, in case of failure, maximizing renewable energy usage, or any other purpose. The information needed by the broker to function is provided by the directory called Energy and Carbon Efficient Cloud Information Service (ECE-CIS). Data centers register themselves at the ECE-CIS and keep it updated regarding any changes in their current state. The information sent by data centers to the ECE-CIS include: available physical resources, data center's PUE, energy source(s), amount of available renewable energy, carbon footprint rate, and physical servers' current utilization. Note that PUE stands for power usage effectiveness and is a metric coined by the Green Grid consortium [78] to represent a data center's energy efficiency. Power usage effectiveness is the ratio of total power consumed by the data center to its power consumed by IT devices.

As shown in Figure 5.1, data centers might use their own on-site renewable energy sources to power their infrastructure and servers in addition to the electricity delivered from off-site grid. The off-site grid energy is usually derived from polluting sources, known as brown energy, so cloud providers are deploying their own on-site renewable energy sources with the aim of cost saving and social impact. Two renewable energy sources considered in this work are solar photovoltaic (PV) and wind, as they are the most common and the fastest growing ones. Solar energy, as can be seen in Figure 5.4a, has a raised cosine distribution during the day, therefore its peak energy level varies by change in time zone for different locations. In contrast, wind energy does not have a clear predictable pattern, as shown in Figure 5.4b. Having these two renewable sources in a data center provides access to clean energy to run requests during different times of the day.

Table 5.2: Description of symbols.

Symbol	Description	Symbol	Description
$\mathcal{D}$	Set of data center sites	n	Number of data center sites
$\mathcal{S}$	Set of physical servers (hosts) in a data center	h	Number of physical servers (hosts)
$C_{execution}$	Energy cost to execute the VMs	C_{extra}	Extra cost due to VM migration
$C_{migration}$	Energy cost to migrate the VMs	$C_{addBrown}$	Additional brown energy consumption at the source data center during VM migration
C_s	Cost of server energy consumption	C_o	Cost of overhead energy consumption
c_n	Cost of network to migrate the VM per unit time	c_b	Cost of brown energy per unit time
p_r	Price of renewable energy per unit usage	p_b	Price of brown energy per unit usage
E_r	Servers total renewable energy consumption	E_b	Servers total brown energy consumption
t_m	Start time of VM migration at the source data center	T_m	Duration to migrate the VM
t_b	Start time of brown energy consumption at the source data center	T_b	Duration of brown energy usage during VM migration at the source data center

5.3.2 Preliminaries

We consider a cloud provider with a set of n data center sites, shown as $\mathcal{D} = \{d_1, d_2, ..., d_n\}$, distributed in a geographical region. Each data center is referred to as an availability zone which consists of a set of h servers/hosts shown as $\mathcal{S} = \{s_1, s_2, .., s_h\}$. The list of all the symbols used in this chapter are given in Table 5.2.

Total Cost. The total cost of energy, Equation (5.1), is the cost of energy used to run/execute VMs in the data center and the extra cost.

$$C_{total} = C_{execution} + C_{extra}\,. \tag{5.1}$$

Extra Cost. Extra cost, Equation (5.2), is associated with the energy used to migrate VMs between data center sites and the additional brown energy usage in the source data center while VM migration takes place.

$$C_{extra} = C_{migration} + C_{addBrown}\,. \tag{5.2}$$

The aforementioned costs (execution, migration, and additional brown) can be detailed as follows:

Execution Cost. Execution cost is the energy cost to run VMs in the data center and is shown in Equation (5.3). The energy cost to run VMs consists of server (C_s) and overhead (C_o) costs imposed due to running VMs within a data center.

$$C_{execution} = C_s + C_o \,. \tag{5.3}$$

To calculate overhead energy, we use PUE that is equal to the total energy goes to a data center divided by the total energy consumed by IT devices and is computed as

$$PUE = \frac{C_s + C_o}{C_s} \,. \tag{5.4}$$

As a result,

$$C_{execution} = C_s \times PUE \,. \tag{5.5}$$

Server Cost. Servers host the incoming workload and based on their configuration are capable to accommodate different number of VMs. The cost of servers C_s is computed as follows:

$$C_s = p_r \times E_r + p_b \times E_b \,, \tag{5.6}$$

where E_r and E_b are the energy consumption of servers using renewable and brown energy sources and p_r and p_b are the related prices, respectively.

The energy consumption of servers is the product of the power consumption of servers and the time period they have been working under that power. The power consumption depends on several hardware resources including CPU, memory, and disks [116]. According to Blackburn and Grid [86], the total power consumed by a server is determined by the incoming load to that server, which is shown as CPU utilization. The relationship between the server power consumption and CPU utilization can be a constant, cubic, or quadratic [52].

Migration Cost. Migration cost is part of the extra cost and is the energy consumed by the network to migrate the VMs between data center sites. Live migration of VMs

requires relocating the VMs and placing them in their new destination [69]. The cost due to transferring the VMs is proportional to the VM size and the number of bytes that need to be transferred between data center sites, similar to AWS EC2 pricing [70]. For the sake of simplicity, we limit the migration cost to a specific type of VM with a constant network cost c_n per unit time for the live migration, and T_m is the time required to perform and complete the migration. Equation (5.7) represents the migration cost.

$$C_{migration} = c_n \times T_m \,. \tag{5.7}$$

Additional Brown Energy Cost. This part represents the penalty of brown energy consumption while VM migration takes place at the source data center. As mentioned earlier, we consider two different types of energy sources: brown and renewable. The renewable energy is drawn from on-site solar and wind power generators. Therefore, there is a one-time installation and fixed maintenance cost for them; which leads to very low price per unit usage in their lifespan. On the other hand, brown energy is derived from off-site electricity that, besides its high per unit usage cost, also leaves carbon dioxide in the environment. We show the brown energy cost for the specific type of VM as c_b per unit time and the time duration brown energy has been used while migration takes place as T_b. Therefore, the cost of additional brown energy usage can be shown as

$$C_{addBrown} = c_b \times T_b \,. \tag{5.8}$$

5.3.3 System Objective

Considering prices of different energy sources and their environmental impact, running VMs using renewable energy sources eventually leads to a lower total cost. We consider powering data centers using renewable energy unless it is not available. Since renewable energy sources have intermittent nature, there is the possibility of their shortage in the lifetime of a VM running in the data center. In this case, cloud provider could migrate the VM to another availability zone with excess renewable energy available. Performing VM migration could lead to lowering and even eliminating brown energy usage, but it

imposes extra costs to the system. In this work, our objective is to *minimize the total cost of running VMs in the system through VM migration*. As shown in Equation (5.9), the objective function consists of energy used in data centers to run VMs, and extra energy used to migrate VMs to the data center with access to renewable energy and the additional brown energy used in the source data center while migration takes place.

$$\text{minimize } C_{execution} + C_{extra}\,. \tag{5.9}$$

The first part (execution cost) in the objective function is inevitable even if no migration takes place. Therefore, to achieve our goal we restate the objective function as *to minimize the extra cost due to VM migration*. However, optimal cost minimization within a data center with very large number of VMs is a complex problem. We narrow down our formulation to a single VM migration problem, which eventually leads to overall cost minimization when the cost for the individual VM is minimized.

5.3.4 Virtual Machine Migration Problem

To maximize renewable energy usage and be aligned with the system objective, we perform VM migration in the absence of renewable energy. The extra energy, Equation (5.2), consists of the energy used by the network, Equation (5.7), and additional brown energy used at the source data center, Equation (5.8), while the VM migration takes place.

We break down the extra cost into three different cases, as shown in Equation (5.10) and Figure 5.2.

$$C_{extra} = \begin{cases} C_1 & \text{if } t_m < t_b \;\; and \;\; t_b - t_m \geq T_m, \\ C_2 & \text{if } t_m < t_b \;\; and \;\; t_b - t_m < T_m, \\ C_3 & \text{if } t_m \geq t_b. \end{cases} \tag{5.10}$$

where

$$\begin{aligned} C_1 &= c_n.T_m\,, \\ C_2 = C_3 &= c_n.T_m + c_b.(t_m - t_b + T_m)\,. \end{aligned} \tag{5.11}$$

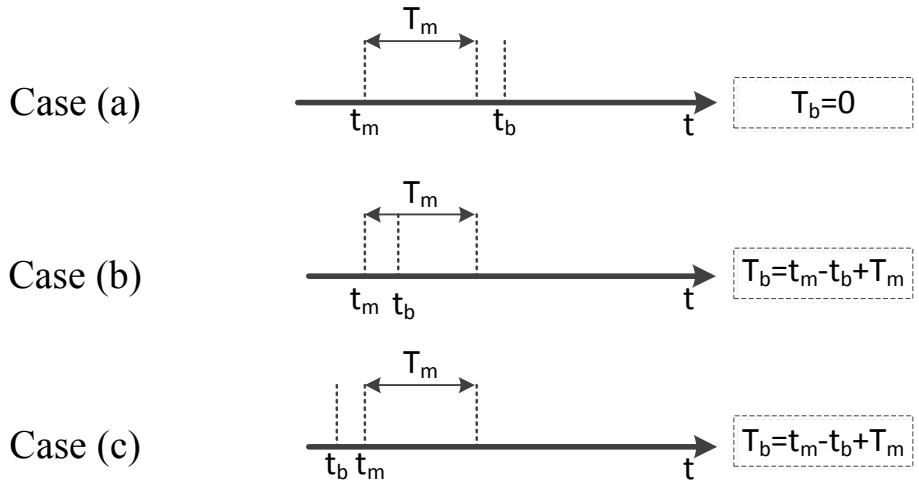


Figure 5.2: Example of migration time (t_m) versus start time of brown energy consumption (t_b).

The first case (C_1) indicates when the VM migration starts at t_m and finishes before the start of brown energy consumption t_b. This is shown in case (a) in Figure 5.2 as well and can be formulated as: $t_m < t_b$ and $t_b - t_m \geq T_m$. Therefore, the time duration required for VM migration to be completed T_m comes to at end before the data center starts to use brown energy sources and the only extra cost in this case is the migration cost as shown in Equation (5.11). C_2, case (b) of Figure 5.2, occurs when migration starts before finishing of renewable energy $t_m < t_b$, but it completes after start of brown energy usage $t_b - t_m < T_m$. As shown in Equation 5.11, besides the migration cost, the cost of brown energy usage in the source data center is added to the extra cost as well. Finally, C_3 which is the case (c) in Figure 5.2, occurs after the time no renewable energy is available in the data center, i.e.,$t_m \geq t_b$.

5.4 Optimal Offline Virtual Machine Migration

In this section, we study the offline solution of a single VM migration problem among data center sites to increase the usage of renewable energy sources. Without loss of generality, we assume that the brown energy cost per unit time to be 1 and normalize the

network cost c_n to the brown energy cost, as shown in Equation (5.12).

$$c_b = 1 \text{ and } c_n = s; \text{ where } s \in \mathbb{R}^+ . \tag{5.12}$$

Moreover, we consider the following relation for t_b, t_m, and T_m.

$$t_b - t_m = aT_m; \text{ where } a \in \mathbb{R} . \tag{5.13}$$

Considering Equations (5.12) and (5.13), we rewrite Equations (5.10) and (5.11) as follows:

$$C_{extra} = \begin{cases} C_1 = s.T_m & \text{if } a \geq 1, \\ C_2 = C_3 = s.T_m + (1-a)T_m & \text{if } a < 1. \end{cases} \tag{5.14}$$

Theorem 5.1. *The optimal offline (OPT) cost is $s.T_m$.*

Proof. However, finding the optimal offline cost associated with Equation (5.14) is straight forward, we provide the detailed proof for better understanding of VM migration problem under different system conditions. In order to find the optimal offline solution, we need to find the condition where the cost function has the minimum cost. Based on Equation (5.14), we have

1. C_1 equals $s.T_m$, where $a \geq 1$.

2. If $a < 1$ then $1 - a$ is always a positive value and C_2 or C_3 are always greater than $s.T_m$, which means $C_2 > C_1$ or $C_3 > C_1$.

As a result, the optimal offline happens at $a \geq 1$ or $t_b - t_m \geq T_m$. This means that the optimal offline happens when migration starts and finishes before the start of brown energy usage in the data center. This leads to the optimal offline cost $s.T_m$. □

5.5 Online Virtual Machine Migration

In this section, we construct two online algorithms to minimize cost of VM migration. The reason for proposing online algorithms is that optimal offline algorithm is only attainable when we have full future knowledge about the system and renewable energy level. Here we propose two deterministic online VM migration algorithms: *optimal online deterministic (OOD)* VM migration with no future knowledge and *future-aware dynamic provisioning (FDP)* VM migration with limited knowledge (up to a window-size) regarding renewable energy level. Our online algorithms are inspired by ski-rental problem [117]. We decide when to migrate a VM to another data center with excess renewable energy to minimize brown energy consumption. It should be noted that the decision to whether or not to migrate a VM to another data center is considered to be happening in serial. Making decision to migrate the VMs in this way, we assume that we only make decision regarding migration of one VM at a time and our knowledge about the renewable energy level at the destined data center is precise to large extent. Moreover, in our model we keep two copies of VM while migration and switching is happening. Keeping a copy of the VM in the source data center till VM migration fully completes assures that user experience in terms of latency and response time would not be affected by the migration time and network delay.

In order to be able to evaluate the performance of our online algorithms, we use the competitive ratio analysis [118].

Definition. *An online algorithm is called c-competitive if, for all possible inputs, the outcome of the online algorithm (C_A) in comparison to the optimal offline outcome (C_{OPT}) has the following relation: $C_A / C_{OPT} \leq c$.*

5.5.1 Optimal Online Deterministic Virtual Machine Migration

Our goal is to propose an algorithm that could achieve optimal result using only the current information available. Theorem 5.2 shows the optimal online deterministic algorithm for a single VM migration problem is attained when migration takes place by the beginning time of brown energy usage, that is, $t_m = t_b$.

Theorem 5.2. *The optimal online deterministic algorithm is achieved when $t_m = t_b$ and it is $(1 + 1/s)$-competitive.*

Proof. Based on the cost function in Equation (5.14) and Theorem 5.1, we can write the competitive ratio for any arbitrary online algorithm with no future knowledge as follows:

$$\frac{C_{OOD}}{C_{OPT}} \leq \begin{cases} \dfrac{s.T_m}{s.T_m} = 1 & \text{if } a \geq 1, \\ \dfrac{s.T_m + (1-a)T_m}{s.T_m} = 1 + \dfrac{1-a}{s} & \text{if } a < 1. \end{cases} \tag{5.15}$$

where $a = \dfrac{t_b - t_m}{T_m}$ as defined in Equation (5.13).

Any online algorithm with no future knowledge can only have the knowledge of the current time t_i, and t_b if $t_b \geq t_i$, that is, the time from which VM started using brown energy. Accordingly, two different groups of online algorithms with no future knowledge can be defined that they set t_m as a function of

1. the current time t_i, i.e., $t_m = f_1(t_i)$, and

2. the start time of brown energy usage, i.e., $t_m = f_2(t_b)$.

For algorithms from the former group, $a = \dfrac{t_b - f_1(t_i)}{T_m}$, since a is not a function of t_b, a can grow arbitrarily large when the adversary will select t_b such that it is infinitely greater than $f(t_i)$, i.e., $a \to \infty$, and as a result, $\dfrac{C_{OOD}}{C_{OPT}} \to \infty$. Therefore, all algorithms from the first group are not competitive.

For algorithms from the latter group, $a = \dfrac{t_b - f_2(t_b)}{T_m}$, the time of migration t_m is dependent to the start time of brown energy usage t_b, which is known for the algorithm, therefore

$$\begin{aligned} & t_m \geq t_b\,, \\ \text{as} \quad & a = \frac{t_b - t_m}{T_m} \Rightarrow a \leq 0\,. \end{aligned} \tag{5.16}$$

Considering $a \leq 0$, the minimum competitive ratio is achieved when $a = 0$ for the second inequality in Equation (5.15). This means migration starts by the beginning of brown energy usage, i.e., $t_m = t_b$. As a result, the best competitive ratio is $1 + \dfrac{1}{s}$. □

5.5.2 Future-Aware Dynamic Provisioning Virtual Machine Migration

As mentioned earlier, we consider access to renewable energy sources along with the electricity derived from off-site grid. Two renewable sources consider in this chapter, solar and wind, have different pattern during the day. As shown in the Figure 5.4a, solar energy has a predictable pattern during the day and its peak is foreseeable. In contrast, wind energy does not have a predictable diurnal pattern. But one can use the average temporal pattern of wind energy, which can be captured in the region [119]. It is often assumed that the renewable energy availability in the near look-ahead window can be predicted with a good accuracy in reality, such as auto-regressive techniques used in the works by Kansal et al. [120] and Cox [121]. If there are prediction errors in the model, decisions would be affected by the same error margin as prediction errors. For example, 10% prediction error causes 10% error in decision making. If the time window is small enough, such as minutely windows, then renewable energy prediction can be predicted with considerably high precision almost similar to real time measuring; therefore, it will not affect the decisions significantly. In Chapter 6, we present a prediction model for renewable energy that can predict up to 15 minutes ahead into the future with nearly 98% accuracy around $\pm$10% of the actual values. The question is how much knowledge can help and get the online algorithm performance close to the optimal offline algorithm.

We assume that at any given time, t_i, the future renewable energy is predictable for a window-size ω, which means the amount of renewable energy in the system is known for the period $[t_i, t_i + \omega]$. Now we elaborate on how the window-size affects the decision making process and improves the online algorithm performance. The following two cases are plausible:

1. If window-size is greater or equal to the time required to perform the migration, $\omega \geq T_m$, it would be the same as the scenario for optimal offline algorithm. Therefore, there is enough time to migrate the VM to a data center with access to renewable energy and avoid brown energy usage.

2. If window-size is smaller than the time of migration, $\omega < T_m$, then $t_m + \omega \geq t_b$.

Theorem 5.3. *The competitive ratio for the future-aware dynamic provisioning algorithm is:*

Algorithm 3: Most Available Renewable Energy (MARE) VM Placement Algorithm

```
Input: datacenterList, hostList
Output: destination
while vmRequest do
    Get data centers' Information from ECE-CIS;
    foreach datacenter in datacenerList do
        availSolar ← Get Current availableSolar;
        availWind ← Get Current availableWind;
        availRenewable ← availSolar + availWind;
    Sort datacenerList in a descending order of availRenewable;
    foreach datacenter in datacenterList do
        foreach host in hostList do
            if host is suitable for vm then
                destination ← (data center, host);
                return destination;
    destination ← null; //rejection of request;
    return destination;
```

$$\frac{C_{FDP}}{C_{OPT}} \leq 1 + \frac{1}{s} - \frac{\omega}{s.T_m}, \textit{where } w \leq T_m.$$

Proof. The optimal offline algorithm migrates the VM, T_m unit of time earlier than t_b, and the optimal online deterministic algorithm with no future knowledge migrates the VM by the time of t_b. The FDP algorithm with limited future knowledge minimizes the cost when migrates the VM as soon as t_b is known. That is, FDP can migrate the VM at most up to ω unit of time earlier, when t_b can be seen within the look-ahead window. Therefore, there would be ω unit less brown energy consumption, which improves the online algorithm cost. Equation (5.17) shows the competitive ratio for future-aware online algorithm.

$$\frac{C_{FDP}}{C_{OPT}} \leq \frac{s.T_m + T_m - \omega}{s.T_m} = 1 + \frac{1}{s} - \frac{\omega}{s.T_m}, \tag{5.17}$$

where $w \leq T_m$ as competitive ratio is always greater or equal to 1. □

5.5.3 Virtual Machine Placement

By the arrival of each VM request, the broker should allocate resources to the VM and for this purpose it needs to decide where to place the VM. We treat VM placement as a bin-packing problem with different bin sizes, which are physical servers in this context. Since bin-packing is an NP-hard problem, we use derivation of best-fit heuristic to solve it. To be aligned with our purpose and taking the most from available renewable energy in distributed data centers, we consider a modification of the best-fit heuristic that we proposed in Chapter 3. The modification to the *ECE* algorithm in Chapter 3 is denoted as *Most Available Renewable Energy (MARE)*. The MARE sorts data center sites according to the amount of available renewable energy and submits the VM to the data center with the highest available amount. The pseudocode of the VM placement algorithm is presented in Algorithm 3.

The time complexity of Algorithm 3 with v VM requests, n data center sites, and h physical servers within each data center in detail is as follows: Lines 3-6 take $O(n)$ and the sort function in Line 7 can be done in $O(n\log(n))$. Lines 8-12 take $O(nh)$, in the worst case. Thus, the total running time for the algorithm is $O(v(n + n\log(n) + nh))$. Since the number of VM requests and hosts dominate the total number of data center sites (n), the total time complexity of the algorithm is $O(vnh)$.

In addition, we consider another VM placement algorithm without any knowledge regarding renewable energy availability, denoted as *Random* algorithm. By the arrival of a new VM request, Random algorithm chooses a random data center uniformly.

5.6 Performance Evaluation

We perform simulation-based experiments to evaluate our proposed algorithms. Our aim is to measure the energy cost savings incurred due to migration of VMs to data centers with access to renewable energy sources. Moreover, we measure the improvement made by applying the proposed approaches in cutting the amount of carbon emission.

Workload Data. We use Google cluster-usage traces [14] for workload as there is no other publicly available real-world workload traces for IaaS cloud providers to the best of our

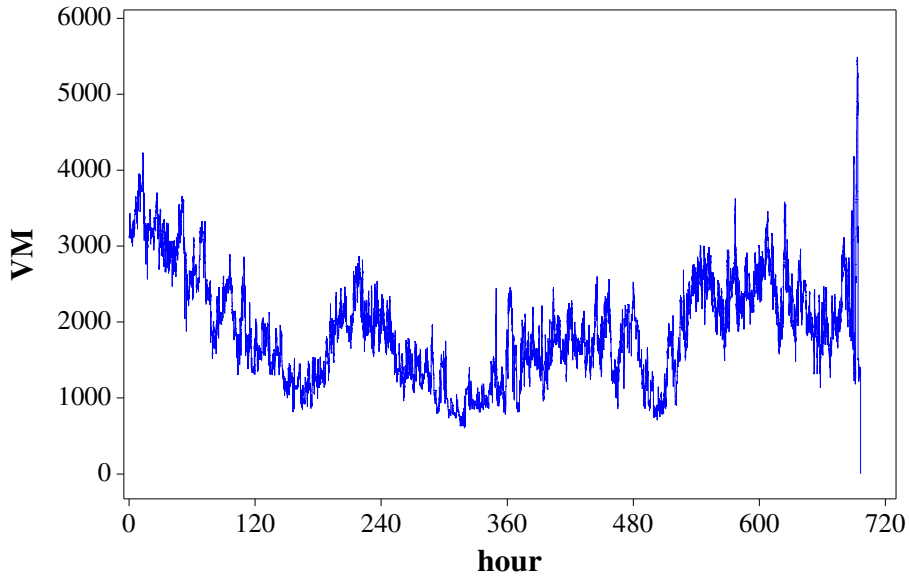


Figure 5.3: One-month Google workload trace.

knowledge. The Google dataset has records of one cluster's usage (which is a set of 12,000 physical machines) and includes submitted requests to that cluster over a period of one month. Each request has requirements shown as amount of requested CPU, memory, and storage. Because these traces are user requests not representing VM instances demand, we need to make a mapping between request submissions from users to IaaS computing demand. We use the same technique used by Toosi et al. for their workload generation to generate VM request traces [122].

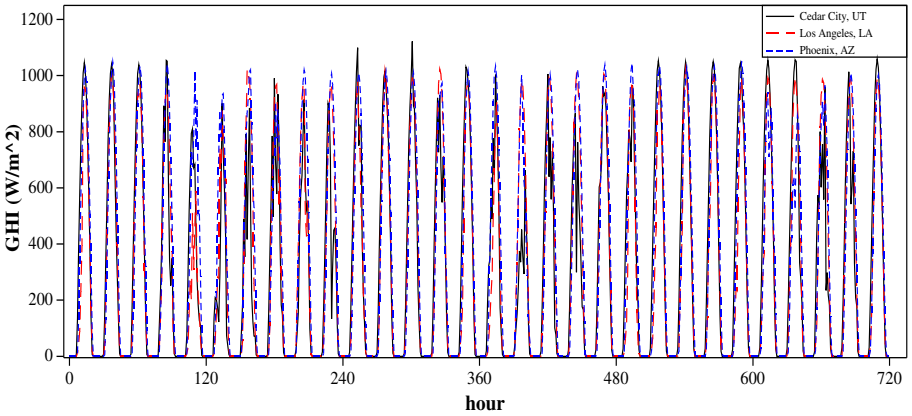
Google traces include record of users, each submitting several tasks, with specific resource requirements. Considering the fact that 93% of the Google cluster machines have the same computing capability, we assume all physical machines in the cluster have the same resources (in terms of CPU, memory, etc.) and map our VM size to that of the physical machine. To derive VM request traces from Google traces, whenever a user submits a task, we check if there is already a VM instantiated by that user in the system with enough computing resources to run the new task. Otherwise, if there is no VM owned by the user with enough capacity to accommodate the new task, we instantiate a new VM to serve the user's task [122]. We also terminate a VM when there is no task running on it. By this, we can create a trace of VM requests submitted from users. The trace contains

250,171 VM requests, each has the start time and holding time in the data center. We consider the VM specifications in our model similar to the *standard small instances* introduced by Amazon EC2 [70]. Figure 5.3 shows the number of VM requests per hour received by the provider, generated based on the scheduling algorithm we used to generate VM requests according Google cluster traces. This figure shows the shape of the workload and its fluctuation in our simulations.

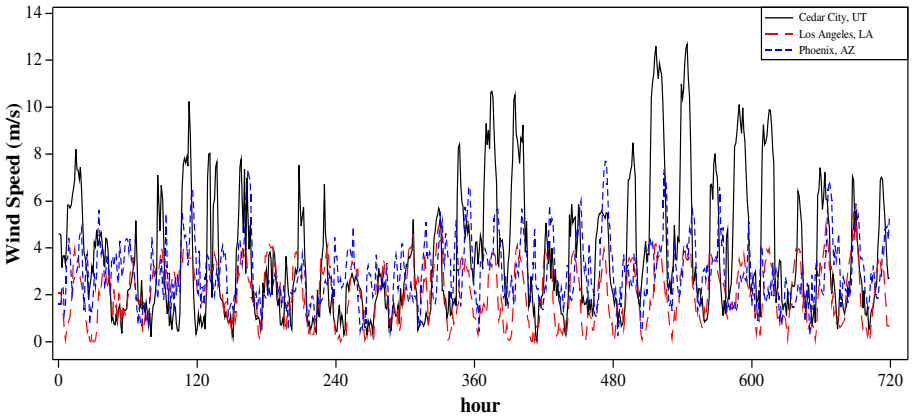
Data Centers' Configuration. We consider 3 data center sites located in the US-West region. The locations are chosen from the data centers map [98], and are as follows: *Phoenix* in Arizona, *Los Angeles* in California, and *Cedar City* in Utah. The number of servers in each data center is set in a way that data centers' capacity would not be a limitation for not being able to take advantage of available renewable energy. Based on the previous discussion, the servers in the data centers are homogeneous with equal processing capacity. We model servers in data centers based on the latest HP ProLiant DL360 Gen9 server [123], with following specifications: Intel Xeon E5-2670v3, 10 cores $\times$ 2.3 GHz, 256 GB memory. The power consumed by each request running on a server within a data center is assumed to be on average a constant rate per time slot (e.g., 550W/hour).

We consider PUE value of 1.4 for all data center sites to calculate the overhead energy usage. The reason is that we aim to evaluate algorithms in a setting where PUE values are not determinative. We select the $carbon\ footprint = 0.350\ Tones/MWh$ for the off-site grid electricity, derived from the US Department of Energy Electricity Emission Factors [16]. The electricity price of $p_b = 6.22\,cents/kWh$ is chosen for the off-site electricity from the US Energy Information Administration [17]. This price represents the electricity price for the on-peak period, between 8AM and 10PM. We opt the off-peak price to be half of the on-peak. Moreover, as discussed earlier, we consider a fixed price for renewable energy usage per unit as $p_r = 1.0\,cents/kWh$.

Renewable Energy Traces. We use the measurements reported by NREL [15] for irradiance and meteorological data from different stations to capture wind and solar energy with 1-hour granularity from May, 1^{st} to May, 29^{th} 2013. To calculate the output for PV power, we use the hourly solar irradiance reported for flat plates on tilted surface at a 45-degree angle and PV efficiency of 30%. We calculate the solar output based on [124]



(a) Global horizontal irradiance (GHI).



(b) Wind speed.

Figure 5.4: Renewable energy traces.

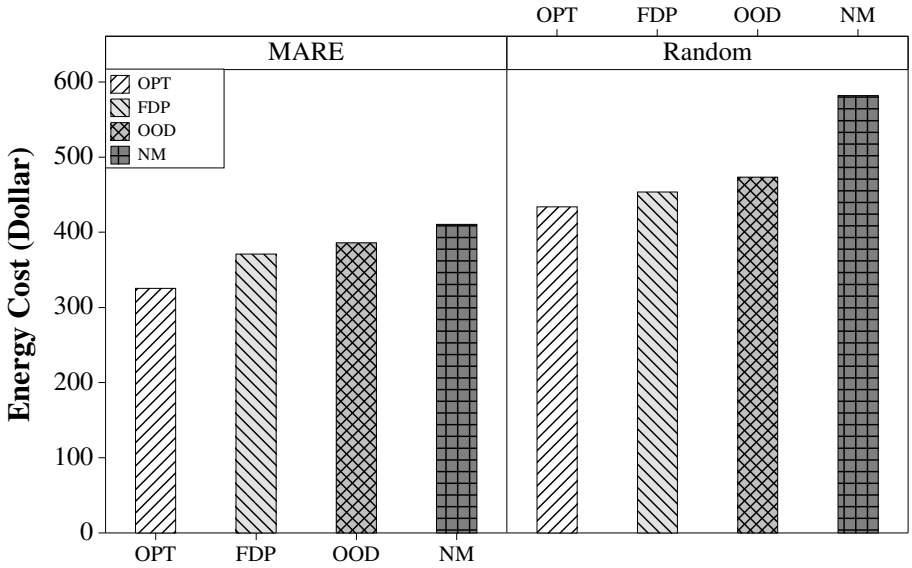


Figure 5.5: Total energy cost. FDP, future-aware dynamic provisioning; NM, no migration; OOD, optimal online deterministic; OPT, optimal offline.

and the total area for the flat plates is considered to be $100m^2$, derived from the configuration by Solarbayer [100]. To generate hourly wind energy, we use the proposed method by Fripp et al. [119]. The hourly wind speed, air temperature, and air pressure, derived from NREL measurements, are fed to the model and the generated power is computed accordingly, assuming each data center uses a GE 1.5MW wind turbine. Figures 5.4a and 5.4b show the solar and wind energy availability for three different cities in our system model, respectively.

5.6.1 Experiment Setup

Benchmark Algorithm. We compare the proposed offline, optimal online, and future-aware algorithms with a baseline benchmark algorithm with no VM migration. The benchmark does not take any further action and does not perform any migration after initial placement and instantiation of the VMs in the data ceners. The benchmark is referred to as No-Migration (NM) policy.

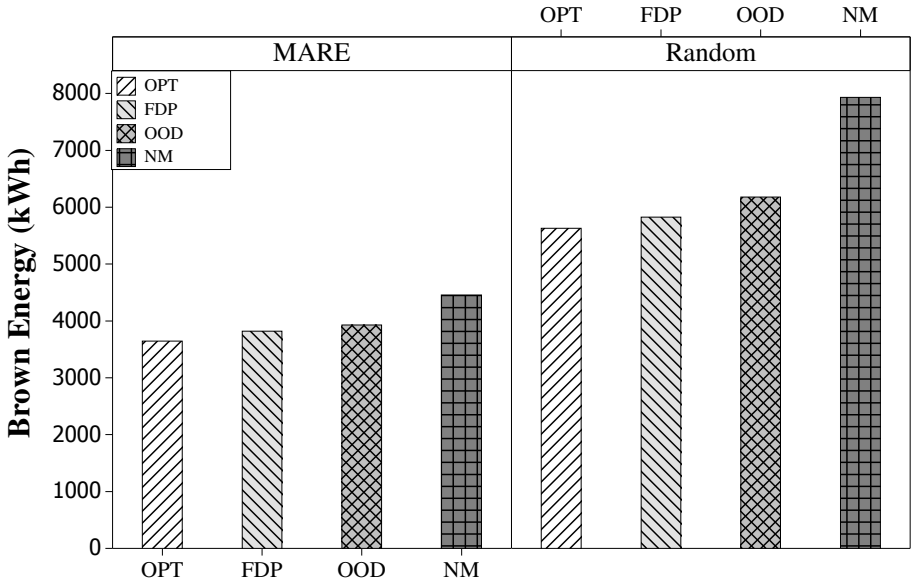


Figure 5.6: Brown energy consumption. FDP, future-aware dynamic provisioning; NM, no migration; OOD, optimal online deterministic; OPT, optimal offline.

5.6.2 Experiment Results and Analysis

In the experiments, we use the real-world traces derived from Google to study the performance of the proposed offline, optimal online, and future-aware algorithms all in combination with two VM placement policies against the benchmark algorithm. The results are shown in Figures 5.5 to 5.7.

Figure 5.5 shows the total energy cost incurred by all algorithms, when the window-size for future-aware algorithm is set to 4.5 minutes. The results indicate that having initial knowledge about the current renewable energy level in the data centers has a substantial effect in the amount of cost reduction. It can be seen that there is a significant cost reduction for policies under the MARE placement in comparison to the Random. Since offline policy has the full knowledge of renewable energy in the system, it achieves the lowest cost, 14% and 18.5% energy cost reduction in comparison to future-aware and online policies, respectively. Future-aware policy performs slightly better than optimal online algorithm and reduces the total cost by 4% in comparison to the optimal online policy that makes decision instantly without any future knowledge. The benchmark policy has the highest cost, since after placement of VMs and when there is no renewable

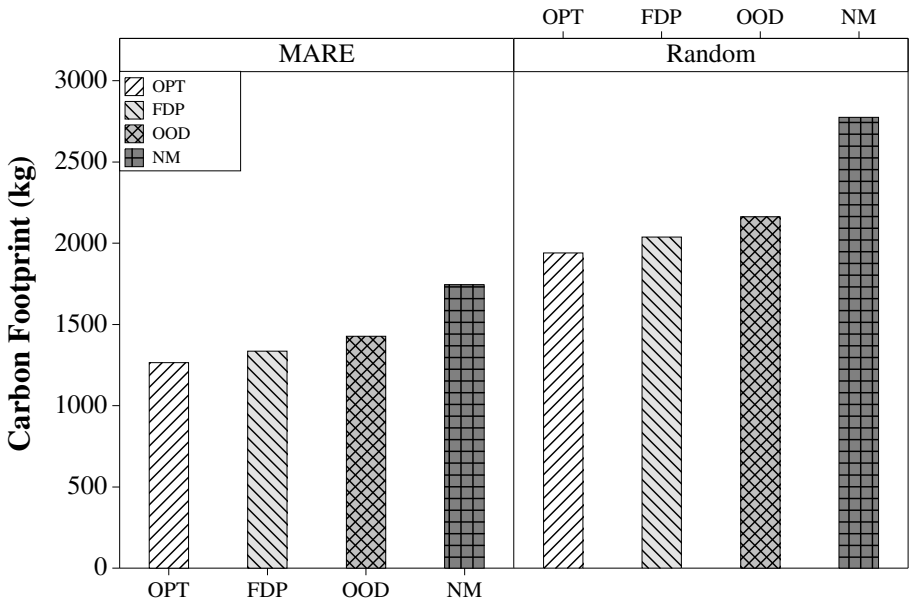
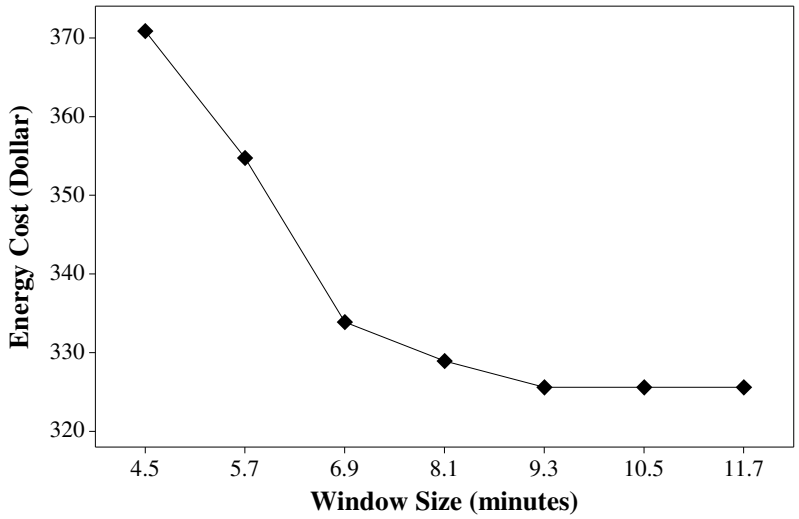


Figure 5.7: Carbon footprint. FDP, future-aware dynamic provisioning; NM, no migration; OOD, optimal online deterministic; OPT, optimal offline.

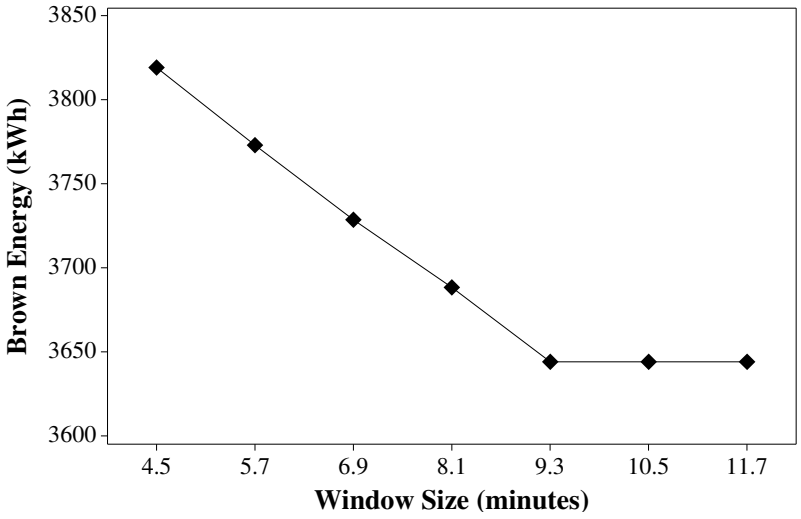
energy available in the data center it does not take any further action. The benchmark policy on average consumes 26% more energy cost in comparison to the optimal offline policy under different VM placement algorithms.

We also measured the amount of brown energy consumption as well as carbon footprint in the system as shown in Figures 5.6 and 5.7, respectively. Policies under MARE VM placement achieved considerable reduction in brown energy consumption in comparison to the case when VM placement randomly chooses destined data center. Within each category, offline with full knowledge of renewable energy consumes less brown energy, 5.6% and 12.9% less brown energy in comparison to future-aware and online policies, respectively. Future-aware and online policies reduce brown energy consumption by 30.5% and 22%, respectively, in comparison to the benchmark with no migration. The same behavior can be seen for carbon footprint in Figure 5.7, because reduction in brown energy consumption eventually leads to lower carbon footprint.

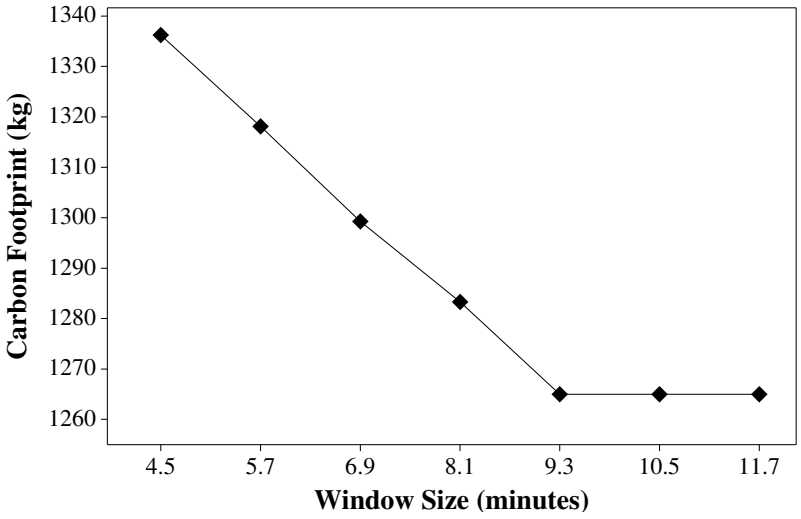
As shown, future-aware policy achieves results that fall between the outcome of the offline algorithm with full knowledge, and optimal online with no knowledge about future renewable energy level. We change the window-size to see its impact on the perfor-



(a) Energy cost.



(b) Brown energy consumption.



(c) Carbon footprint.

Figure 5.8: Effect of window-size on the results of future-aware dynamic provisioning algorithm under MARE VM placement policy.

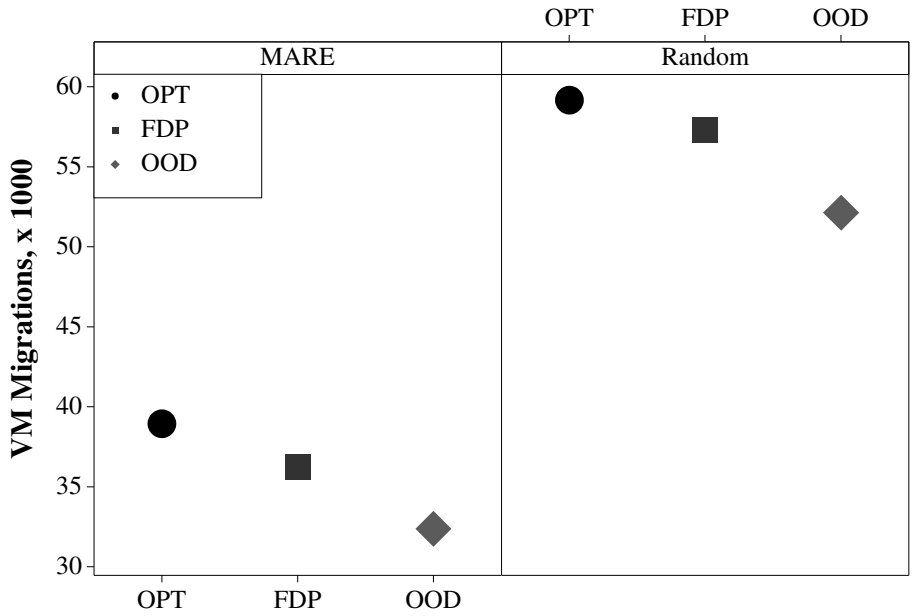


Figure 5.9: Number of virtual machine (VM) migrations. FDP, future-aware dynamic provisioning; MARE, most available renewable energy; OOD, optimal online deterministic; OPT, optimal offline.

mance of the future-aware dynamic processioning algorithm. As Figure 5.8 illustrates, increase in the window-size reduces total cost, brown energy consumption, and carbon footprint. Increase in the window-size makes future-aware algorithm closer to its offline competitor. The performance of the future-aware policy improves and gets close to the optimal offline until window-size reaches 9 minutes. After this point no improvement is achieved, since this is the point that window-size reaches the VM migration time in our experiments. This supports the theoretically proven supposition in Section 5.4 that if enough knowledge of future is available, the optimal decision suggests a VM migration that finishes before the start of brown energy usage in the data center.

As per Figure 5.5, the cost ratio of deterministic and future-aware online policies versus the optimal offline algorithm are 1.18 and 1.13, respectively. Moreover, based on the simulation setup $s = 3.5$, which leads to deterministic and future-aware online algorithms be 1.28 and 1.14 competitive in comparison to the optimal offline algorithm, respectively. The simulation results are compatible with the calculated competitive ratio as per the provided definition of c-competitive in Section 5.5.

Figure 5.9 depicts the total number of migrations happening in the system for each

policy during the one-month simulation period and total of 250,171 VM requests. We observe that migration policies under MARE placement achieve lower number of migrations in comparison to the same migration policies under Random placement. The reason is that under MARE placement, a wise data center selection is made for initial VM request placement which reduces the need for possible future migrations. Amongst three different migration policies, offline has the highest number of VM migrations. Since it has full knowledge of the amount of renewable energy in the system and begins to migrate the VMs before the start time of brown energy usage, unless there is no renewable available in other data centers. Similarly, future-aware policy makes more VM migrations than online policy, due to further knowledge regarding renewable energy level.

5.7 Summary

Using on-site renewable energy sources instead of electricity derived from off-site grid helps cloud providers to reduce their energy cost and their reliance on polluting energy sources. Since the nature of renewable energy sources (solar/wind) is intermittent, we take advantage of having access to several geographically distributed data center sites of a cloud provider to perform intra-region VM migration and utilize the most of the available renewable energy. In this chapter, we introduced algorithms with full and partial knowledge of future availability of renewable energy levels to migrate the VMs to another data center within a region in the absence of sufficient renewable at the host data center. We first introduced the optimal offline algorithm to minimize the energy cost. Because of the necessity of having full knowledge of future level of renewable energy for optimal offline, we propose two online algorithms. The first online algorithm is a deterministic algorithm that does not have any knowledge regarding the future level of renewable energy and the second one is denoted as future-aware online algorithm with limited knowledge, up to a window-size (ω), of future level of renewable energy. We have compared the results of the proposed optimal offline, optimal online, and future-aware algorithms with a basic benchmark algorithm that does not perform any migration, all in combination with two VM placement algorithms. One VM placement is aware

of the current renewable level, known as MARE, and the other one randomly chooses the destined data center.

We have evaluated the proposed algorithms through extensive simulations using real-world traces for renewable energy (solar and wind) and one-month workload trace of a Google cluster usage. The offline algorithm with full knowledge of renewable energy level performs the best in comparison to the future-aware and optimal online algorithms. The optimal online algorithm incurs 18.5% more cost compared to the offline algorithm when no future knowledge is available. Moreover, simulation results show that future-aware algorithm's performance gets competitive with offline algorithm by the increase in its window-size until the window-size reaches the network delay or the time needed that a migration takes place and gets completed. Next chapter introduces a short-term prediction model of renewable energy that can be used by the future-aware online algorithm to improve its performance and bring it close to the performance of the optimal offline algorithm.

Chapter 6

Short-Term Prediction Model to Maximize Renewable Energy Usage

The increasing demand for services offered by cloud providers results in a large amount of electricity usage by their data center sites and a high impact on the environment. This has motivated many cloud providers to move towards using on-site renewable energy sources to partially power their data centers using sustainable sources. This way, they can reduce their reliance on brown electricity delivered by off-site providers, which is typically drawn from polluting sources. However, most sources of renewable energy are intermittent and their availability changes over time. Therefore, having short-term prediction helps the cloud provider to make informed decisions and migrate the virtual machines (VMs) between data center sites in the absence of the renewable energy. In this chapter, we propose a short-term prediction model using Gaussian mixture model (GMM). The model uses the previously observed energy levels to train itself and predict the energy level for many-steps ahead into the future. We analyzed the accuracy of the proposed prediction model using real meteorological data. The experiment results show that the GMM model can predict up to 15 minutes ahead into the future with nearly 98% accuracy around $\pm$10% of the actual values. This helps the cloud provider to perform online VM migration with performance close to the optimal offline algorithm, which has the full knowledge of renewable energy level in the system. Moreover, the accuracy of the model has been verified using the workload data from Amazon biggest region in US East (N. Virginia). However, due to the confidentiality of that data set, we only rely on the results of the carried experiments using real meteorological renewable energy traces.

This chapter is derived from the publication: Atefeh Khosravi and Rajkumar Buyya, *"Short-Term Prediction Model to Maximize Renewable Energy Usage in Cloud Data Centers"*, Sustainable Cloud and Energy Services: Principles and Practice, W. Rivera (editor), Springer International Publishing AG, 2017 (in press, accepted in April 2017).

6.1 Introduction

DATA centers are the backbone of the Internet that consist of thousands of servers. They are one of the fastest growing industries that offer different types of services to users around the world. However, data centers are known to consume huge amount of electricity. According to a report by NRDC [125], US data centers in 2013 alone consumed 91 billion kilowatt-hours of electricity. This is equivalent to two-year power consumption of New York City's households and by 2020 is estimated to increase to 140 billion kilowatt hours. This could be equivalent to nearly 150 million tons of carbon pollution. Therefore, many cloud service providers focused on reducing their reliance on electricity driven from fossil fuels and transition to renewable energy sources.

Recently, large cloud providers started building their on-site renewable energy sources. Companies such as Amazon [107], Facebook [105, 106], Google [109], and Microsoft [110] all have their own on-site solar/wind farms. Renewable energy sources have intermittent nature. This means that their availability changes during the day and based on time of the year. However, since all the large cloud providers have geographically distributed data center sites, they can benefit from this location diversity. This helps them to migrate the VMs in the absence of renewable energy in a data center to a site with excess renewable energy.

Since, most sources of renewable energy have intermittent nature knowing the future level of energy helps the cloud provider to make informed decision on when to migrate the VMs to maximize renewable energy usage. The cloud provider can benefit from short-term prediction of renewable energy to perform future-aware online algorithms to migrate the VMs, as it has been stated in Chapter 5. This helps the provider to increase the performance of the online algorithms close to the optimal offline, which has full knowledge of the future level of renewable energy.

In this chapter, we propose a short-term prediction model based on the Gaussian mixture model [126]. The proposed model predicts renewable energy level for many-steps ahead into the future. A primary requirement to perform prediction is knowing the current and previous states of the renewable energy levels, since the future level can be inferred from current and previous states and their correlation. The GMM model uses his-

tory data to train itself. We use renewable energy measurements reported by NREL [15] as history and test data in our experiments. Moreover, we verified the accuracy of the proposed prediction model using workload demand collected from AWS biggest region, US East, Virginia. However, due to the confidentiality of the data set, we only rely on the analysis carried out using renewable energy traces collected from NREL that have been used in Chapter 5.

The rest of the chapter is organized as follows: Section 6.2 describes the prediction model objective. The formulation and component estimation of the prediction model is explained in Section 6.3. Section 6.4 elaborates on the required steps to construct the model. The approaches and methodologies to train the history data are explained in Section 6.5. Experiment results are presented in Section 6.6 and Section 6.7 provides a summary of the chapter.

6.2 Prediction Model Objective

Energy production at a data center within time period $[1, T]$ is time-series data and can be shown as $\mathbf{y} = [y_1, y_2, .., y_T]^T$, where y_t is the energy production at time t. We show the predicted renewable energy production in a data center at time t as $\hat{y}_t$. The closer the predicted energy $\hat{y}_t$ is to the observed production energy y_t, the more accurate the prediction.

Therefore, our objective is to minimize the prediction error over time interval $[t_1, t_2]$ where $t_1 \leq t_2$, and is stated as follows:

$$\begin{aligned} &\underset{\hat{y}_t}{\text{minimize}} \quad \sum_{t \in [t_1, t_2]} e[(\hat{y}_t - y_t)]\,, \\ &\text{subject to} \quad \hat{y}_t \geq 0\,, \\ &\qquad \text{and} \quad predictionModelCost \leq ThresholdCost\,. \end{aligned} \tag{6.1}$$

The first constraint guarantees the predicted energy production always has non-negative value. Finally, the second constraint guarantees the computation cost of running the prediction, in terms of running time, CPU, and memory usage, over a certain time period

will not exceed a predetermined threshold.

6.3 Prediction Model Formulation

We use the current and previous states of the energy production to perform prediction. The next state of energy production has strong but not deterministic relationship with the current and previous states. This relationship could be shown as a conditional probability. If we denote the current state of the energy production as y_t then the probability of the next state can be denoted as:

$$p(y_{t+1}|y_t, y_{t-1}, ..., y_{t-N+1}) , \tag{6.2}$$

where N is considered as the number of previous states taken into account for the prediction. For the sake of simplicity, we show the previous states considered in the prediction as $\mathbf{x} = [y_t, y_{t-1}, ..., y_{t-N+1}]^T$. Therefore, to obtain the energy production prediction we need to compute the following conditional estimation:

$$\hat{y}_{t+1} = \mathbb{E}[y_{t+1}|\mathbf{x}] . \tag{6.3}$$

6.3.1 Prediction Using Gaussian Mixture Model

To perform the prediction in near future using historical renewable energy production, we use Gaussian mixture models (GMM). In order to obtain the prediction value, first we need to compute $p(y_{t+1}|\mathbf{x})$. Since the aforementioned probability is unknown, we use GMM to approximate it, assuming it is a combination of multiple Gaussian components [126]. GMM is a powerful tool for data analysis and is characterized by M number of mixtures/components, each with a given mean $\boldsymbol{\mu}$, variance $\boldsymbol{\Sigma}$, and weight ω. The GMM probability density function can be written as follows:

$$p(\mathbf{x}|\Theta) = \sum_{j=1}^{M} \omega_j \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) , \tag{6.4}$$

where

$$\begin{aligned}
&\Theta = \{(\omega_1, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), (\omega_2, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2), ...(\omega_M, \boldsymbol{\mu}_M, \boldsymbol{\Sigma}_M)\}\,, \\
&\sum_{j=1}^{M} \omega_j = 1\,, \\
&\mathcal{N}(\mathbf{x}, ; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) = \frac{1}{\boldsymbol{\Sigma}_j \sqrt{2\pi}} e^{-\frac{(\mathbf{x}-\boldsymbol{\mu}_j)^2}{2\boldsymbol{\Sigma}_j}}\,.
\end{aligned} \tag{6.5}$$

GMM parameters, Θ, can be estimated using the expectation-maximization (EM) algorithm [127]. EM is the most popular approach being used and it iteratively optimizes the model using maximum likelihood maximization.

As we mentioned before, the next energy production value has a conditional probability with the current and previously observed production:

$$\begin{aligned}
\hat{y} &= \mathbb{E}[y|\mathbf{x}] \\
&= \int y p(y|\mathbf{x}) dy\,.
\end{aligned} \tag{6.6}$$

Since $p(y|\mathbf{x})$ in the Equation (6.6) is not known, we use Bayes' Theorem for its estimation stated as follows:

$$p(y|x) = \frac{p(y, \mathbf{x})}{p(\mathbf{x})}\,, \tag{6.7}$$

and the joint probability distribution for y and $\mathbf{x}$, $p(y, \mathbf{x})$, could be derived using GMM. Therefore, Equation (6.7) could be restated as:

$$\begin{aligned}
p(y|\mathbf{x}) =& \frac{\sum_{i=1}^{M} \omega_i \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{i\mathbf{x}^T}, \boldsymbol{\Sigma}_{i\mathbf{xx}}) \mathcal{N}(y; \boldsymbol{\mu}_{iy|\mathbf{x}^T} \boldsymbol{\Sigma}_{iy|\mathbf{x}})}{\sum_{j=1}^{M} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{j\mathbf{x}^T}, \boldsymbol{\Sigma}_{j\mathbf{xx}})} \\
=& \sum_{i=1}^{M} \beta_i \mathcal{N}(y; \boldsymbol{\mu}_{iy|\mathbf{x}^T}, \boldsymbol{\Sigma}_{iy|\mathbf{x}})\,,
\end{aligned} \tag{6.8}$$

where

$$\begin{aligned} \beta_i &= \frac{\omega_i \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{i\mathbf{x}^T}, \boldsymbol{\Sigma}_{i\mathbf{x}\mathbf{x}})}{\sum_{j=1}^{M} \omega_j \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{j\mathbf{x}^T}, \boldsymbol{\Sigma}_{j\mathbf{x}\mathbf{x}})}, \\ \boldsymbol{\mu}_{iy|\mathbf{x}^T} &= \mu_{iy} - \boldsymbol{\Sigma}_{iy\mathbf{x}} \boldsymbol{\Sigma}_{i\mathbf{x}\mathbf{x}}^{-1} (\boldsymbol{\mu}_{i\mathbf{x}^T} - \mathbf{x}) . \end{aligned} \tag{6.9}$$

Finally, by substituting Equation (6.8) into Equation (6.6), we have:

$$\begin{aligned} \hat{y} &= \sum_{i=1}^{M} \beta_i \int y \, \mathcal{N}(y; \boldsymbol{\mu}_{iy|\mathbf{x}^T}, \boldsymbol{\Sigma}_{iy|\mathbf{x}^T}) dy \\ &= \sum_{i=1}^{M} \beta_i \boldsymbol{\mu}_{iy|\mathbf{x}^T} . \end{aligned} \tag{6.10}$$

6.3.2 Optimal GMM Components Estimation

We use expectation maximization (EM) algorithm to estimate GMM parameters Θ. EM is an iterative method to find the maximum likelihood estimate (MLE) of the parameters. In order for EM to perform the two steps of expectation (E) and maximization (M), it needs to receive the number of GMM mixtures as an input.

There have been several studies and different methods to obtain the optimal number of mixtures and selecting the efficient model, rather than simply taking a random or educated guess. Bayesian information criterion (BIC) [128] is a criterion introduced for model selection and is penalized based on the model complexity. BIC maximizes the maximum likelihood function for each model. It is based on the increasing function of an error and the model with the lowest BIC, the more efficient in terms of predicting the demand.

6.4 Construction of Prediction Model

Figure 6.1 shows the required steps towards constructing the prediction model. Different steps involved in performing the prediction are discussed in the rest of this section.

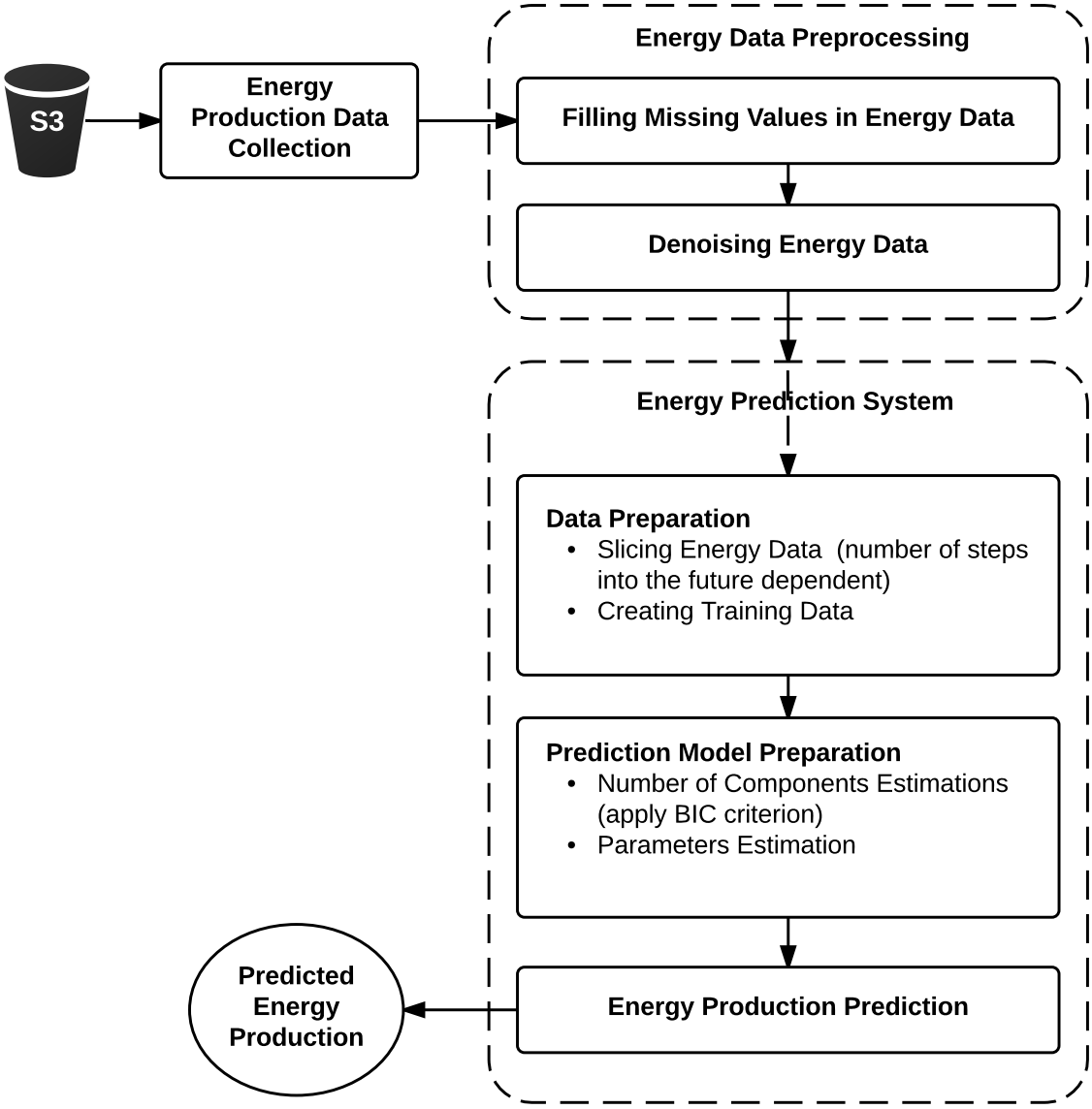


Figure 6.1: Renewable energy production prediction model.

6.4.1 Filling Missing Values in Renewable Energy History Data

Access to accurate history data is critical for prediction. Since having access to perfect history data is not always the case, often there are missing points in time regarding collected history data. Keeping the time-stamp related to each renewable energy data is important to feed into the prediction model. Filling the gaps by simply shifting the energy history data back in time changes the energy data-time mapping. Therefore, we need to fill up

the missing values in the collected energy data while keeping each renewable energy's time-stamp. For each collected solar and wind energy, if there are missing data points in the beginning or at the end of a time period, we replicate the first or last observed energy data, respectively. Otherwise, if there are missing energy data in the middle of the time series, we use linear interpolation between the first and the last observed energy data. As presented in Figure 6.1, filling missing values in the renewable energy history data is part of the preprocessing step, before performing the prediction.

6.4.2 Denoising the Renewable Energy Data

Before training the data and performing the prediction, we need to smooth the collected renewable energy data and remove the sharp acceleration and deceleration of the energy data to achieve a fair prediction. To smooth the history data, we use the fast fourier transform (FFT) algorithm [129] to remove the high frequencies in the energy data and reconstruct it again with only low frequency information.

6.4.3 Training History Data

As shown in Figure 6.1, we need to prepare the history data to feed into the prediction model. Training set will be constructed according to the following pattern.

$$\mathbf{Z} = \begin{bmatrix} x_1 & x_2 & x_3 & \dots & x_N & y_1 \\ x_2 & x_3 & x_4 & \dots & x_{N+1} & y_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ x_T & x_{T+1} & x_{T+2} & \dots & x_{T+N} & y_T \end{bmatrix}, \tag{6.11}$$

where, in our model, $y_i = x_{N+i}$ for $i \in [1, T]$.

To perform the renewable energy production prediction $\hat{y}_{T+1}$, we use the previously observed production values. The granularity of the data history should be equal to the length of the prediction being performed from the last observed renewable energy upto 1-step ahead in time. We denote the granularity of the data history as g, which should be equal to performing the prediction for 1-step ahead into the future (g = 1-step ahead

prediction length).

6.4.4 Feature Set Selection

Performing renewable energy prediction requires access to the history data and training the data to estimate prediction model parameters. As stated earlier, we use N previously observed states to predict the next energy production. GMM parameters estimation are driven from running $gmm.fit$ on the training set $\mathbf{Z}$ containing history data. The training set is constructed from multiple rows, each equal to a $\mathbf{z} = [\mathbf{x}, x_{t+1}]$ vector, where $\mathbf{x} = [x_{t-N+1}, ..x_{t-1}, x_t]$. The elements of $\mathbf{z}$ do not necessarily need to be consecutive observed values. Vector $\mathbf{z}$ elements selection have a major effect on the estimation of the prediction model parameters and accordingly the predicted value of the energy production.

6.5 Prediction Approach and Methodologies

As we mentioned earlier, training the data and filling the training matrix with the right feature set is important to lead us to an accurate prediction. Depending on the time-step ahead into the future that the prediction is taking place, we consider two different approaches to train the data history. To perform the energy prediction for the s^{th}-step ahead into the future, the following two approaches are considered:

- Short-term approach: Selecting every subsequent s^{th} item in the data history.
- Long-term approach: Selecting every subsequent hour:minute corresponding to the hour:minute of the s^{th}-step in the data history.

Moreover, in order to construct the training matrix we consider two different methodologies, as

- Direct multi-step ahead prediction: Direct multi-step ahead prediction (DMSA) performs the energy prediction for s-steps ahead into the future using only the history data. In this approach, the energy production prediction for $\hat{y}_{t+s}$ is independent of the prediction results for energy production before time $t + s$ and is made directly using the data available upto time t.

- Propagated multi-step ahead prediction: Propagated multi-step ahead (PMSA) prediction uses the predicted energy production as an input to the model for next energy production prediction. PMSA uses the $\hat{y}_{t+s-1}$ value as an input to predict the value of $\hat{y}_{t+s}$. The main aim of propagated prediction is to use the results of the previous successful predictions for the next predictions, since prediction results are more accurate for time-steps closer to the last observed energy data.

6.6 Prediction Model Evaluation

This section discusses the experiment setup and the validation of the prediction model. However, as it has been stated before, the accuracy of the prediction model has been tested using the workload demand collected from AWS biggest region, US East, Virginia. We used one month of data with granularity 15 minutes as history data to train the model and predict 7 days ahead into the future. However, due to the confidentiality of the used data set and also our goal to validate the model for renewable energy production, we run a separate set of experiments based on renewable energy production prediction.

6.6.1 Experiment Setup

Renewable Energy Traces

We use the renewable energy measurements from NREL [15] to calculate solar and wind energy production for a data center. The solar and wind energy traces used in this chapter are the same as the renewable energy used in Chapter 5. The measurements are with 1 minute granularity from May, 1st to May, 29th 2013. We use Global horizontal irradiance (GHI) measurements to calculate the output of the solar photovoltaic (PV). The GHI measurements are for PV flat panels on tilted surface at a 45-degree angle and PV efficiency of 30%. We calculate the solar output based on [124] and the total area for the flat plates is considered to be $100m^2$, derived from the configuration by Solarbayer [100].

To calculate wind energy production, we use the proposed model by Fripp et al. [119]. We feed the wind speed, air temperature, and air pressure, derived from NREL measure-

ments, to the model to calculate wind power at the data center, assuming the data center uses a GE 1.5MW wind turbine.

Benchmark Prediction Models

We compare the results of the prediction model against three different models. Naive that assumes prediction at each point in time is the same as the previously observed value, $\hat{y_{t+1}} = y_t$, linear regression [130] and random forest [131].

6.6.2 Prediction Analysis Metrics

We investigate the performance of the prediction model by studying the following quality metrics:

Bounded Predicted Values

We use bounded predicted values as a measure to quantify the percentage of the predicted values around x% of the actual values. This is a good measurement to know for different prediction models, what is the percentage of the predicted values bounded within an error margin (e.g., $\pm$20%).

R-Squared

In analyzing the accuracy of a prediction, a good prediction model would have the predicted versus actual values as close to the 45-degree line, as shown in Figure 6.3. R-squared is a statistical measure that shows how close the predicted values are to the actual values. R^2 gives an intuitive measure of the proportion of the predicted values that could be explained by the actual values. In other words, an R^2 with value x means that x% of the prediction variation is explained by the actual values.

R^2 value is between 0 and 100%. The higher the R-squared, the better the prediction fits the actual values. If a prediction model could explain 100% of the variance, the

predicted values would always equal the actual values and therefore, all the data points would fall on the 45-degree line.

Standard Error

Standard error (S), same as R^2, tells us how well the predicted and actual values would fall on the same line. Standard error is the average distance between the predicted and the actual values. The smaller the S the better the prediction and indicates that the predicted and actual values fall on the 45-degree line. Moreover, standard error is a good indication to show the accuracy of the prediction. A standard error with value s tells that approximately 95% of the predicted versus actual values fall within $\pm 2 \times s$ of the 45-degree line.

Mean Absolute Error (MAE)

Using a metric that measures the average magnitude of the errors is always useful and indicates how big of an error can be expected from the prediction on average. A perfect prediction would have a MAE zero. Since MAE is skewed in favor of large errors (prediction outliers), we need to use other metrics such as p^{th}-percentile to better validate the accuracy of the prediction model.

P-Percentile

P-percentiles are useful to know the distribution of the prediction error. A p^{th} percentile of a distribution shows that roughly p% of the error values are equal to or less and $(1 - p)$% of the error values are larger than that number. Percentiles range in $[0, 100]$. The 0^{th}-percentile shows the min and 100^{th}-percentile shows the max value in a distribution. We measure the p^{th} percentiles on the absolute values of the prediction error ($|\hat{y} - y|$). This way we focus on the unsigned errors and measure how close the prediction and actual values are together, without considering the direction of the error.

It should be noted that when reporting percentiles, we need to consider that if the data distribution is heavy-tailed (right-skewed), significant outliers could be hidden, even not

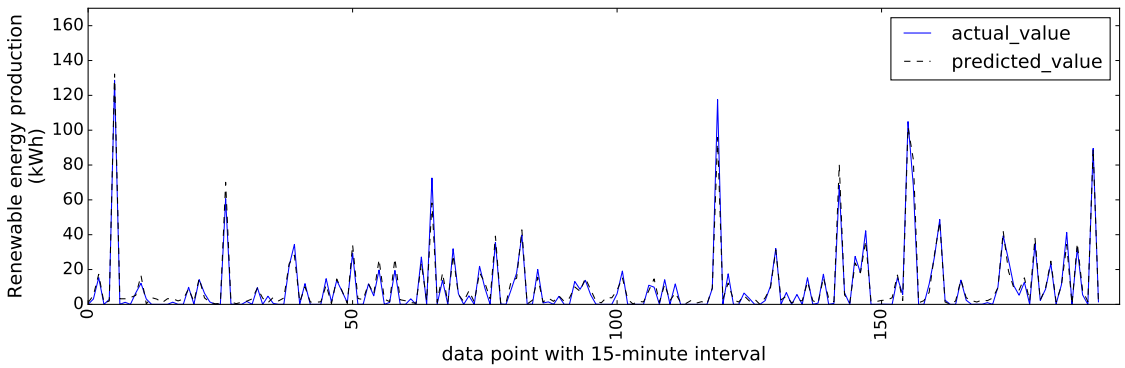


Figure 6.2: Results of prediction model for 8 days period of renewable energy production for 15-minute ahead prediction.

reflected in 90^{th} or 99^{th} percentiles. Therefore, we also report p-100 which shows the maximum error value in the prediction.

6.6.3 Prediction Results and Analysis

In the following, we validate the accuracy of the proposed prediction model using the renewable energy measurements from NREL [15]. From the collected renewable energy levels for May 2013, we consider the first three weeks as the data history to train the model and the last 8 days as test data to verify the prediction accuracy. We run the prediction model on the previously observed renewable energy production (the data history) to predict the renewable energy level for the next 15 minutes with the granularity of 1 minute. Then, we move the data history window 15-minute ahead to predict the next 15 minutes. We repeat this till we predict 8 days of renewable energy level.

Since the prediction window size is relatively small, 15 minutes, we use the short-term approach, discussed in Section 6.5, to fill the elements in the training matrix. Moreover, we use DMSA methodology, which is independent from the newly predicted values, for feature set selection and training the data. We defer applying long-term approach and PMSA methodology for the interested reader. However, in the carried experiments using AWS data to perform one-week ahead prediction, we applied short-term approach for predictions up to 36 hours ahead in time and long-term approach beyond that time.

Figure 6.2 shows the renewable energy production prediction against the actual val-

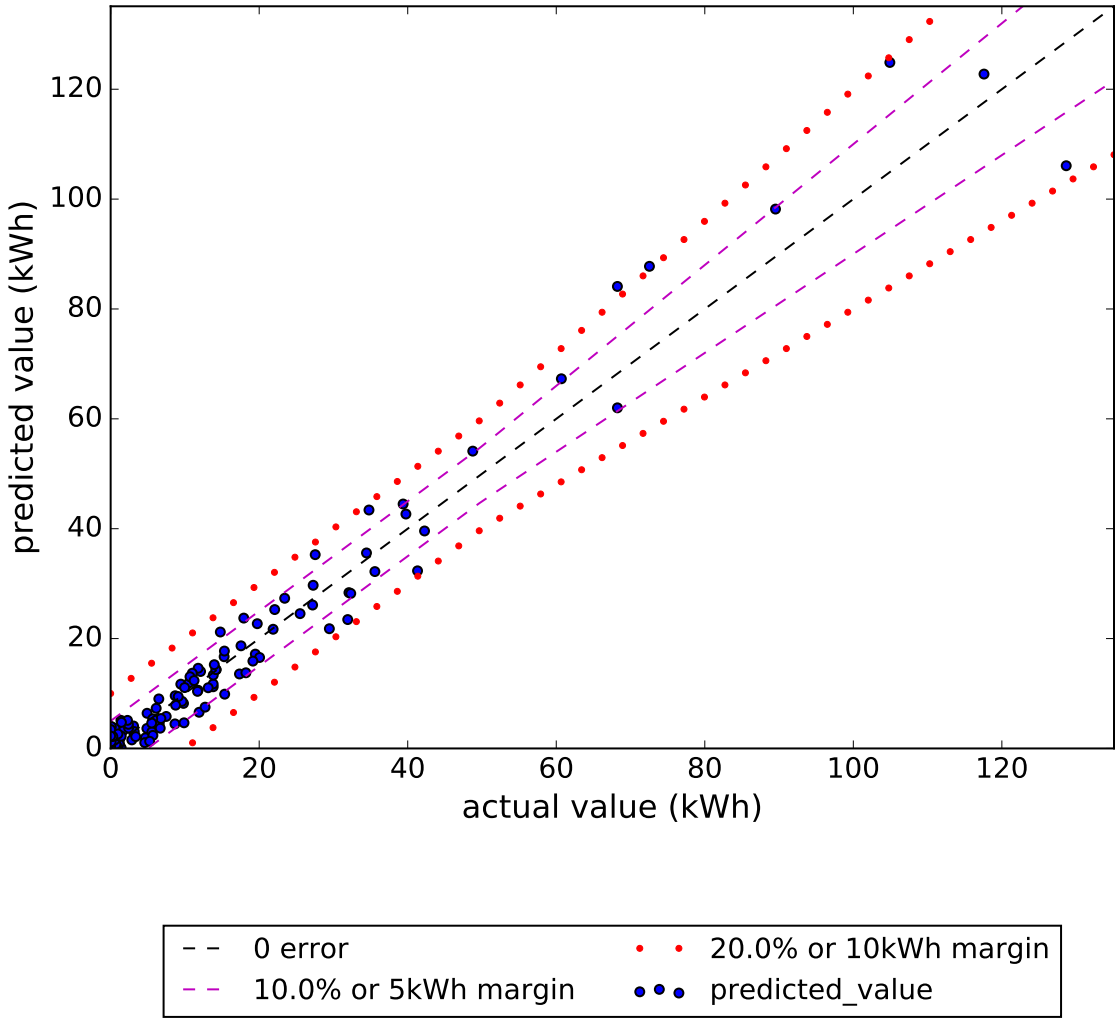


Figure 6.3: Predicted vs. actual values for 8 days period of renewable energy production for 15-minute ahead prediction with ±10% and ±20% around the actual value.

ues for 8 days. We also demonstrate the GMM prediction results using scatter plot, Figure 6.3. As it can be seen in this figure, we measure the percentage of the predicted values bounded within ±10% and ±20% relative error and each with considering an absolute error of 5kWh and 10kWh, respectively. Using an absolute error constraint on top of the relative error margins, prevents small errors to affect our decision making. This has been stated as bounded predicted values in Table 6.1.

We use the previously discussed prediction analysis metrics to evaluate the accuracy of the prediction. The results are presented in Table 6.1. The prediction model column

states GMM model and other benchmark models used in our analysis. The results show that almost all the predicted values in GMM ($\approx$ 100%), fall within 20% of observed actual values, whilst linear regression, random forest and naive model all are lower than GMM. Even checking bounded predictions bounded within $\pm$10% of the actual values is still close to perfect prediction (97.39%). This means GMM can predict renewable energy with considerably high precision almost similar to real time measuring.

In the rest of the reported metrics, R^2, S, MAE, P-90, P-99, and P-100, GMM is performing better than the rest of the models. Having R^2 of 97% shows that almost all the predicted values are aligned and could be explained by the actual values. Moreover, as per the measured MAE, the prediction error on average is 2.42 kWh, which is a negligible value.

Table 6.1: Prediction accuracy under different quality metrics.

Prediction Model	*Bounded Predictions*	R^2	S	MAE	P-90	P-99	P-100
GMM	99.48%	97%	0.18	2.42	4.16	4.39	21.76
Linear regression	89.81%	86.34%	0.21	3.97	6.78	8.01	25.72
Random forest	81.27%	77.71%	0.43	5.45	9.63	11.84	31.65
Naive	47.41%	0.01%	1.7	16.04	35.56	118.34	128.67

6.7 Summary

In this chapter, we presented a short-term renewable energy production prediction to predict the renewable energy level for many time-steps ahead into the future. The proposed model is based on the Gaussian mixture model and uses history data to train itself and predict the next level of renewable energy in a data center. Knowing the future level of renewable energy helps the cloud provider to make an informed decision to migrate the VMs in the absence of the renewable energy in a data center to a data center with excess renewable energy. This way, the cloud provider can maximize the usage of renewable energy.

To validate the accuracy of the proposed model, we used renewable energy measurements by NREL. The prediction results show that GMM model can predict up to 15 minutes ahead into the future with nearly 98% precision around $\pm 10\%$ of the actual values. This means that cloud provider can perform future-aware online VM migrations with performance close to the optimal offline, that has the full knowledge of the future level of renewable energy.

Chapter 7
Conclusions and Future Directions

This chapter summarizes this thesis investigations on energy and carbon-efficient resource management in geographically distributed cloud data centers and highlights its main research outcomes. It also discusses future research directions to pursue in this field.

7.1 Summary of Contributions

CLOUD computing has enabled the long-held dream of delivering computing as a utility to users. It provides access to resources anytime and anywhere on a pay-as-you-go manner. Therefore, the number of individuals and organizations shifting their workload to cloud data centers are growing more than ever. The growing scale of cloud data centers makes them accumulate a large fraction of the world's computing resources that needs a huge amount of electricity to operate. Moreover, most of the popular cloud providers have data centers in diverse geographies, which provides the opportunity to have different energy sources. In this regard, having techniques that makes data centers energy and carbon-efficient and reduces their operational costs is crucial.

This thesis addresses the problem of energy and carbon-efficient resource management in geographically distributed cloud data centers. It focuses on the techniques for VM placement, investigates the parameters with largest effect on the energy and carbon cost, migration of VMs between data center sites to harvest renewable energy sources, and prediction of renewable energy to maximize its usage. In particular, Chapter 1 described the thesis motivation and objective in more details. It also highlighted its main contributions and presents the structure of the thesis.

In Chapter 2, we provided a comprehensive understanding of the existing body of

knowledge in the area of energy and carbon footprint-aware resource management in cloud data centers. Most of the techniques in green cloud resource management focus on a single server and a single data center. We provided the limitations they face, specially not being able to harvest renewable energy sources at different geographical locations. We also studied the works considering geographically distributed cloud data centers, their target goal and technique, and whether or not they consider carbon footprint aspect while being energy efficient.

Chapter 3 presented a VM placement algorithm to reduce energy consumption and carbon footprint in a cloud computing environment. We used energy and carbon-efficient cloud information service (ECE-CIS) that obtains energy and carbon related information from data centers and enables the cloud broker to perform carbon and energy-efficient VM placement. We introduced energy and carbon-efficient (ECE) VM placement algorithm that considers data centers' carbon footprint rate and PUE in decision making. We compared the ECE VM placement algorithm with five competing algorithms. The experiment results show that ECE has a better performance in terms of carbon footprint and power consumption in comparison to the competitive algorithms, while it keeps the same level of SLA.

Chapter 4 investigated different parameters that affect energy and carbon cost for a cloud provider. We considered carbon cost as part of the total cost. This enabled the cloud provider not only decrease the total cost but also reduce the CO_2 emission. We also considered overhead energy consumption in support of the IT devices as part of the data centers' total energy cost. For this, we employed PUE as a metric that affects overhead energy of a data center, which is responsible for almost half of the energy consumption. We exploited a model for PUE as a function of data center's IT load and outside temperature. Further, we considered data centers have on-site renewable energy sources. We presented efficient two-stage VM placement approaches that respond to dynamic PUEs.

We carried out extensive simulations of the proposed dynamic VM placement algorithm and six variations that neglect different components of the cost, and studied the effect of different parameters and combinations of them on the amount of green and

brown energy usage, carbon footprint, and total energy and carbon cost of the cloud data centers. The results showed that ERA-DP that considers dynamic PUE, availability of renewables, and changes in energy consumption has the highest effect in reducing the total cost of energy and carbon and also reducing brown energy usage; whilst has the same level of SLA compared to the other algorithms.

In Chapter 5, we explored how much energy cost savings can be made knowing the future level of renewable energy in the data center sites. We took advantage of migrating VMs to the data centers with excess renewable energy. We proposed two online deterministic algorithms, one with no future knowledge called deterministic and one with limited knowledge of the future renewable availability called future-aware. We studied the algorithms performance against the optimal offline algorithm with full knowledge of the future level of renewable energy. We evaluated the proposed algorithms through extensive simulations using real-world traces for renewable energy (solar and wind) and one-month workload trace of a Google cluster usage. The offline algorithm with full knowledge of renewable energy level performs the best in comparison to the future-aware and optimal online algorithms. The optimal online algorithm incurs more cost compared to the offline algorithm when no future knowledge is available. Moreover, the future-aware algorithm's performance gets competitive with offline algorithm by the increase in its window-size regarding the knowledge of future renewable energy.

Finally, in Chapter 6, we proposed a prediction model that helps the future-aware online deterministic algorithm to make an informed decision and migrate the VMs between data center sites in the absence of the renewable energy. The proposed model is based on the Gaussian mixture model and uses history data to train itself and predict the next level of renewable energy in a data center. Therefore, the cloud provider can maximize the usage of renewable energy. We validated the model accuracy using real-world traces of meteorological data from NREL. Moreover, we evaluated the proposed prediction model accuracy in a separate analysis using AWS biggest region workload data.

7.2 Future Research Directions

Despite the contributions of the current thesis in energy and carbon-efficient resource management in geographically distributed data centers, there are a number of open research challenges that need to be addressed in order to further advance the area.

7.2.1 VM Migration over Transmission Network

VM migration across data center sites to harvest the renewable energy sources is still at its early stages. It is important to study the effect of minimizing brown energy usage and carbon cost versus network cost and delay imposed due the data transfer over the network. Moreover, as a future direction, one can study the effect of inter-region migrating of VMs to evaluate the improvements in energy cost versus network delay.

7.2.2 VM Type Selection for Migration

Selecting the VMs to migrate depending on the application running on top of the VM with respect to users' service level agreement is also another area of future study. This is important, since there are situations that VM migration could lead to service level agreement violation of some users with special requirements or VM migration needs large amount of data transfer over the network because of data unavailability in the destination.

7.2.3 Effect of Multiple VM Migration

As per the carried study in Chapter 5, an important topic of future research is considering a more complex problem, which involves the migration of multiple VMs. In a scenario that considers migration of multiple VMs at the same time, one can study the effect of sharing the network on the transfer time, and evaluating the competitiveness of the possible online algorithms in comparison to the optimal offline algorithm.

7.2.4 Placement Algorithms Based on VM Holding Time

Considering VM holding time at the time of VM placement and selecting the destination data center and server is also an interesting area for further investigation. If a server hosts VMs with different holding times, then VMs termination in different times could lead to resource wastage, and consequently high energy consumption and carbon footprint. Therefore, studying the impact of VM holding times and the hosted data center and server on the total energy and carbon footprint could help the cloud provider make informed decision at the time of initial VM placement.

7.2.5 Renewable Energy Storage

There are studies that consider storing excess renewable energy in batteries to use at times of the day that renewable sources are not available. Since main cloud providers started to build their own on-site renewable energy sources and having large scale renewable energy power plants, studying the cost-effectiveness of storing the renewable energy for future usage and contributing to the electrical grid is an important area for future study.

7.2.6 Interaction with Newly Emerged Paradigms

Recently, there have been newly emerged paradigms, such as fog or edge computing that extends the cloud computing to the edge of the network. Fog computing's main dedication is to bring low latency services to the users. Research in this area is still at its early stages. It would be interesting for one to study the impact of resource management techniques with the goal of reducing energy and carbon footprint, while still bringing low latency services to the users.

Appendix A

PUE Relation with IT Load and Temperature

A.1 PUE Relation with IT Load and Temperature

In order to find the relation between PUE, IT load, and outside temperature for a data center, we use the graphs defined by Rasmussen [37]. Firstly, we write the general form of PUE as:

$$\begin{aligned} PUE(U_t, H_t) &= 1 + \frac{N_I + U_t N_\triangle(H_t)}{P_I + U_t P_\triangle} \\ &= 1 + \frac{\frac{N_I}{P_\triangle} + \frac{U_t N_\triangle(H_t)}{P_\triangle}}{\frac{P_I}{P_\triangle} + U_t} \end{aligned} \tag{A.1}$$

In (A.1), we consider:

$$\bar{N_I} = \frac{N_I}{P_\triangle},\ \bar{N_\triangle}(H_t) = \frac{N_\triangle(H_t)}{P_\triangle},\ \bar{P_I} = \frac{P_I}{P_\triangle} \tag{A.2}$$

Now for PUE, we have:

$$PUE(U_t, H_t) = 1 + \frac{\bar{N_I} + U_t \bar{N_\triangle}(H_t)}{\bar{P_I} + U_t} \tag{A.3}$$

Moreover, from the PUE and temperature graph in the work [37], we can write PUE at a constant utilization as:

$$PUE(H_t) = M + N.H_t \tag{A.4}$$

In order to find the relation of PUE based on the IT load and outside temperature, we use three points of the PUE and IT load graph in [37] to find the values for $\bar{N_I}$, $\bar{N_\triangle}(H_t)$, and $\bar{P_I}$. For this, firstly we write the general form for (A.3) as:

$$\begin{aligned} \bar{N_I} + U_t\bar{N_\triangle}(H_t) + \bar{P_I} + U_t &= PUE \times \bar{P_I} + PUE \times U_t \\ \bar{N_I} + U_t\bar{N_\triangle}(H_t) + (1 - PUE)\bar{P_I} &= (PUE - 1)U_t \end{aligned} \tag{A.5}$$

The three different equations based on three different points of PUE and IT load graph are:

$$\begin{aligned} \bar{N_I} + 0.1\bar{N_\triangle}(H_t) - 2.5\bar{P_I} &= 0.25 \\ \bar{N_I} + 0.5\bar{N_\triangle}(H_t) - 0.8\bar{P_I} &= 0.4 \\ \bar{N_I} + \bar{N_\triangle}(H_t) - 0.6\bar{P_I} &= 0.6 \end{aligned} \tag{A.6}$$

By using linear equation of three variables, we have:

$$\bar{N_I} = 0.1935,\ \bar{N_\triangle}(H_t) = 0.4026,\ \bar{P_I} = -0.0065 \tag{A.7}$$

We assume that the $\bar{N_\triangle}(H_t)$ value is at $H_t = 28$. PUE value at this temperature from PUE and temperature graph is 2.28. And from (A.3), we calculate the IT load (U_t) to see whether or not at this utilization in PUE and IT load graph, $PUE = 2.28$.

$$2.28 = 1 + \frac{0.1935 + 0.4026U_t}{U_t - 0.0065} \Rightarrow U_t = 0.23 \tag{A.8}$$

As we can see from PUE and IT load graph, at $U_t = 0.23$, PUE value equals 2.28. Therefore, we set data center-B utilization in PUE and temperature graph, $U_t = 0.23$. From this, we find the value of $\bar{N_\triangle}(H_t)$ in a different temperature. At $H_t = 12$ and $PUE = 2.1$,

the new $\bar{N_{\triangle}}(H_t)$ value would be:

$$\begin{aligned} 2.1 &= 1 + \frac{0.1935 + 0.23\bar{N_{\triangle}}(12)}{0.23 - 0.0065} \\ &\Rightarrow \bar{N_{\triangle}}(12) = 0.227 \end{aligned} \tag{A.9}$$

Now by having two points for $\bar{N_{\triangle}}(H_t) = A + BH_t$, we can calculate A and B values.

$$\begin{aligned} B &= \frac{\bar{N_{\triangle}}(H_2) - \bar{N_{\triangle}}(H_1)}{H_2 - H_1} = \frac{0.4026 - 0.227}{28 - 12} \\ &\Rightarrow B = 0.0109 \\ A &= \bar{N_{\triangle}}(H_1) - BH_1 = 0.227 - 0.0109 \times 12 \\ &\Rightarrow A = 0.0953 \end{aligned} \tag{A.10}$$

Therefore, we have:

$$\begin{aligned} PUE(U_t, H_t) &= 1 + \frac{\bar{N_I} + AU_t + BU_tH_t}{\bar{P_I} + U_t} \\ &= 1 + \frac{0.1935 + 0.00953U_t + 0.0109U_tH_t}{U_t - 0.0065} \end{aligned} \tag{A.11}$$

We consider that servers are perfectly power proportional, that means servers do not consume any power at the idle state. Therefore, we can set $\bar{P_I} = 0$. Moreover, by rounding the constant values, the final model for PUE is going by:

$$PUE(U_t, H_t) \simeq 1 + \frac{0.2 + 0.01U_t + 0.01U_tH_t}{U_t} \tag{A.12}$$

Bibliography

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation computer systems*, vol. 25, no. 6, pp. 599–616, 2009.

[2] P. Mell and T. Grance, "The nist definition of cloud computing," *National Institute of Standards and Technology*, vol. 53, no. 6, p. 50, 2009.

[3] "Google Apps," (accessed on 18/04/2017). [Online]. Available: https://apps.google.com/

[4] "Salesforce," (accessed on 18/04/2017). [Online]. Available: https://www.salesforce.com/

[5] "Google App Engine," (accessed on 18/04/2017). [Online]. Available: https://appengine.google.com/

[6] "Microsoft Azure," (accessed on 18/04/2017). [Online]. Available: https://azure.microsoft.com/

[7] C. Vecchiola, X. Chu, and R. Buyya, "Aneka: a software platform for .NET-based cloud computing," *High Speed and Large Scale Scientific Computing*, vol. 18, pp. 267–295, 2009.

[8] J. Varia, "Best practices in architecting cloud applications in the aws cloud," *Cloud Computing: Principles and Paradigms*, pp. 457–490, 2011.

[9] "Google Cloud," (accessed on 18/04/2017). [Online]. Available: https://cloud.google.com/

[10] "Rackspace," (accessed on 18/04/2017). [Online]. Available: https://www.rackspace.com/

[11] E. Heyd, "America's Data Centers Consuming Massive and Growing Amounts of Electricity," 2014, (accessed on 05/03/2017). [Online]. Available: https://www.nrdc.org/media/2014/140826

[12] J. W. Smith and I. Sommerville, "Green cloud: A literature review of energy-aware computing," *Dependable Systems Engineering GroupSchool of Computer Science, University of St Andrews, UK*, 2010.

[13] U. Lublin and D. Feitelson, "The workload on parallel supercomputers: modeling the characteristics of rigid jobs," *Journal of Parallel and Distributed Computing*, vol. 63, no. 11, pp. 1105–1122, 2003.

[14] C. Reiss, J. Wilkes, and J. L. Hellerstein, "Google cluster-usage traces: format+ schema," *Google Inc., White Paper*, 2011.

[15] "Measurement and Instrumentation Data Center (MIDC)," (accessed on 25/05/2015). [Online]. Available: http://www.nrel.gov/midc/

[16] "US Department of Energy, Appendix F, Electricity Emission Factors," (accessed on 22/01/2013). [Online]. Available: http://cloud.agroclimate.org/tools/deprecated/carbonFootprint/references/Electricity_emission_factor.pdf

[17] "EIA-electricity data," (accessed on 30/01/2015). [Online]. Available: http://www.eia.gov/electricity/monthly/pdf/epm.pdf

[18] J. G. Koomey, "Worldwide electricity used in data centers," *Environmental Research Letters*, vol. 3, no. 3, 2008.

[19] P. Baer, "Exploring the 2020 global emissions mitigation gap," *Analysis for the Global Climate Network, Stanford University, Woods Institute for the Environment*, 2008.

[20] m. G. mypvdataKoomey, "Estimating total power consumption by servers in the us and the world," 2007.

[21] T. Brey and L. Lamers, "Using virtualization to improve data center efficiency," *The Green Grid, Whitepaper*, vol. 19, 2009.

[22] S. Srikantaiah, A. Kansal, and F. Zhao, "Energy aware consolidation for cloud computing," in *Proceedings of the Conference on Power Aware Computing and Systems*. San Diego, CA, USA: USENIX Association, 2008, pp. 10–10.

[23] A. Beloglazov, R. Buyya, Y. C. Lee, and A. Zomaya, "A taxonomy and survey of energy-efficient data centers and cloud computing systems," *Advances in Computers*, vol. 82, no. 2, pp. 47–111, 2011.

[24] E. Harney, S. Goasguen, J. Martin, M. Murphy, and M. Westall, "The efficacy of live virtual machine migrations over the internet," in *Proceedings of the 2nd international workshop on Virtualization technology in distributed computing*. Reno, Nevada: ACM, 2007, pp. 1–7.

[25] L. A. Barroso and U. Hölzle, "The case for energy-proportional computing," *IEEE Computer*, vol. 40, no. 12, pp. 33–37, 2007.

[26] M. Lin, A. Wierman, L. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," in *Proceedings of the IEEE INFOCOM*. IEEE, 2011, pp. 1098–1106.

[27] L. Lefèvre and A. Orgerie, "Designing and evaluating an energy efficient cloud," *The Journal of Supercomputing*, vol. 51, no. 3, pp. 352–373, 2010.

[28] B. Aksanli, J. Venkatesh, L. Zhang, and T. Rosing, "Utilizing green energy prediction to schedule mixed batch and service jobs in data centers," in *Proceedings of the 4th Workshop on Power-Aware Computing and Systems*. ACM, 2011, pp. 5:1–5:5.

[29] "MYPVDATA Energy Recommerce," https://www.mypvdata.com/.

[30] "National renewable energy laboratory (NREL)," http://www.nrel.gov/.

[31] Í. Goiri, R. Beauchea, K. Le, T. Nguyen, M. Haque, J. Guitart, J. Torres, and R. Bianchini, "Greenslot: scheduling energy consumption in green datacenters," in *Pro-*

ceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis. ACM, 2011, pp. 20:1–20:11.

[32] A. Mu'alem and D. Feitelson, "Utilization, predictability, workloads, and user runtime estimates in scheduling the ibm sp2 with backfilling," *IEEE Transactions on Parallel and Distributed systems*, vol. 12, no. 6, pp. 529–543, 2001.

[33] H. Goudarzi and M. Pedram, "Energy-efficient virtual machine replication and placement in a cloud computing system," in *Proceedings of the 5th International Conference on Cloud Computing (CLOUD)*. IEEE, 2012, pp. 750–757.

[34] J. Xu and J. A. Fortes, "Multi-objective virtual machine placement in virtualized data center environments," in *Proceedings of the Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on & Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*. IEEE, 2010, pp. 179–188.

[35] M. E. Haque, K. Le, Í. Goiri, R. Bianchini, and T. D. Nguyen, "Providing green slas in high performance computing clouds," in *Proceedings of the International Green Computing Conference (IGCC)*. IEEE, 2013, pp. 1–11.

[36] Í. Goiri, K. Le, T. D. Nguyen, J. Guitart, J. Torres, and R. Bianchini, "Greenhadoop: leveraging green energy in data-processing frameworks," in *Proceedings of the 7th ACM european conference on Computer Systems*. ACM, 2012, pp. 57–70.

[37] N. Rasmussen, "Electrical efficiency measurement for data centers," *Schneider, Whitepaper*, vol. 154, 2007.

[38] Í. Goiri, K. Le, J. Guitart, J. Torres, and R. Bianchini, "Intelligent placement of datacenters for internet services," in *Proceedings of the 31st International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2011, pp. 131–142.

[39] K.-y. Chen, Y. Xu, K. Xi, and H. J. Chao, "Intelligent virtual machine placement for cost efficiency in geo-distributed cloud systems," in *Proceedings of the International Conference on Communications (ICC)*. IEEE, 2013, pp. 3498–3503.

[40] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for internet-scale systems," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, pp. 123–134, 2009.

[41] S. Akoush, R. Sohan, A. C. Rice, A. W. Moore, and A. Hopper, "Free lunch: Exploiting renewable energy for computing." in *HotOS*, vol. 13, 2011, pp. 17–17.

[42] K. Le, R. Bianchini, M. Martonosi, and T. Nguyen, "Cost-and energy-aware load distribution across data centers," *Proceedings of the Workshop on Power-Aware Computing and Systems (HotPower)*, pp. 1–5, 2009.

[43] Y. Zhang, Y. Wang, and X. Wang, "Greenware: Greening cloud-scale data centers to maximize the use of renewable energy," in *Proceedings of the ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*. Springer, 2011, pp. 143–164.

[44] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew, "Geographical load balancing with renewables," *ACM SIGMETRICS Performance Evaluation Review*, vol. 39, no. 3, pp. 62–66, 2011.

[45] M. Lin, Z. Liu, A. Wierman, and L. L. Andrew, "Online algorithms for geographical load balancing," in *Proceedings of the International Green Computing Conference (IGCC)*. San Jose, CA, USA: IEEE, 2012, pp. 1–10.

[46] W. Kwon and A. Pearson, "A modified quadratic cost problem and feedback stabilization of a linear system," *IEEE Transactions on Automatic Control*, vol. 22, no. 5, pp. 838–842, 1977.

[47] S. K. Garg, C. S. Yeo, A. Anandasivam, and R. Buyya, "Environment-conscious scheduling of hpc applications on distributed cloud-oriented data centers," *Journal of Parallel and Distributed Computing*, vol. 71, no. 6, pp. 732–749, 2011.

[48] S. K. Garg, C. S. Yeo, and R. Buyya, "Green cloud framework for improving carbon efficiency of clouds," *Proceedings of the 17th International Conference on Parallel Processing*, pp. 491–502, 2011.

[49] C. Chen, B. He, and X. Tang, "Green-aware workload scheduling in geographically distributed data centers," in *Proceedings of the IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. Taipei, Taiwan: IEEE, 2012, pp. 82–89.

[50] A. Celesti, A. Puliafito, F. Tusa, and M. Villari, "Energy sustainability in cooperating clouds," in *Proceedings of the 3rd International Conference on Cloud Computing and Services Science (CLOSER)*, Aachen, Germany, 2013, pp. 83–89.

[51] H. Xu, C. Feng, and B. Li, "Temperature aware workload management in geo-distributed datacenters," in *Proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*. ACM, 2013, pp. 373–374.

[52] S. Pelley, D. Meisner, T. F. Wenisch, and J. W. VanGilder, "Understanding and abstracting total data center power," in *Proceedings of the Workshop on Energy-Efficient Design*, Austin, Texas, USA, 2009.

[53] R. Zhou, Z. Wang, A. McReynolds, C. E. Bash, T. W. Christian, and R. Shih, "Optimization and control of cooling microgrids for data centers," in *Proceedings of the 13th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*. IEEE, 2012, pp. 338–343.

[54] S. Ren, Y. He, and F. Xu, "Provably-efficient job scheduling for energy and fairness in geographically distributed data centers," in *Proceedings of the 32nd International Conference on Distributed Computing Systems (ICDCS)*. Macau, China: IEEE, 2012, pp. 22–31.

[55] K. Le, R. Bianchini, T. D. Nguyen, O. Bilgir, and M. Martonosi, "Capping the brown energy consumption of internet services at low cost," in *Proceedings of the International Green Computing Conference*. Chicago, IL, USA: IEEE, 2010, pp. 3–14.

[56] G. E. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis: forecasting and control*. Wiley. com, 2013.

[57] S. Brooks and B. Morgan, "Optimization using simulated annealing," *The Statistician*, pp. 241–257, 1995.

[58] K. Le, R. Bianchini, J. Zhang, Y. Jaluria, J. Meng, and T. D. Nguyen, "Reducing electricity cost through virtual machine placement in high performance computing clouds," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. ACM, 2011, pp. 22:1–22:12.

[59] "Dror feitelson. parallel workloads archive," (accessed on 12/03/2015). [Online]. Available: http://www.cs.huji.ac.il/labs/parallel/workload/

[60] N. Buchbinder, N. Jain, and I. Menache, "Online job-migration for reducing the electricity bill in the cloud," in *Proceedings of the the International Conference on Research in Networking*. Valencia, Spain: Springer, 2011, pp. 172–185.

[61] J. Luo, L. Rao, and X. Liu, "Spatio-temporal load balancing for energy cost optimization in distributed internet data centers," *IEEE Transactions on Cloud Computing*, vol. 3, no. 3, pp. 387–397, 2015.

[62] J. L. Berral, Í. Goiri, T. D. Nguyen, R. Gavalda, J. Torres, and R. Bianchini, "Building green cloud services at low cost," in *Proceedings of the 34th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2014, pp. 449–460.

[63] R. Brown *et al.*, "Report to congress on server and data center energy efficiency," in *Lawrence Berkeley National Laboratory*. Public law, 2008, pp. 109–431.

[64] C. Pettey, "Gartner estimates ict industry accounts for 2 percent of global co2 emissions," 2007.

[65] C. Stewart and K. Shen, "Some joules are more precious than others: Managing renewable energy in the datacenter," in *Proceedings of the Workshop on Power Aware Computing and Systems*, 2009.

[66] J. Haas, J. Froedge, J. Pflueger, and D. Azevedo, "Usage and public reporting guidelines for the green grids infrastructure metrics (pue/dcie)," 2009.

[67] C. Lien, Y. Bai, and M. Lin, "Estimation by software for the power consumption of streaming-media servers," *IEEE Transactions on Instrumentation and Measurement*, vol. 56, no. 5, pp. 1859–1870, 2007.

[68] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Dang, and K. Pentikousis, "Energy-efficient cloud computing," *The Computer Journal*, vol. 53, no. 7, pp. 1045–1051, 2010.

[69] C. Clark, K. Fraser, S. Hand, J. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*, vol. 2. USENIX Association, 2005, pp. 273–286.

[70] "Amazon Web Services," (accessed on 11/02/2015). [Online]. Available: http://aws.amazon.com/

[71] P. Van Mieghem, *Performance analysis of communications networks and systems*. Cambridge University Press, 2006.

[72] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, "Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Software: Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011.

[73] S. Greenberg, E. Mills, B. Tschudi, P. Rumsey, and B. Myatt, "Best practices for data centers: Lessons learned from benchmarking 22 data centers," in *Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings*, Asilomar, CA, USA, 2006, vol. 87, pp. 76–87.

[74] K. Mills, J. Filliben, and C. Dabrowski, "Comparing vm-placement algorithms for on-demand clouds," in *Proceedings of the 3rd International Conference on Cloud Computing Technology and Science*. IEEE, 2011, pp. 91–98.

[75] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[76] J. Hamilton, "Overall data center costs," *Perspectives at http://perspectives. mvdirona. com*, 2010.

[77] C. J. Hepburn, "Regulating by prices, quantities or both: an update and an overview," *Oxford Review of Economic Policy*, vol. 22, no. 2, pp. 226–247, 2006.

[78] C. Belady, A. Rawson, J. Pfleuger, and T. Cader, "Green grid data center power efficiency metrics: Pue and dcie," Technical report, Green Grid, Tech. Rep., 2008.

[79] "Latest Microsoft Data Center Design Gets Close to Unity PUE," (accessed on 07/04/2017). [Online]. Available: http://www.datacenterknowledge.com/archives/2016/09/27/latest-microsoft-data-center-design-gets-close-to-unity-pue/

[80] A. Shehabi, S. Smith, N. Horner, I. Azevedo, R. Brown, J. Koomey, E. Masanet, D. Sartor, M. Herrlin, and W. Lintner, "United states data center energy usage report," *Lawrence Berkeley National Laboratory, Berkeley, California. LBNL-1005775 Page*, vol. 4, 2016.

[81] "APAC data center survey reveals high PUE figures across the region," (accessed on 07/04/2017-04-07). [Online]. Available: http://www.datacenterdynamics.com/news/apac-data-center-survey-reveals-high-pue-figures-across-the-region/75116.fullarticle

[82] C.-M. Wu, R.-S. Chang, and H.-Y. Chan, "A green energy-efficient scheduling algorithm using the dvfs technique for cloud datacenters," *Future Generation Computer Systems*, vol. 37, pp. 141–147, 2014.

[83] D. Shen, J. Luo, F. Dong, X. Fei, W. Wang, G. Jin, and W. Li, "Stochastic modeling of dynamic right-sizing for energy-efficiency in cloud data centers," *Future Generation Computer Systems*, 2014.

[84] M. Giacobbe, A. Celesti, M. Fazio, M. Villari, and A. Puliafito, "An approach to reduce energy costs through virtual machine migrations in cloud federation," in *Proceedings of the Symposium on Computers and Communication (ISCC)*. IEEE, 2015, pp. 782–787.

[85] ——, "Evaluating a cloud federation ecosystem to reduce carbon footprint by moving computational resources," in *Proceedings of the Symposium on Computers and Communication (ISCC)*. IEEE, 2015, pp. 99–104.

[86] M. Blackburn and G. Grid, *Five ways to reduce data center server power consumption*. The Green Grid, 2008.

[87] "Standard Performance Evaluation Corporation. SPECpower." (accessed on 30/06/2014). [Online]. Available: http://www.spec.org/power_ssj2008

[88] J. Peterpaul, "Solar panel module and support therefor," Jan. 13 1987, uS Patent 4,636,577.

[89] P. Thibodeau, "Wind power data center project planned in urban area," *ComputerWorld, Apr*, 2008.

[90] "NREL: News - NREL releases renewable energy data book detailing growing industry," (accessed on 07/07/2014). [Online]. Available: http://www.nrel.gov/news/press/2013/5302.html

[91] "Facebook Installs Solar Panels at New Data Center," (accessed on 30/06/2015). [Online]. Available: http://www.datacenterknowledge.com/archives/2011/04/16/facebook-installs-solar-panels-at-new-data-center/

[92] "Apple Plans 20MW of Solar Power for iDataCenter," (accessed on 08/08/2014). [Online]. Available: http://www.datacenterknowledge.com/archives/2012/02/20/apple-plans-20mw-of-solar-power-for-idatacenter/

[93] "Wind-Powered Data Center in Wyoming," (accessed on 30/05/2015). [Online]. Available: http://www.datacenterknowledge.com/archives/2007/11/29/wind-powered-data-center-in-wyoming/

[94] A. Krioukov, C. Goebel, S. Alspaugh, Y. Chen, D. E. Culler, and R. H. Katz, "Integrating renewable energy using data analytics systems: Challenges and opportunities." *IEEE Data Eng. Bull.*, vol. 34, no. 1, pp. 3–11, 2011.

[95] B. Aksanli, J. Venkatesh, L. Zhang, and T. Rosing, "Utilizing green energy prediction to schedule mixed batch and service jobs in data centers," *ACM SIGOPS Operating Systems Review*, vol. 45, no. 3, pp. 53–57, 2012.

[96] D. Gmach, J. Rolia, C. Bash, Y. Chen, T. Christian, A. Shah, R. Sharma, and Z. Wang, "Capacity planning and power management to exploit sustainable energy," in *Proceedings of the International Conference on Network and Service Management (CNSM)*. IEEE, 2010, pp. 96–103.

[97] P. Hoeller and M. Wallin, *Energy prices, taxes and carbon dioxide emissions*. OECD Paris, 1991.

[98] Data center map. (accessed on 20/05/2015). [Online]. Available: http://www.datacentermap.com/

[99] "Solar Radiation Data Manual for Flat-Plate and Concentrating Collectors," (accessed on 30/01/2015). [Online]. Available: http://rredc.nrel.gov/solar/pubs/redbook/

[100] "Solarbayer," (accessed on 25/05/2015). [Online]. Available: http://www.solarbayer.com/

[101] "Carbon Tax Center, Pricing carbon efficiently and equitably," (accessed on 30/01/2015). [Online]. Available: http://www.carbontax.org/

[102] Travel weather averages (weatherbase). (accessed on 30/01/2014). [Online]. Available: http://www.weatherbase.com/

[103] J. Hamilton, "Cooperative expendable micro-slice servers (cems): low cost, low power servers for internet-scale services," in *Proceedings of the Conference on Innovative Data Systems Research (CIDR)*, Asilomar, CA, USA, 2009.

[104] J. Mankoff, R. Kravets, and E. Blevis, "Some computer science issues in creating a sustainable world," *IEEE Computer*, vol. 41, no. 8, pp. 102–105, 2008.

[105] "Facebook Installs Solar Panels at New Data Center," (accessed on 13/03/2017). [Online]. Available: http://www.datacenterknowledge.com/archives/2011/04/16/facebook-installs-solar-panels-at-new-data-center/

[106] "Facebook in Fort Worth: Our newest data center," (accessed on 13/03/2017). [Online]. Available: https://code.facebook.com/posts/1014459531921764/facebook-in-fort-worth-our-newest-data-center/

[107] "AWS and Sustainable Energy," (accessed on 13/03/2017). [Online]. Available: http://aws.amazon.com/about-aws/sustainable-energy/

[108] "Apple and the Environment," (accessed on 09/07/2015). [Online]. Available: http://www.apple.com/environment/

[109] "Renewable energy," (accessed on 13/03/2017). [Online]. Available: http://www.google.com/green/energy/

[110] "Microsoft To Use Solar Panels in New Data Center," (accessed on 13/03/2017). [Online]. Available: http://www.datacenterknowledge.com/archives/2008/09/24/microsoft-uses-solar-panels-in-new-data-center/

[111] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *Concurrency and Computation: Practice and Experience*, vol. 24, no. 13, pp. 1397–1420, 2012.

[112] F. Kong and X. Liu, "A survey on green-energy-aware power management for datacenters," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, pp. 30:1–30:38, 2014.

[113] A. N. Toosi and R. Buyya, "A fuzzy logic-based controller for cost and energy efficient load balancing in geo-distributed data centers," in *Proceedings of the 8th IEEE/ACM International Conference on Utility and Cloud Computing (UCC)*, Limassol, Cyprus, 2015.

[114] L. Rao, X. Liu, L. Xie, and W. Liu, "Minimizing electricity cost: optimization of distributed internet data centers in a multi-electricity-market environment," in *Proc. of INFOCOM*. San Diego, CA, USA: IEEE, 2010, pp. 1–9.

[115] K. Le, O. Bilgir, R. Bianchini, M. Martonosi, and T. D. Nguyen, "Managing the cost, energy consumption, and carbon footprint of internet services," *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 1, pp. 357–358, 2010.

[116] L. Minas and B. Ellison, *Energy efficiency for information technology: How to reduce power consumption in servers and data centers*. Intel Press, 2009.

[117] A. R. Karlin, M. S. Manasse, L. Rudolph, and D. D. Sleator, "Competitive snoopy caching," *Algorithmica*, vol. 3, no. 1-4, pp. 79–119, 1988.

[118] A. Borodin and R. El-Yaniv, *Online computation and competitive analysis*. Cambridge University Press, New York, 2005.

[119] M. Fripp and R. H. Wiser, "Effects of temporal wind patterns on the value of wind-generated electricity in california and the northwest," *IEEE Transactions on Power Systems*, vol. 23, no. 2, pp. 477–485, 2008.

[120] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 6, no. 4, p. 32, 2007.

[121] D. R. Cox, "Prediction by exponentially weighted moving averages and related methods," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 30, no. 2, pp. 414–422, 1961.

[122] A. N. Toosi, K. Vanmechelen, K. Ramamohanarao, and R. Buyya, "Revenue maximization with optimal capacity control in infrastructure as a service cloud markets," *IEEE Transactions on Cloud Computing*, vol. 3, no. 3, pp. 261–274, 2015.

[123] "HP ProLiant DL360 Generation9 (Gen9)," (accessed on 16/11/2015). [Online]. Available: http://h20195.www2.hp.com/v2/gethtml.aspx?docname=c04375623

[124] "Photovoltaic Education Network," (accessed on 25/05/2015). [Online]. Available: http://pveducation.org/

[125] NRDC and Anthesis, "Scaling Up Energy Efficiency Across the Data Center Industry: Evaluating Key Drivers and Barriers," *Natural Resources Defense Council*, 2014. [Online]. Available: https://www.nrdc.org/sites/default/files/data-center-efficiency-assessment-IP.pdf

[126] D. M. Titterington, A. F. Smith, and U. E. Makov, "Statistical analysis of finite mixture distributions," *Wiley series in probability and mathematical statistics. Applied probability and statistics*, 1985.

[127] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[128] G. Schwarz *et al.*, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[129] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex fourier series," *Mathematics of computation*, vol. 19, no. 90, pp. 297–301, 1965.

[130] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to linear regression analysis*. John Wiley & Sons, 2015.

[131] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001. [Online]. Available: https://doi.org/10.1023%2Fa%3A1010933404324