Exploiting Renewable Energy and UPS Systems with a Renewable-Aware Scheduler to Boost Energy Efficiency of Data Centers

by

Xiaopu Peng

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
August 6, 2022

Keywords: Energy Efficient Computing, Renewable Energy, Distributed UPS Systems,
Energy Efficient Data Centers, Data Centers Job Scheduling, Renewable-Aware Scheduling,

Approved by

Xiao Qin, Chair, Alumni Professor of Computer Science and Software Engineering
Kai-Hsiung Chang, Professor Emeritus, Computer Science and Software Engineering
Saad Biaz, Professor, Computer Science and Software Engineering
Yang Zhou, Assistant Professor, Computer Science and Software Engineering

Abstract

It is prudent to leverage on-site renewable sources like solar and wind to build environmentally friendly and energy-efficient data centers. Data centers deploy distributed UPS systems to handle the intermittent nature of renewable energy. As the pilot study of this dissertation research, we propose a renewable-energy manager, REDUX, to offer a smart way of managing server energy consumption powered by a distributed UPS system and renewable energy. REDUX maintains a desirable balance between renewable-energy utilization and data center performance. REDUX makes judicious use of UPS devices to allocate energy resources when renewable energy generation is low or fluctuate condition. REDUX not only guarantees the stable operation of daily workload but also curtails the energy cost of data centers by improving power resource utilization.

Due to intermittent nature of renewable energy and fluctuating grid energy price, advanced data centers often deploy distributed UPS systems with high efficiency, scalability, and reliability. In the second part of this dissertation study, we investigate the problem of energy management for data centers with renewable resources and energy storage. As an extension project of REDUX system, we devise an interface-backbone layer framework designed unified energy efficiency management system called REDUX2. Properly allocating fluctuating renewable energy, UPS battery energy storage, and grid power with dynamic price, REDUX2 is primed to minimize the long-term electricity bill for data centers. With prediction of renewable energy supply, categorization of grid power price level, and energy storage in the UPS devices, REDUX2 orchestrates workload distribution facilitated by the heuristic algorithms that handle high-level control strategies such as renewable energy smoothing, UPS device control, back-filling, and non-urgent job admission deferrals. Compared with

the existing solutions, REDUX2 demonstrates a prominent capability of mitigating average peak workload and boosting renewable-energy utilization.

Leveraging on-site renewable sources like solar and wind provides ample opportunities on developing environmentally friendly and energy-efficient data centers. Evidence demonstrates that renewable-aware job schedulers conserve energy by adjusting the arrangement of non-urgent workload according to renewable energy states. To further upgrade REDUX2 to the next high level, we bring forth a unify designed energy management system - REDUX3 - equipped with a renewable-aware scheduler. REDUX3 judiciously schedules jobs submitted by users and manages the energy supply of data centers powered by grid, renewable energy, and distributed uninterruptible power supply (UPS) systems. REDUX3 seamlessly incorporates a renewable-aware workload scheduler to make back-fills and defer admission s of non-urgent jobs, thereby properly allocating fluctuating renewable energy, UPS battery energy storage, and grid power with dynamic price. Our experimental results confirm that REDUX3 is adroit at minimizing long-term electricity bills for data centers by boosting renewable energy utilization.

Acknowledgments

It is my great honor here to present my gratitude to the invaluable guidance, experience sharing, constant support and encouragement from my advisor, people in our research group and family members during my study at department of Computer Science and Software Engineering at Auburn University.

I would like to, first of all, give my most sincere and deepest gratitude to my advisor, Dr. Xiao Qin, for his great efforts, trust and patience during my Ph.D. study at Auburn. As an once in a lifetime fortunate, I will never forget our first meetings back in summer 2015 when Dr. Qin encouraged my determination to come back to the U.S. and to restart my graduate studies as a computer science student. Since then, his extensive knowledge in the field of energy-efficient computing systems and inexhaustible enthusiasm for research keeps inspiring and driving me to accomplish my dissertation research. While working on my research papers to be published or reviewed, his insightful advice and suggestion helped and enlightened me in setting up motivations behind the research, building energy efficiency models, and developing a renewable-aware scheduler that can reduce energy costs in data centers.

I am also tremendously grateful to be advised by my committee members: Dr. Kai Chang, Dr. Saad Biaz, and Dr. Yang Zhou, who reviewed my proposal and dissertation documents. These professors gave me an array of valuable suggestions to bolster the overall quality of my dissertation. Importantly, I would also like to show my appreciation for Dr. Xiaowen Gong - my university reader who shepherded me through the dissertation writing.

Working with our research group is fantastic. I owe my gratitude to Dr. Yi Zhou, Tathagata Bhattacharya, Jianzhou Mao, Ting Cao, Taha Tekreeti, Dr. Chaowei Zhang and Dr. Christian Kauten, who helped me with paper writing, collecting data, and finalizing design ideas. In addition, all the professors and students in the Department of Computer Science and Software Engineering are greatly appreciated because the faculty and fellow students created an excellent atmosphere for study and research.

Finally and most importantly, the endless love from my family is the most powerful strength that keeps me fighting for my research. My mother Zhenqun Mao and my father Xiong Peng always stay with me, cheering for achievement and overcoming all difficulties.

This dissertation research is made possible by the support from the U.S. National Science Foundation under Grants IIS-1618669 and OAC-1642133.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Energy efficiency is one of the key design principles of modern data centers. Traditional power management techniques for data centers either overlook the usage of uninterruptible power supplies (i.e., UPS devices) or pay no attention to power-grid price fluctuation. In this dissertation, we first develop a resource manager called *REDUX* to cost-effectively allocate energy resources by incorporating a distributed UPS system tailored for renewable energy like solar and wind. REDUX system save electricity bills by judiciously managing both inexhaustible on-site renewable energy and inexpensive stored energy from power grid for a typical data center. The overarching goal of REDUX is to curtail energy cost while maintaining high performance in data centers to address two challenging issues: (1) the intermittent feature of renewable energy and (2) dynamically changing grid electricity price. REDUX is responsible for orchestrating daily computing workloads in data centers while cutting back energy costs through improved energy resource utilization.

Inspired by the observations of renewable energy utilization and distinct pattern of job power profiles, we hypothesize that a renewable-aware scheduling mechanism running prior to a dynamic power manager has a potential to enhance the energy efficiency of data centers. So we then anchor this dissertation in a renewable-energy-aware job scheduler to boost the energy-efficiency performance of REDUX, in which workloads are scheduled and dispatched to computing nodes in a data center. The rationale behind REDUX is sparked on a key observation of jobs from the perspective of users - each job has a distinct power consumption profile [68]. It is feasible to schedule jobs by considering expected power profiles accompanied by the status of renewable energy (available or outage) at the beginning of each time period. Our renewable-aware job scheduling mechanism enables REDUX to either (1) dispatch hefty load to the computing nodes with available cheap and green renewable energy or (2) defer non-urgent jobs when grid energy price is high amid insufficient energy stored in batteries.

1.1 Motivations

Motivations below strongly convince us to contrive the REDUX system.

- There are pressing demands for data centers fueled by cost-effective energy.
- Consuming renewable energy in data centers brings economical and environmental benefits.
- Distributed uninterruptible power supplies (UPS) systems are increasingly popular in data centers.
- It is promising to integrate together renewable energy supply, distributed UPS system and fluctuating grid power price as energy supply in an unified framework design.
- Renewable-aware workload scheduling prior to energy sources management is expected to shave energy cost.

1.1.1 Why Energy Saving

In the past decades, there has been a rapid growth of data centers built for a wide range of cloud-based computing services [6]. Because cloud computing platforms are reliable, elastic, and cost effective, cloud computing housed in data centers becomes essential data processing infrastructures for the operation of businesses, academic, and governmental institutions. Unfortunately, massive power consumption in these large-scale data centers has become a serious problem confronted by data-center designers and operators worldwide [19]. Regardless of developed and developing countries, data centers are still heavily lean on the electricity generated with high carbon-emission approaches referred to as brown energy. It is evident that data centers running around the world are at a risk of doubling their energy consumption every five years [60]. To make it worse, a significant portion of electricity is generated through carbon-intensive approaches (a.k.a., brown energy), and the environmental impacts caused by these data centers are under a continuous pressure from the media and society. Designing energy sustainable and environmental protected data centers are under a continuous desire from the society and industry.

1.1.2 Why Renewable Energy

In contrast to brown energy, renewable energy harvested from sources such as wind turbines and solar panels exhibits an assortment of benefits [50]. Tough conversion rate is still not high, most renewable energy sources produce few or even no global warming emissions while remaining a reasonably low and stable price that reduces energy costs after on-site installation [67] [12]. Considering dynamically changing grid power price rated as on-peak and off-peak, on the other hand, it would be profitable and reliable for the data centers to construct an on-site renewable facility in the long run. Other benefits of on-site renewable sources include (1) quick scaling up in power capacity to accommodate load growth and (2) shortening construction time interval compared with conventional power plants [45].

The integration of on-site renewable sources poses a grand challenge due to the intermittent nature of renewable energy to data-center operators [67]. In spite of this challenge, designing renewable-energy powered data centers has promising benefits beyond low carbon footprint. The first benefit is that compared with grid energy supply, renewable-energy price tends to remain reasonably low and stable after on-site installations are completed [12]. Running 24/7, data centers lead to gird energy bills with higher unit prices at on-peak period than off-peak period. Although non-urgent workloads may be deferred to off-peak time, it would be profitable and reliable for the data centers to construct its own on-site renewable facility in the long run. The second benefit is, renewable energy supplies, being highly modular, can incrementally increase power capacity to match load growth. Such a benefit greatly reduces the over-provisioning loss of a data center, because it takes a long time period for a server load to catch up with upgraded provisioning capacity. From the perspective of construction lead-time, on-site renewable energy facilities can be constructed in a shorter time interval compared with conventional power plants [45]. Unlike traditional energy, renewable-energy price tends to remain reasonably stable after on-site installations are completed [12].

As such, a growing number of data centers have kicked off various initiatives to integrate renewable sources into power supplies. And it is not a surprise that a growing number of data centers have already enjoyed their own on-site renewable sources, which are integrated into their power supplies. As an example, *Apple* has constructed a massive 100-acre solar farm adjacent to the *iCloud* data center in North Carolina. The Apple's solar farm annually yields 84 million kWh of clean renewable energy.

1.1.3 Why Distributed Uninterruptible Power Supply (UPS)

There are two widely adopted techniques to prevent costly down time incurred by power budget violations. The first approach is to intentionally over subscribe power infrastructures; the second one is called power capping (e.g., dynamic voltage and frequency scaling) [40]. More often than not, data centers powered by over-subscribed resources inevitably encounter a problem of power premium charge ascribed to the peak power. Such premium cost is a significant portion of electricity bills (e.g., up to 40%). It is common that the power capping approaches result in performance degradation. To prevent costly down time incurred by power budget violations, Govindan *et al.* employed a centralized UPS scheme to furnish energy during a peak demand period to prevent costly down time incurred by power budget violations [34]. Compared with the long lifespan and high reliability of modern battery systems, the restoring cost per energy unit from single battery is nearly negligible if the battery is healthily utilized by the energy source management systems. Though this UPS-based scheme effectively hides the extra power from power grids by shaving high-magnitude power spikes during a short time interval (e.g., 0-60 mins) using UPS devices.

When it comes to a long spike time window (e.g., 1-10 hour), distributed per-server UPS batteries helps to cutback electricity bill by the virtue of battery backups. In contrast to the conventional UPS design, distributed UPS batteries allow data-center operators to readily shave peak power with stored energy [40]. The advantages of distributed UPS systems include good efficiency, high scalability, and adequate reliability. Distributed UPS systems, where a potential single point of failure is eliminated, naturally scale up with the corresponding data-center sizes. A recent study showed that a hybrid distributed UPS at the power distribution unit (a.k.a., PDU) and server levels is a precursor architecture. As an example, a novel distributed battery control system was deployed in a Google state-of-the-art data center [2].

1.1.4 Why Variable Grid Power Price

The discussion of variation of grid (brown) energy price mainly falls into two categories: on-peak/off-peak [30] [32], and random pricing determined by the electricity market [21]. In on-peak/off-peak pricing, grid energy costs less when used during off-peak consumption times and more when consumed during on-peak times and the difference between on-peak and off-peak prices is largest in the summer time [30]. While Deng *et al.* [21] assumes that grid energy price is a random variable determined by pricing policies in the electricity market. While do not make any assumption about the stochastic pattern of grid energy price, they assume that duration of one time slot is small enough (e.g. 1 hour) so that grid energy price remains stable within each time slot. Grange *et al.* [35] concludes that taking the fluctuating grid energy price into account leads to a small increase of non-renewable energy use, but allows an important reduction of the grid energy cost. Through out this dissertation, we follow design of Deng *et al.* [21] with variable grid energy price and using real-world electricity power price.

1.1.5 Why Renewable Aware Scheduling

There have been practical and promising studies into the topic of job scheduling customized for energy-efficient and parallel computing systems [68] [64]. The prior evidence both demonstrates praiseworthy power-aware scheduling policies, which help in significantly shaving electricity cost by the virtue of window-based scheduling coupled with dynamic electricity pricing. Rather than allocating jobs one by one from a wait queue, these scheduling policies dispatch a "window" of jobs at a time. Jobs placed into the window are chosen to maintain fairness through job priorities; job allocations are accomplished in such a way to to minimize electricity bills. Complementing the aforementioned scheduling idea on large-scale parallel computing jobs, this work is anchored to a renewable-energy-aware scheduling design preparing for the next-step energy management in data centers. By integrating this scheduling mechanism into the REDUX system, we expect that REDUX has an edge over its predecessor REDUX system [52], [53] and [51] with respective to saving energy cost. Our novel mechanism not only schedules and dispatches jobs on the basis of job-power profiles and renewable-energy status, but also triggers negligible impacts on system utilization and fairness. Preferentially, the proposed mechanism dispatches intensive workload amid fluent and stable renewable energy, hold non-urgent jobs for the next time slot during the outage of renewable energy.

1.2 Contributions

In this dissertation, we propose a workload-energy management system - *REDUX* - to orchestrate renewable energy integrated with a power grid and a distributed UPS system for data centers. REDUX first examines jobs according to their power profiles with a help of renewable-aware scheduler to lay out a foundation for the subsequent executions in energy management; REDUX then adept at smoothing power supplies with the supply levels of renewable energy resources; By naturally scaling with data center size and eliminating a potential single point of failure, REDUX utilize per-server distributed batteries as an economical and secure battery backup; And finally, REDUX makes judicious decisions on UPS charging and discharging with vital information on renewable-energy supply levels and time-dependent grid power price.

One prominent feature of REDUX is that it orchestrates a desirable balance between renewable-energy utilization and data center performance. Importantly, REDUX conserves the energy cost of data centers through power resource management while offering reliable computing operations. Compared with existing strategies, REDUX demonstrates an exemplary capacity of mitigating average peak workload and boosting renewable-energy utilization.

The main contributions of this dissertation are summarized as follows:

- *Unified Framework Design* We design REDUX - an interface-backbone layered unified framework incorporating all the aspects of energy sources and workload. REDUX properly allocate fluctuating renewable energy, UPS battery energy storage or grid power with dynamic price for computing servers according to workload requirements in data centers, aiming to minimize the long-term electricity bill for the data center. REDUX also takes full advantages of pre-execution renewable-aware workload scheduling, which governs job scheduling window and manages energy sources for computing servers in data centers.

- *Heuristic Algorithms* We design heuristic algorithms act as renewable energy smoothing, UPS device control, and high level control strategies. With provided prediction of renewable energy supply, categorization of grid power price level and energy storage in the UPS devices, REDUX orchestrates workload distribution and make back-fills or defer decisions for the non-urgent jobs.

- *Renewable-aware Job Scheduling* We devise the renewable-aware job scheduling with a dynamic scaling scheduling window as a vital part of the workload management. The scheduling policy leverages low-cost renewable energy in accordance with power profiles, priorities, and renewable energy states, and back-fill workload to idle nodes or defers the executions of non-urgent workload to further cut back the overall electricity cost.

- *Optimization* We evaluate REDUX using real-world workload traces and green energy data. We demonstrate that REDUX paves the way for constructing modern data centers that are economically and environmentally friendly.

1.3 Organization

The rest of this dissertation is organized as follows. The next chapter presents prior studies and related research works. Chapter 2 survey and discuss related work with this dissertation from varies aspects. Chapter 3 present the pilot work from the REDUX project with detailed description on heuristic algorithms. AS an extended work of REDUX project, REDUX2 in Chapter 4 make various more contribution especially with a section detailed with statement problem. The upgraded REDUX3 project which focus on the energy demand side with renewable-aware workload scheduler will be demonstrated in Chapter 5. Finally, Chapter 6 concludes the dissertation, and points out the future research.

Chapter 2

Related Work

The topics related to energy efficiency of data centers is broadly and diversely discussed. Numerous methods are proposed to save energy cost for data centers. This chapter briefly presents previous research in building energy-efficient data centers.

2.1 Computing Cost

A lot of research have been done in reducing computing cost of data centers [4] [59] [70]. For instance, CMPs are widely used in data centers, and the frequency/voltage of CPU cores could be adjusted in order to save power consumption. Popular strategies to reduce computing cost include redistributing workload and powering off idle disks or data nodes. For example, an energy-efficient strategy was proposed which specifies a subset of disks as cache disks and dispatches workloads to these cache disks while making the other disks spin down [16]. Another strategy introduced a Popular Data Concentration (PDC) technique that migrates frequently accessed data to a subset of disks [54]. Then the other disks which are not accessed frequently could be transitioned to low-power mode, and the total computing cost of these data nodes could be reduced.

Many researchers concentrate on resource management and task scheduling in data centers to decrease computing energy consumption [3] [7] [8] [44] [61]. For instance, Beloglazov and Buyya proposed an energy-efficient resource management system for virtualized Cloud data centers [8]. In this system, VMs are consolidated according to the utilization of resources, and virtual network topologies are built between VMs and thermal status of computing nodes to save energy. This management system reduces the operational costs of data centers and provides the required Quality of Service (QoS). Beloglazov *et al.* also demonstrated an architectural framework (including resource provisioning and allocation algorithms) and principles for energy-efficient Cloud computing [7]. Experimental results show that their Cloud computing model has immense potential in energy saving and energy efficiency improvement under dynamic workload scenarios. In addition, Aksanli *et al.* demonstrated an adaptive job scheduler that utilizes the prediction of solar and wind energy production [3]. This job scheduler improves the energy efficiency by three times. Lee and Zomaya pointed out that under-utilized resources account for a large amount of energy use and resource allocation strategies could be applied to achieve high energy efficiency [44]. They proposed two task consolidation heuristics methods that aim to maximize resource utilization and take into account of both active and idle energy consumption. Experimental results illustrated the energy saving capability of their heuristics.

With the growing of data center density and size, designers should take into account of both energy costs and carbon footprint. Altering the usage patterns of data centers is believed to be a practical method to affect demand response. Chiu *et al.* pointed out that shifting computational workloads across geographic regions to match electricity supply may help balance the electric grid [14]. They proposed a symbiotic relationship between data centers and grid operators and a low cost workload migration mechanism. Ren and He proposed an online algorithm, called COCA (optimizing for COst minimization and CArbon neutrality), for minimizing operational cost in data centers while satisfying carbon neutrality without long-term future information [55]. COCA enables distributed server-level resource management: each server autonomously adjusts its processing speed and optimally decides the amount of workloads to process. Analysis of trace-based simulation studies show that COCA reduces cost by more than 25% (compared to state of the art) while resulting in a smaller carbon footprint.

2.2 Energy-Efficient Computing

Cloud computing has become a popular topic in recent years. A study shows that coal and nuclear, which generate severe air pollution, are used to satisfy these large amount of electrical energy demand [17]. Confronting with the rapid increment of energy consumption and severe air pollution, growing attention has been paid to build energy-efficient data centers [1] [20] [37] [43]. At the same time, small or medium sized organizations began to move their computing applications to an Internet-based "cloud" platform in order to improve energy efficiency [66]. A large body of early studies focused on reducing power consumption of a single server by applying the dynamic voltage and frequency scaling technique (i.e., DVFS) [13], low-power chipsets [27], and advanced cooling techniques [22]. Emerging energy-management schemes aim to optimize energy efficiency of servers equipped with multi-core processors [38], GPUs [23], and smart memory cubes [5]. In contrast to the above energy-efficient computing strategies, our REDUX pays attention to reducing energy cost of large-scale data centers.

2.3 Conserving Energy in Data Centers

In recent years, there has been a pressing need to minimize the total power consumption of data centers [41] [21]. Chun *et al.* investigated a hybrid datacenter architecture mixing low power systems and high performance ones [15]. Dabbagh *et al.* developed an integrated energy-aware resource provisioning framework to optimize energy efficiency of data centers [18]. Shuja *et al.* investigated practical approaches to maximizing quality of services while minimizing energy consumption of data center resources. Virtualization techniques have been applied to reduce the power consumption by consolidating multiple virtual machines into one physical server [49]. For example, Tseng *et al.* designed a virtual machine management strategy to predict resource utilization for upcoming videos, thereby turning off idle servers to conseve energy [62]. Another thread of techniques that minimize the electricity cost of data centers is exploiting the temporal and regional diversity of electricity prices (see, for example, [47]). The pragmatic idea of these schemes is to dynamically allocate resources to service workload in data centers with a cheap electricity price. Unlike the existing energy-saving techniques, our proposed REDUX conserves energy by taking a full advantage of renewable energy and UPS devices by taking a full advantage of renewable energy and UPS devices.

2.4 Renewable Energy in Data Centers

There are pressing demands from cloud service providers for cost-effective renewable energy to power data centers [46]. An increasing number of prior studies focused on making green data centers through the integration of renewable power sources [28] [32]. A handful of resource management techniques were developed to address the shortfall intermittency of renewable power. For example, Goiri *et al.* designed a parallel batch job scheduler to match the computational workload with solar energy supplies in a datacenter [30]. To integrate renewable energy, Guo *et al.* incorporated geographical load balancing, opportunistic workload scheduling, and thermal storage management in data centers [36]. And benefits of on-site renewable sources include (1) quick scaling up in power capacity to accommodate load growth and (2) shortening construction time interval compared with conventional power plants [45]. More recently, S.Kwon proposed a two-stage stochastic program, and developed a mathematical optimization model for energy-efficient while ensuring the desired level of renewable energy utilization and quality of service [42]. On the other hand, we applied a practical heuristic algorithm method with unified interface-backbone layer framework design in this study. G.Zhang *et al.* focus their study on addressing the problem of energy management for geo-distributed data centers with renewable resources and energy storages, and aim to minimize the long-term operation cost by leveraging the spatiotemporal diversity of grid power price, water consumption and carbon emission. [71]. However, the intermittent nature of renewable energy and possible dynamic changing grid energy price was ignored.

2.5 Distributed UPS Systems

Uninterruptible power supplies (i.e., UPS devices) offer efficient peak power shaving in data centers [2] [40] [48]. For example, Google data centers employ server-level UPS units, after which a battery is attached on each server [33]. Urgaonkar *et al.* proposed the usage of UPS units to offer ample opportunities of cost reduction in data centers [63], by first prove the feasibility of leveraging energy storage device (e.g., lead-acid batteries) in data centers to reduce peak power cost. Aksanli *et al.* applied distributed UPS devices to store energy during low load activity periods while using UPS stored energy during power spikes [2]. Kontorinis *et al.* explored the total cost of operation of the distributed UPS system in data centers and proposed using local distributed UPS systems to shave the data center's peak power [40]. Govindan *et al.* designed a UPS-based energy buffer to reduce energy costs in data centers [33]. To better utilize the temporal variation of electricity prices, Wang *et al.* deployed UPS systems to shift demand peak away from high-price periods in data centers [65]. To help prolong the live-span and improve the reliability of the UPS devices, S. S. H. Bukhari *et al.* proposed line-interactive or online UPS system that eliminates the harmful inrush current phenomenon [10] [11] [9]. Unlike the aforementioned techniques, REDUX stores energy in UPS units during intervals when renewable energy is stable and incessant or power grid prices are low. For the rest of this paper, we use terms UPS or battery interchangeably.

2.6 Variable Grid Power Price

The discussion of variation of grid (brown) energy price mainly falls into two categories: on-peak/off-peak [30] [32], and random pricing determined by the electricity market [21]. In on-peak/off-peak pricing, grid energy costs less when used during off-peak consumption times and more when consumed during on-peak times and the difference between on-peak and off-peak prices is largest in the summer time [30]. While Deng *et al.* [21] assumes that grid energy price is a random variable determined by pricing policies in the electricity market. While do not make any assumption about the stochastic pattern of grid energy price, they assume that duration of one time slot is small enough (e.g. 1 hour) so that grid energy price remains stable within each time slot. Grange *et al.* [35] concludes that taking the fluctuating grid energy price into account leads to a small increase of non-renewable energy use, but allows an important reduction of the grid energy cost. Through out this dissertation, we follow design of Deng *et al.* [21] with variable grid energy price and using real-world electricity power price.

2.7 Renewable-Aware Scheduling

Curtailing energy consumption has become a necessary objective of modern job schedulers running on large-scale computing system housed in data centers. For example, Zhou *et al.* [72] proposed a power-aware job scheduling approach incorporating variable energy prices and job power profiles. The scheduler relies on a power budget for energy savings, thereby degrading system utilization slightly during on-peak electricity price period. Yang *et al.* [68] intended to provide a generic job-power-aware scheduling mechanism to minimize electricity bills without impacting system utilization during both on-peak and off-peak pricing periods. Wallace *et al.* [64] proposed a data-driven scheduling approach to controlling power consumption of an entire system under any user defined budget. The possibility of cheap and clean renewable energy were overlooked in the above studies. Renewable-energy-aware schedulers receive attention in the research community. For example, Goiri *et al.* [31] proposed GreenSlot, a scheduler for parallel batch jobs in a data center powered by a photovoltaic solar array. Grid power is only treated as a backup energy source, where off-peak cheap grid energy is never stored in any energy storage device. Grange *et al.* [35] again implemented the "GreenVarPrice" version of the GreenSlot scheduler [30], which takes into account the variation of grid energy price. Grange *et al.* highlighted a comparison between the renewable-aware scheduler and the traditional energy-aware scheduler. In these schedulers, there is the lack of a unified high- level framework design.

Chapter 3

REDUX: Managing Renewable Energy using Distributed UPS Systems

To develop environmental friendly and energy-efficient data centers, it is prudent to leverage on-site renewable sources like solar and wind. Data centers deploy distributed UPS systems to handle the intermittent nature of renewable energy. In this pilot study of our dissertation research, we propose a renewable-energy manager called *REDUX*, which offers a smart way of managing server energy consumption powered by a distributed UPS system and renewable energy. REDUX maintains a desirable balance between renewable-energy utilization and data center performance. REDUX makes judicious use of UPS devices to allocate energy resources when renewable energy generation is in low or fluctuate condition. REDUX not only guarantees the stable operation of daily workload, but also reduces the energy cost of data centers by improving power resource utilization. Compared with existing strategies, REDUX demonstrates a prominent capability of mitigating average peak workload and boosting renewable-energy utilization.

The rest of the chapter is organized as follows. Section 3.1 serves as introduction and background of the pilot project of REDUX system. Section 3.3 articulates an array of heuristic algorithms at the heart of REDUX. Section 3.4 describes the experimental settings and results. Finally, Section 3.5 summarizes this chapter.

3.1 A Pilot Project for REDUX and Its Successors

In the first part of the dissertation, we spearhead an effort to develop a resource manager called *REDUX* to cost-effectively allocate energy resources by incorporating a distributed UPS system tailored for renewable energy like solar. The REDUX system lays out a solid foundation for its successors - REDUX2 (see Chapter 4), REDUX3 (see Chapter 5). This section briefly reviews the background of renewable energy and distributed UPS in Section 3.1.1, followed by presenting the prominent features of REDUX in Section 3.1.2. Section 3.1.3 presents a comparison table where major differences between the REDUX system and related systems are tabulated.

3.1.1 Renewable Energy with Distributed UPS and Fluctuating Grid Energy Price

While considering renewable energy, existing power management techniques for data centers either overlook the usage of uninterruptible power supplies (i.e., UPS devices) or pay no attention to power-grid price fluctuation. REDUX, developed in the pilot study of the dissertation research, aims to minimize energy cost by deploying UPS devices in data centers to address two challenging issues, namely, the intermittency feature of renewable energy and dynamically changing electricity price. REDUX is responsible for maintaining stable operation of daily computing workload in data centers while cutting back energy costs by improving overall energy resource utilization.

In contrast to brown energy, renewable energy harvested from sources such as wind turbines and solar panels exhibits an assortment of benefits [50]. Most renewable energy sources are clean and produce few global warming emissions while reduces energy costs by offering affordable electricity. Renewable energy offers affordable electricity that reduces energy costs. Renewable energy supplies, being highly modular, can incrementally increase power capacity to match load growth. Moreover, renewable-energy price tends to remain stable after completing on-site installations [12], where on-site renewable energy facilities, of course, may be constructed in a shorter time framework compared with traditional power plants [45]. As such, a growing number of data centers have kicked off various initiatives to integrate on-site renewable sources into power supplies. For instance, *Apple* has recently constructed a massive 100-acre solar farm adjacent to the *iCloud* data center in North Carolina. This solar farm yields 84 million kWh of clean renewable energy annually.

The discussion of variation of grid (brown) energy price mainly falls into two categories: on-peak/off-peak [30] [32], and random pricing determined by the electricity market [21]. In on-peak/off-peak pricing, grid energy costs less when used during off-peak consumption times and more when consumed during on-peak times and the difference between on-peak and off-peak prices is largest in the summer time [30]. While Deng *et al.* [21] assumes that grid energy price is a random variable determined by pricing policies in the electricity market. While do not make any assumption about the stochastic pattern of grid energy price, they assume that duration of one time slot is small enough (e.g. 1 hour) so that grid energy price remains stable within each time slot. Grange *et al.* [35] concludes that taking the fluctuating grid energy price into account leads to a small increase of non-renewable energy use, but allows an important reduction of the grid energy cost.

Two popular techniques preventing costly down time incurred by power budget violations are (1) over-subscribing power infrastructures and (2) power capping [40]. Data centers powered by over-subscribed resources inevitably encounter a problem of power premium charge ascribed to the peak power: the premium cost contributes a major portion of electricity bills. We advocate for the UPS-based scheme because UPS devices allow us to effectively hide the extra power from power grids by completely shaving high-magnitude power spikes in a short time window.

Distributed UPS devices, representing an economical solution, enable data-center operators to dynamically shave peak power with stored energy [40]. We employ a distributed UPS system in our pilot study because such a system are efficient, highly scale-able, and adequately reliable. Distributed UPS systems also address the issue of single-point failures: our initial design equipped with the distributed UPS system scales up with growing data-center sizes.

3.1.2 Prominent Feature of REDUX

In this pilot project of the dissertation research, we advocate managing energy resources by leveraging a distributed UPS system to make data centers economically and environmentally friendly. We develop the *REDUX* manager to allocate renewable energy integrated with a power grid. REDUX deals with the three supply levels of renewable energy resources, namely, high, fluctuate, and low. REDUX makes judicious decisions on UPS charging and discharging with sufficient information on renewable-energy supply levels and time-dependent grid power price. For instance, REDUX makes full use of UPS devices to furnish energy resources when renewable energy generation is low or under a fluctuate condition.

One prominent feature of REDUX is that it orchestrates a desirable balance between renewable-energy utilization and data center performance. Importantly, REDUX conserves the energy cost of data centers through power resource management while offering reliable computing operations. Compared with existing strategies, REDUX demonstrates a prominent capability of mitigating average peak workload and boosting renewable-energy utilization.

3.1.3 A Comparison Table

Let us classify the previous studies that are related to our pilot project into three core areas, namely, (1) renewable energy in data centers, (2) fluctuating grid power price, and (3) distributed UPS systems; Together with other prominent features in hierarchical design style and energy efficient control management including management regions and dynamic Management. Table 3.1 summarizes the major differences between our proposed REDUX and the existing power management techniques for data centers.

In what follows, we briefly discuss the related systems and solutions listed in Table 3.1.

- ReUPS [48] Successfully integrated both renewable energy and a centralized UPS system, but ignores the fluctuation nature of grid power price, and lack of a hierarchical design of the framework.

Table 3.1: Comparisons between REDUX and the existing power management techniques for data centers.

Power Management	UPS Enabled	Renewable Energy	Management Regions	Grid Price Fluctuations	Dynamic Management	Hierarchical Design
ReUPS [48]	√	√	√	×	√	×
Greenworks [46]	√	√	×	×	×	√
iSwitch [45]	×	√	√	×	×	√
EcoPower [21]	×	√	×	√	√	×
GreenSlot [30]	×	√	×	√	√	×
GreenSwitch [32]	×	√	×	√	√	×
REDUX (This Study)	√	√	√	√	√	√

- Greenworks [46] Also considered both renewable energy and a centralized UPS system and ignores the fluctuation nature of grid power price, however unlike REUPS [48], it lacks consider of management regions and dynamic management

- iSwitch [45] did not considered any energy storage method like distributed UPS system, which will lead to waste of energy when renewable energy is sufficient or grid power price is cheap.

- EcoPower [21], GreenSlot [30] and GreenSwitch [32] noticed the variable grid energy price and introduced the dynamic management of energy supply and demand. However, they both lack a energy storage mechanism and a unified hierarchical framework design.

3.2 Framework Design

We start this section by depicting the REDUX system framework (see Section 3.2.1), which offers a high-level design from the perspective of power management in data centers. Next, we shed light on the responsibilities of the core modules in REDUX (see Section 3.2.2).

3.2.1 The REDUX Platform

REDUX is a management system orchestrating UPS units and renewable energy for computing servers according to workload requirements in data centers. In what follows, we outline the correlations between REDUX and the external entities (e.g., renewable energy), followed by the descriptions of the primary modules in REDUX.

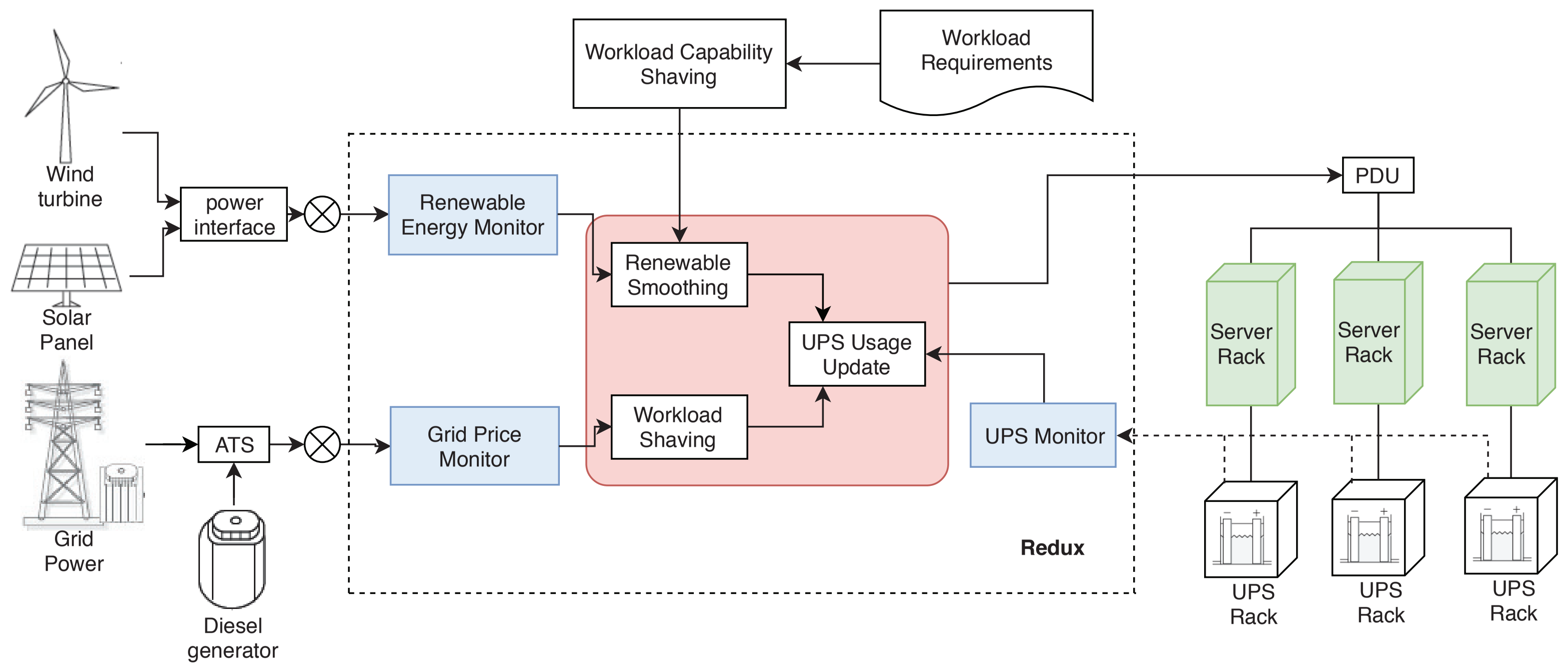

Figure 3.1: The framework of the *REDUX* system. REDUX orchestrates renewable energy, UPS units, and server racks according to workload requirements.

Fig. 5.1 plots the framework design of the REDUX system, where renewable energy (e.g., wind and solar), grid power, and diesel generators are seamlessly integrated (see the left-hand side of Fig. 5.1). REDUX employs a distributed UPS system, where a UPS device is attached to each server rack (see the right-hand side of Fig. 5.1).

REDUX persistently communicates with the five external entities, namely, renewable energy supplies, a power grid, UPS units, server racks and workload requirements. A wide range of energy supplies may include, but is not limited to, wind and solar energy, an electrical grid and diesel generators. The distributed UPS system aims to buffer energy under light workload and consume the UPS energy in case of power spikes.

There are two distinctive ways of acquiring workload requirements forwarded to REDUX. First, in the batching mode, the workload requirements become available when users submit jobs to data centers. Second, in case of the interactive mode, the workload requirements are proactively and constantly monitored and predicted by data center operators. The estimated workload requirements are then considered as the trace data inputs of workload requirements,

REDUX consists of two layers : an interface layer (see all light-blue colored modules in Fig. 5.1) and a backbone layer (see the modules inside the light-red colored box in Fig. 5.1). The interface layer is comprised of four modules, including the renewable energy monitor, the grid price monitor, the UPS monitor, and the workload capability shaving module. After the workload capability shaving module retrieves workload requirements from clients, the workload information is processed and forwarded to REDUX. The backbone layer is detailed in the next subsection.

3.2.2 The Backbone of REDUX

The three core modules in the backbone layer communicate with five external entities (i.e., renewable energy supplies, a power grid, UPS units, server racks, workload requirements) through the four interface modules in the interface layer.

The backbone layer of REDUX contains three modules, namely the renewable smoothing module, the workload shaving module, and the UPS-usage update module. The algorithms of these three modules are presented in Section 3.3. The functionalities of the three core modules of REDUX are summarized as follows:

- **Renewable Smoothing.** The renewable smoothing module (see Algorithm 2 in Section 3.3.) is responsible for effectively dealing with cases where renewable energy fluctuates due to an environmental status. This module copes with the fluctuating renewable energy through a distributed UPS system.

- **Workload Shaving.** The workload shaving module is focused on treating overload conditions, in which workload exceeds the underlying data center's computing and energy capacity. This module is invoked in Steps 3 and 10 of Algorithm 1 in Section 3.3.

- **UPS-usage Updating.** The responsibility of the UPS-usage update module is two-fold. First, this module keeps track of the status of all UPS units in a data center. Second, the module is in control of making battery charging and discharging decisions. Refer to Algorithm 3 in Section 3.3 for the design of the UPS-usage update module.

3.3 Heuristic Algorithms

In this section, we formally formulate the heuristic algorithms of REDUX system. We first present a high-level control algorithm in Section 3.3.1. Then, Sections 3.3.2 and 3.3.3 shed light on the design of the smoothing and updating algorithms implemented in the backbone layer (see also Section 3.2.2) of REDUX.

3.3.1 The High-Level Control Algorithm

We take a hierarchical design approach to the development of REDUX, in which a high-level control algorithm coordinates all the three core modules in the backbone layer (see the backbone layer in Section 3.2.2). This subsection is dedicated to the design of the high-level controller, whereas the subsequent subsections articulate the algorithms of the core modules, namely, renewable energy smoothing and UPS-usage updating (see also Sections 3.3.2 and 3.3.3).

Algorithm 1 The high-level control algorithm in REDUX.

Require:
  Renewable energy states $S_{RE} = \{STA, FLU, OTA\}$
  Workload State $S_w = \{OVR, OFP\}$
  Energy demand from workload during time t, W_t
Ensure:
  Redux Cost of All Energy
1: $t = 0$
2: **while** $t <= T$ **do**
3: $workload\_smoothing()$; /* with workload capability */
4: $update\_ups\_utilitylevel()$;
5: **if** $S_{RE} = FLU$ **then**
6: $ren\_supply\_smooth()$;
7: **end if**
8: $estimate\_overpeak\_level()$;
9: **if** $(S_w = OVR)$ **then**
10: $workload\_smoothing()$;
11: /* with estimated overpeak workload */
12: **end if**
13: $update\_ups\_supply()$;
14: $E_t^G = W_t \; - \; E_t^{RE} \; - \; E_t^{UPS}$
15: $calculate\_energy\_cost()$;
16: $t++$
17: **end while**
18: **return** Redux cost

Algorithm 1 depicts the pseudo-code of the high-level algorithm in REDUX, where the state supply of renewable energy S_{RE} has these cases: Stable (STA), Fluctuate (FLU) and Outage (OTA); And the state of current workload S_w also has three cases: Incapable (INC), Overpeak (OVR) and Offpeak (OFP); Finally, The state of pirce level of grid S_{GP} is either $HIGH$ or LOW. The high-level coordinator kicks off the power management task with the workload-shaving procedure (see Line 3).

The purpose of workload shaving is the overloading of a data center with respect to power supplies and computing capacity. More specifically, if the submitted load is below the overloading threshold, the pending jobs will be immediately processed in the current time slot. Otherwise, these jobs will be queued and delayed by the high-level controller. Jobs resulting in overload should be scheduled and processed in the subsequent time slots. It is note worthy that scheduling such jobs under the overloading condition imposes significant impact on the data center's overall performance metrics such as throughput and response time. The investigation of scheduling policies in the overloading case is out of scope of this study. We intend to address this concern as a future research direction.

The procedure of updating UPS utility levels (see $update\_ups\_utilitylevel()$ in Line 4) aims to protect UPS units against frequent charging and discharging, thereby prolonging lifetime of the distributed UPS system. This procedure determines a UPS energy level under which the UPS units should be protected. Such levels largely depend on current workload conditions as well as required energy capacity.

If the renewable energy is in the fluctuation state, the renewable-energy-supply smoothing policy is invoked in Line 5 of Algorithm 1. The smoothing procedure is governed by the current renewable supply, a stable energy supply level, and a UPS energy level. Line 8 (see also $estimate\_overpeak\_level()$) of the high-level control algorithm is dynamically and approximately project a peak workload level. Such estimation is accomplished by providing the information pertaining to the current workload, renewable supply, UPS energy level and electricity price. If the workload exceeds the peak level (see Line 9), then the workload-shaving procedure is kicked in to defer the extra load to the next time slots (see Line 10). The functionality of Line 13 (see $update\_ups\_supply()$ in the top-level controller is two-fold: (1) updating the UPS status information and (2) making UPS discharging and charging decisions. After Line 14 computes the energy supply from the electrical grid, Line 15 calculates the energy cost. The $calculate\_energy\_cost()$ function enables us to quantitatively compare REDUX against two exiting power management strategies (see also Section 3.4).

3.3.2 Smoothing Renewable Energy Supply

Recall that Line 7 in Algorithm 1 is a smoothing operation for renewable energy supplies. This operation is dedicated to smooth renewable energy output for data centers.

Algorithm 2 Smoothing Renewable Energy Supplies

Require:

grid price $P^G(t)$, and state S_{GP}
renewable price P^{RE}, renewable supply E^{RE}
updated stable renewable supply E^{STA}

Ensure:

updated renewable supply $\bar{E}^{RE}$
updated stable renewable supply $\bar{E}^{STA}$

1: **if** $E^{RE} < E^{STA}$ **then**
2:     **if** $S_{GP} = HIGH$ **then**
3:         Discharge UPS units until stable renewable supply;
4:     **else**
5:         Use the power grid for smoothing;
6:         **if** $P^G(t) < P^{RE}$ **then**
7:             Recharge the UPS units;
8:         **end if**
9:     **end if**
10: **else**
11:     Use renewable energy to power servers;
12:     Recharge the UPS units using renewable energy;
13: **end if**

Algorithm 2 outlines the pseudo-code of the smoothing operation. First of all, to simulate a stable threshold for the following steps, we again use the exponential windowed average method to figure out the result of stable renewable supply level. Then, if the renewable energy supply is lower than the stable threshold (i.e., see $E^{RE} < E^{STA}$ in Line 1), the renewable energy output fluctuates due to renewable resource variations. In such a fluctuation case, energy should be supplemented by the underlying UPS systems (see Lines 2-3) or the power grid (see Lines 5-8). Specifically, when the price of the electricity grid is high, the UPS units discharge and supply energy to the servers (see Line 3). In case the power grid offers cheap energy, REDUX fully utilizes the grid to smooth the renewable energy supply (see Line 5). It is economically wise to recharge the UPS system using the power grid at cheap price (see Line 7). The servers will be powered by cost-effective renewable energy if the renewable energy generation is sufficient (see Line 11). Further, when the renewable resources are abundant, the UPS devices are recharged by the renewable energy (see Line 12).

3.3.3 Updating UPS Energy Supply

Now we delineate an algorithm for UPS energy supply and storage control. This algorithm is invoked in Step 13 in Algorithm 1 (see Section 3.3.1. The UPS storage control algorithm keeps track of the status of the UPS energy supply and storage(i.e., $E^{UPS}(t)$ and $E^{UPS}_{storage}(t)$).

Algorithm 3 illustrates the procedure of updating UPS energy supply and storage information under three distinctive renewable-energy conditions, namely, the outage (see Lines 1-8), fluctuation (see Lines 9-14), and stable state (see Lines 15-19).

If a renewable energy outage occurs, energy must be supplied by either the power grid or the distributed UPS system. REDUX chooses to discharge the UPS units if the current workload is high and the power price from the grid is expensive (see Lines 2-3); the UPS devices are recharged when the grid price becomes cheap (see Lines 5-6).

When the renewable energy resource is in the fluctuation state and the current workload is below the peak level, the UPS units are recharged by cheap power grids (see Lines 10-11).

In case of abundant renewable energy resources (i.e., $S_{RE} = STA$), REDUX decides to recharge the UPS units under one of the two conditions: (1) the power price of the grid is fairly low or (2) current workload is in the off-peak window and the power grid price is high (see Lines 16-17).

3.4 Preliminary Results from the Pilot Project

In this section, we evaluate the performance of REDUX driven by real-world renewable energy and power grid price data. We compare REDUX with a baseline solution called

Algorithm 3 Updating UPS Energy Supply

Require:

Energy demand from workload amount in time slot t, $W(t)$
updated over-peak workload, W_{OVR}
Renewable energy states, $S_{RE} = \{STA, FLU, OTA\}$
Grid price states, $S_{GP} = \{HIGH, LOW\}$
UPS energy storage, $E^{UPS}_{storage}(t)$

Ensure:

net UPS energy supply, $E^{UPS}(t)$
updated UPS energy storage, $\bar{E}^{UPS}_{storage}(t)$

if $S_{RE} = OTA$ **then**
    if $W_t >= W_{OVR}$ **and** $S_{GP} = HIGH$ **then**
        Discharge UPS units;
    end if
    if $W_t < W_{OVR}$ **and** $S_{GP} = LOW$ **then**
        recharge UPS units until full
    end if
end if
if $S_{RE} = FLU$ **then**
    if $W_t < W_{OVR}$ **and** $S_{GP} = LOW$ **then**
        Recharge the UPS units;
        Update the UPS supply with E^{UPS};
    end if
end if
if $S_{RE} = STA$ **then**
    if $S_{GP} = LOW$ **or** $(W_t < W_{OVR}$ **and** $S_{GP} = HIGH)$ **then**
        Recharge the UPS units while renewable energy is accessible;
    end if
end if

noREDUX as well as two existing power management techniques - *GreenSwitch* [32] and *ReUPS* [48]. We measure the accumulated energy cost of a simulated data center governed by REDUX, noREDUX GreenSwitch, and RE-UPS under various conditions. Before presenting experimental results, let us discuss the experimental settings in Section 3.4.1.

3.4.1 Experimental Settings

In our experiments, we test REDUX and its competitors using the electricity price of a real-world power grid in New York state during a period of three months from June 1 to August 31, 2006. We pay attention to this time period, because the solar data was collected

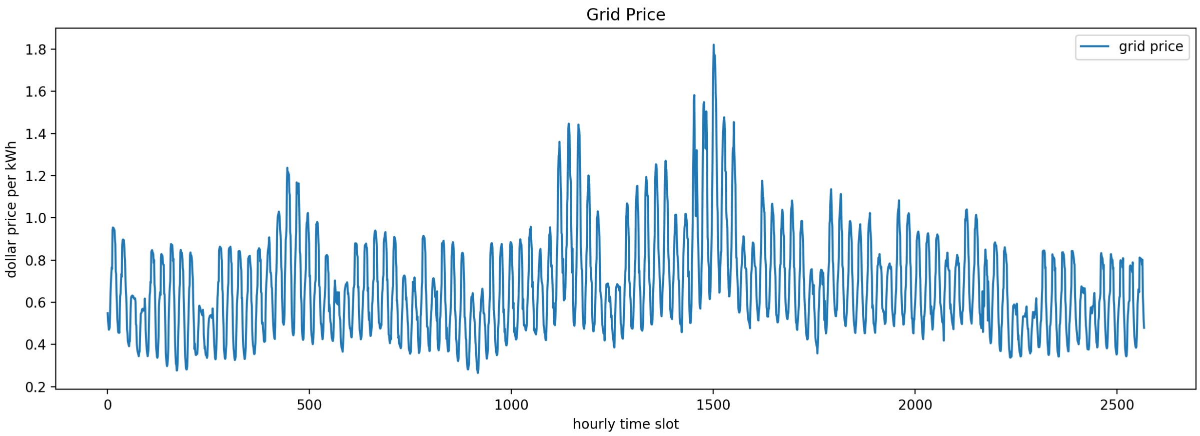

Figure 3.2: The electricity price of a power grid in the New York state during the period of three month ranging from June 1 to August 31, 2006. Unit: $\times 100\$/\mathrm{MWh}$

during this period. Fig. 5.4 shows that the power grid price dynamically changes during the three-month time interval, where three noticeable price peaks occur at around 480, 1200, and 1500 hours. The maximum and minimum prices in the trace are 182 and 26 $/MWh, respectively.

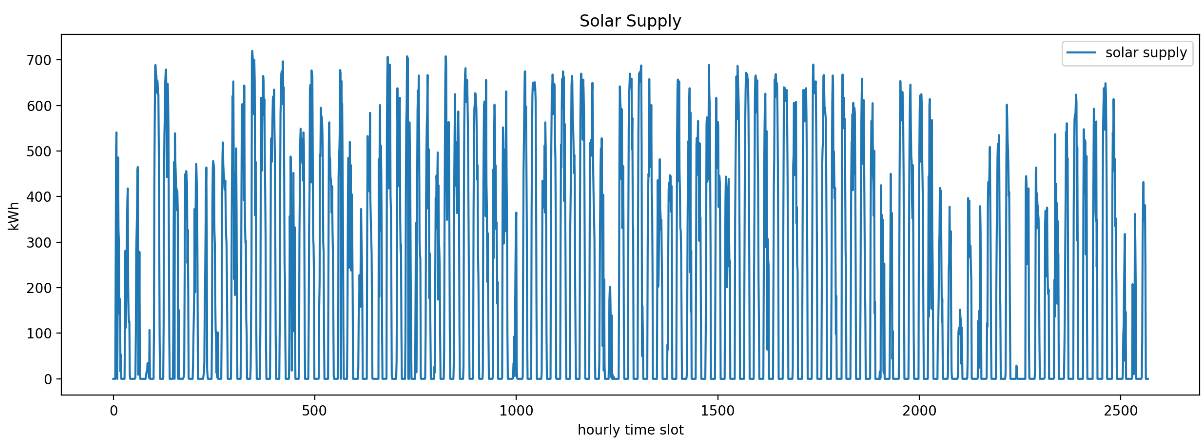

Figure 3.3: The solar supply of New York state during the three-months period ranging from June 1 to August 31, 2006.

From the National Renewable Energy Laboratory's (NREL) database, we retrieve the solar supply of New York state during the three-month period from June 1 to August 31, 2016. The solar supply data measured in KWh is plotted in Fig. 3.3. The solar and wind power price is set at \$0.09/KWh and \$0.07/KWh, respectively. When it comes to the UPS devices, the energy cost for recharging and discharging batteries is set to \$0.07/kWh.

3.4.2 REDUX vs. noREDUX

In the first experiments, we compare REDUX with noREDUX - a baseline power manager where renewable energy resources and UPS units are not incorporated.

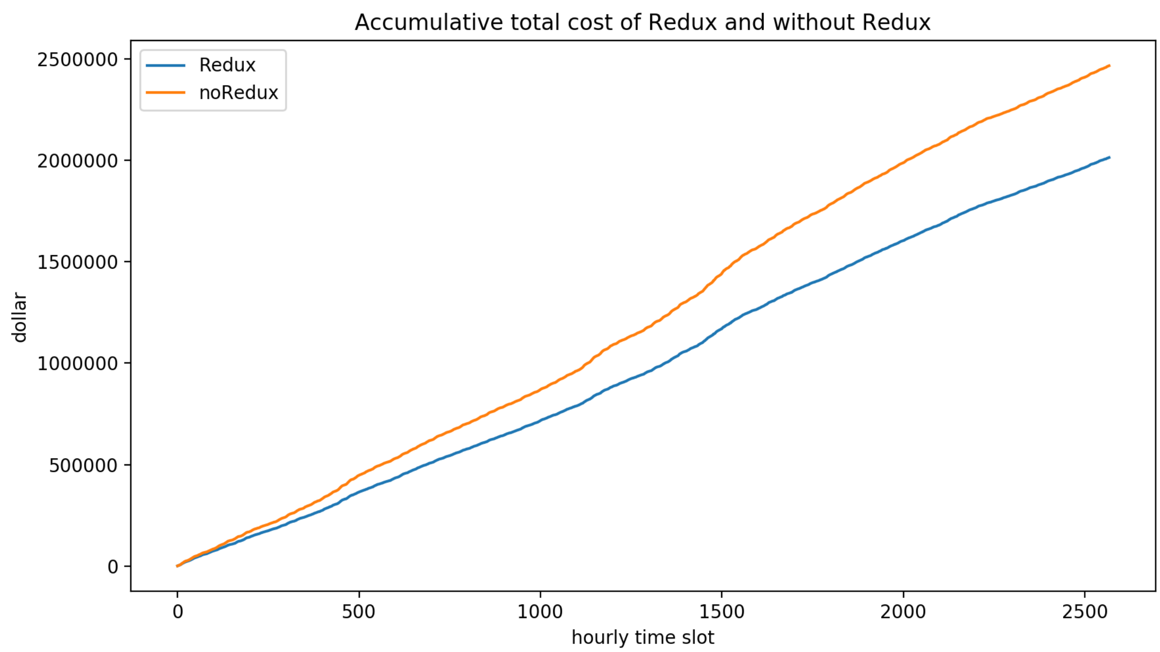

Figure 3.4: REDUX significantly reduces the energy cost of a data center managed by noREDUX - a traditional scheme.

Fig. 3.4 reveals the energy-cost comparison between REDUX and noREDUX. Regardless of REDUX or noREDUX, the cumulative energy cost linearly increases with time. The results show that REDUX significantly cut back the energy of the data center governed by noREDUX. For example, REDUX is able to reduce the energy cost of noREDUX by 32.6% by the end of the first month; such cost savings becomes 27.7% and 24.0% by the end of the second month and third month, respectively.

The energy-cost savings are expected because REDUX orchestrates renewable energy resources and UPS units to conserve energy cost when the electricity price of the power grid is expensive. For example, there are multiple sharp peaks (e.g., see 480, 1200, and 1500 hours in Fig. 5.4) in the grid price trace data. REDUX makes judicious decisions to power the data center using either renewable energy or the distributed UPS system rather than the expensive power grid.

3.4.3 Improving GreenSlot and ReUPS

Now we compare REDUX with two existing schemes - GreenSlot and ReUPS. Fig. 3.5 shows the cumulative energy cost of data centers managed by REDUX, GreenSlot, and ReUPS, respectively. Like the trend plotted in Fig. 3.4, the energy cost of the three management schemes grows with time.

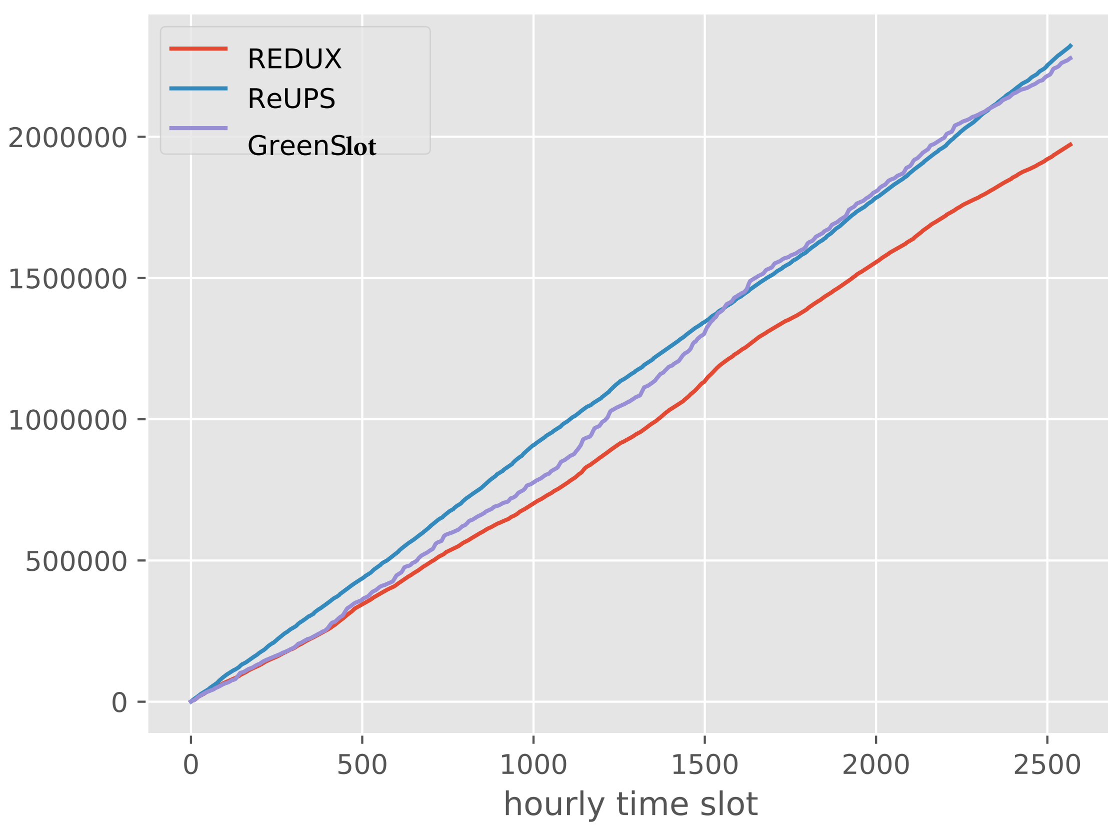

Figure 3.5: REDUX reduces the accumulated energy cost of a data center managed by other schemes

More importantly, Fig. 3.5 illustrates that REDUX noticeably outperforms GreenSlot and ReUPS in terms of energy cost. For instance, by the end of the first month, REDUX slashes the energy cost of GreenSlot and ReUPS by 10.3% and 26.5%, respectively; such cost savings becomes 16.8% and 14.5% by the end of second month, and 15.5% and 17.7% by the end of the third month, respectively. Table 3.2 summarizes the cumulative energy cost of the three management strategies by the end of the first, second, and third month, respectively. The energy-cost saving rates of REDUX over GreenSlot and ReUPS can also be found in Table 3.2.

REDUX is superior to GreenSlot, because REDUX makes use of the distributed UPS system to smooth renewable energy supply, which helps REDUX to smarter control the energy resources distribution over time slots. The improvement of REDUX over ReUPS is ascribed to the awareness of changing power grid price. REDUX makes full advantage of renewable energy coupled with the UPS devices to offer energy supply when the power grid price becomes high (see, for example, multiple sharp peaks in Fig. 5.4). When the power grid price is surging, REDUX switches power supply from the grid to either renewable energy resources or the UPS system, depending on current workload conditions.

Table 3.2: Cumulative energy-cost comparisons among REDUX, GreenSlot, and ReUPS the end of the first, second, and third month.

Power Management	**1st Month**	**1st Month Reduction Rate**	**2nd Month**	**2nd Month Reduction Rate**	**3rd Month**	**3rd Month Reduction Rate**
REDUX (This Study)	$604,821.44	N/A	$1,330,663.89	N/A	$1,972,270.34	N/A
GreenSlot [32]	$765,215.58	10.3%	$1,524,093.47	16.8%	$2,321,717.9	15.5%
ReUPS [48]	$667,445.95	26.5%	$1,555,229.6	14.5%	$2,278,287.49	17.7%

3.4.4 Impacts of Renewable Energy

In the last experiment, we investigate the impact of renewable energy level on the performance of REDUX and GreenSwitch. The performance of ReUPS is close to that of REDUX in this experiment and; therefore, we ignore the results of ReUPS in this subsection. Fig. 3.6 shows the energy cost of REDUX and GreenSwitch when the renewable energy supply is set at the levels of 50, 500 and 950 kWh. The results suggest that when the renewable energy level increases from 50 to 500, the overall energy cost of the data center is cut back significantly. Unfortunately, when the renewable energy level increases further to 950, there is no noticeable reduction in energy costs. When the renewable energy level is 50, energy costs of REDUX is much lower than that of GreenSwitch thanks to the adoption of UPS units in REDUX.

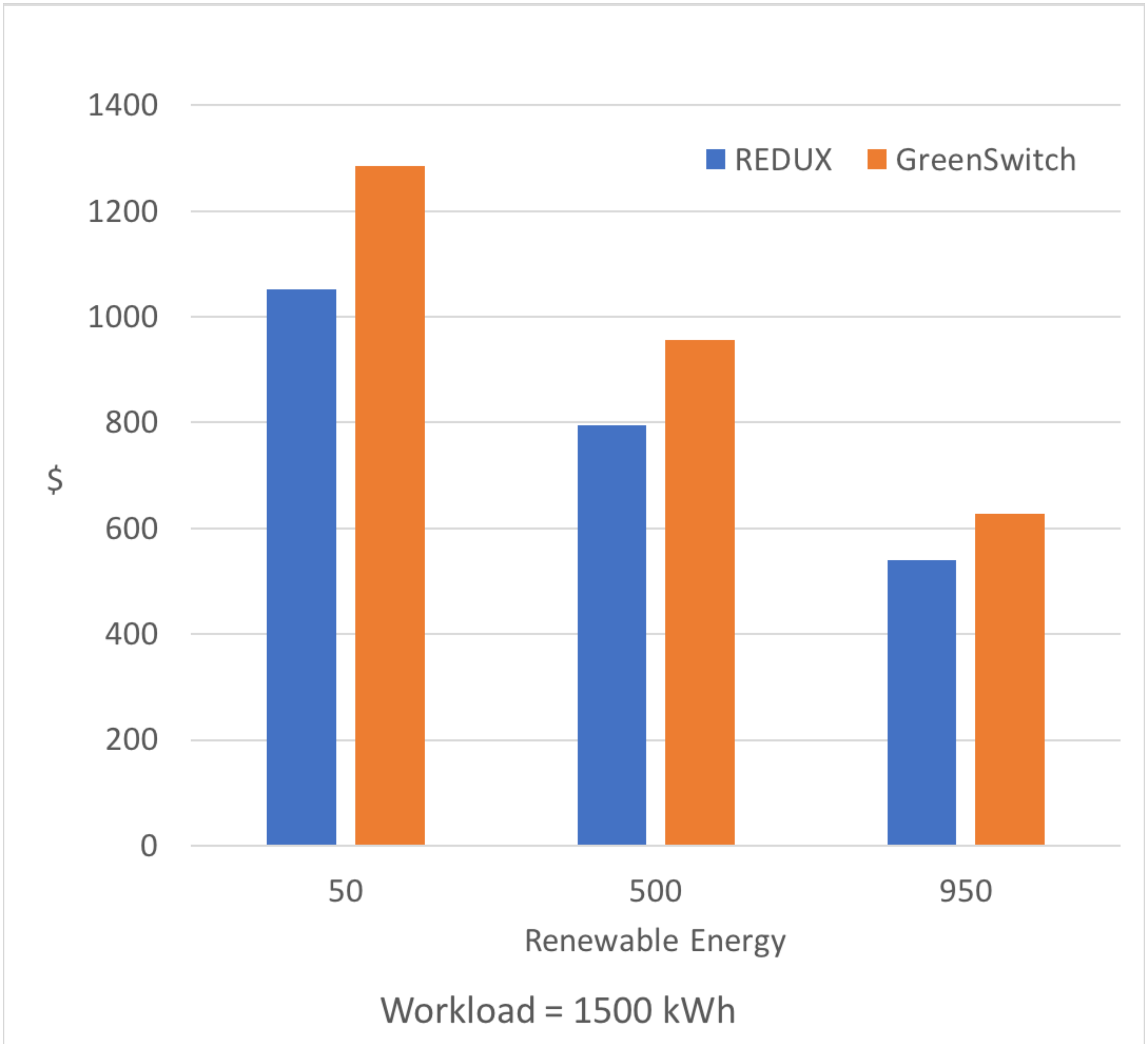

Figure 3.6: The impacts of renewable energy supply on energy cost of REDUX and GreenSwitch when the workload is set to 1500.

3.5 Summary

In this chapter, we presented a resource manager called *REDUX* to on-site renewable sources like solar coupled with a distributed UPS system to reduce energy cost of data centers. REDUX orchestrates UPS units to judiciously cope with the intermittency nature of renewable energy integrated with power grids. REDUX makes critical decisions on UPS charging and discharging with sufficient information on renewable-energy supply levels and time-dependent grid power prices. More specifically, REDUX makes prudent discharge of batteries to offer energy resources when renewable energy production is low or in the fluctuate condition. REDUX makes judicious decisions to charge batteries when renewable energy levels are high coupled with low loads.

One salient feature of REDUX is to effectively slash energy costs of data centers powered by both renewable and conventional energy. Compared with the prior solutions, REDUX delivers a prominent capability of mitigating average peak workload and boosting renewable-energy utilization. The experimental results demonstrate that REDUX paves the way for constructing modern data centers that are economically and environmentally friendly.

Chapter 4

REDUX2: Exploiting Renewable Energy and UPS Systems for Green Data Centers

In this chapter, we develop the workload-energy management system - *REDUX2* - an extended version of the REDUX energy resource manager (see Chapter 3) that incorporates a distributed UPS system tailored for renewable energy like solar and wind, and allocate renewable energy integrated with a power grid. More prosaically, We propose a interface-backbone layer framework designed unified energy efficiency management system called REDUX2, which properly allocate fluctuating renewable energy, UPS battery energy storage or grid power with dynamic price aiming to minimize the long-term electricity bill for the data center. With prediction of renewable energy supply, categorization of grid power price level and energy storage in the UPS devices, REDUX2 orchestrates workload distribution with heuristic algorithms which act as renewable energy smoothing, UPS device control, and high level control strategies, and make back-fills or defer decisions for the non-urgent jobs. Compared with the existing strategies, REDUX2 provides a comprehensive view for the energy efficiency data center and demonstrates a prominent capability of mitigating average peak workload and boosting renewable-energy utilization.

Due to intermittent nature of renewable energy and fluctuating grid energy price, advanced data centers often deploy distributed UPS systems with high efficiency, scalability, and reliability. In the second part of the dissertation study, we continue investigating the problem of energy management for data centers with renewable resources and energy storage. In our pilot project [?] [53] articulated in Chapter 3, we have demonstrated that the REDUX system save electricity bills by judiciously managing both inexhaustible on-site renewable energy and inexpensive stored energy from power grid for a typical data center. The overarching goal of REDUX2 is to curtail energy cost while maintaining high performance in data centers to address two challenging issues: (1) the intermittent feature of renewable energy and (2) dynamically changing grid electricity price. REDUX2 is responsible for orchestrating daily computing workloads in data centers while cutting back energy costs through improved energy resource utilization. Four emerging trends below strongly motivate us to contrive the REDUX2 system.

The rest of this chapter is organized as follows. Section 4.1 describes the extended area from REDUX2 to REDUX and present the brief discussion of related work listed in the comparison table with REDUX2 project. The framework and models of REDUX2 are detailed in Section 4.2. The Details of the problem statement accompanied by vital models for the REDUX2 system is formulated in Section 4.3. Section 4.4 articulates the updated high-Level control algorithm at the heart of REDUX2. Section 4.5 describes in detail the experimental configurations and settings. Section 4.6 discusses the experimental results along with the related analysis. Finally, Section 4.7 concludes this chapter.

4.1 Extension from REDUX to REDUX2

REDUX2 system is significantly expanded from REDUX project with a new section of detailed problem statement, updated framework design with redesign of high-Level control algorithm, together with comprehensive new experimental results, newly plotted figures, newly drawn tables, and analysis of new experimental results.

More specifically,

- We proposed new energy source models including Grid Energy Model, Renewable Energy Model and UPS Energy Model. Then we redesigned a more unified framework and high-Level control algorithm according to the implementation of all above models.
- We introduced the Parallel Workloads Archive(PWA) - workload traces including job information. Then we implement a job-power profiling function to transform workloads into power demands measured in kW/h.

- We simulated a data center with a representative scale size by resembling a system configuration prescribed in the PWA dataset. Then we substantially extended the simulation results of the REDUX to measure the impact such as the UPS device capability and workload conditions on total cost.

REDUX2 is adept at smoothing power supplies with the three supply levels of renewable energy resources, namely, high, fluctuate, and low. By naturally scaling with the data center size and eliminating a potential single point of failure, per-server distributed batteries become an economical and secure solution for battery backup. REDUX2 also makes judicious decisions on UPS charging and discharging with vital information on renewable-energy supply levels and time-dependent grid power price. For instance, REDUX2 makes full use of UPS devices to render energy resources when renewable energy supply is inadequate or under a fluctuate condition.

One prominent feature of REDUX2 is that it orchestrates a desirable balance between renewable-energy utilization and data center performance. Importantly, REDUX2 conserves the energy cost of data centers through power resource management while offering reliable computing operations. Compared with existing strategies, REDUX2 demonstrates an exemplary capacity of mitigating average peak workload and boosting renewable-energy utilization.

4.1.1 Contributions of REDUX2

The main contributions of the REDUX2 project are summarized as follows:

- We design REDUX2, an interface-backbone layer framework designed unified energy efficiency management system, which properly allocate fluctuating renewable energy, UPS battery energy storage or grid power with dynamic price for computing servers according to workload requirements in data centers, aiming to minimize the long-term electricity bill for the data center.

- We design heuristic algorithms act as renewable energy smoothing, UPS device control, and high level control strategies. With provided prediction of renewable energy supply, categorization of grid power price level and energy storage in the UPS devices, REDUX2 orchestrates workload distribution and make back-fills or defer decisions for the non-urgent jobs.

- We evaluate REDUX2 using real-world workload traces and green energy data. Simulation results show that compared with the prior solutions, REDUX2 conserves the total energy cost anywhere between 10% to 25%. We demonstrate that REDUX2 paves the way for constructing modern data centers that are economically and environmentally friendly.

4.1.2 A Comparison Table

Table 4.1 summarizes the key discrepancies between the REDUX2 system described in this Chapter and the existing power management solutions for green data centers.

Table 4.1: Comparisons between REDUX2 and the existing power management techniques for data centers.

Power Management	**UPS Enabled**	**Renewable Energy**	**Management Regions**	**Grid Price Fluctuations**	**Unified Framework**
iSwitch [45]	×	√	√	×	×
ReUPS [48]	√	√	√	×	×
EcoPower [21]	×	√	×	√	×
GreenSwitch [32]	×	√	×	√	×
Geo-distributed [71]	√	√	×	×	×
Two-stage [42]	×	√	×	×	×
REDUX2 (This Study)	√	√	√	√	√

In what follows, we briefly introduce the related systems and solutions listed in Table 4.1, and compare each with our work in this chapter.

- iSwitch [45] did not considered any energy storage method like distributed UPS system, which will lead to waste of energy when renewable energy is sufficient or grid power price is cheap.

- ReUPS [48] Successfully integrated both renewable energy and a centralized UPS system, but ignores the fluctuation nature of grid power price, and lack of a hierarchical design of the framework.

- EcoPower [21] and GreenSwitch [32] noticed the variable grid energy price and introduced the dynamic management of energy supply and demand. However, they both lack a energy storage mechanism and a unified hierarchical framework design.

- Geo-distributed [71] focus on addressing the problem of energy management for geo-distributed data centers with renewable resources and energy storage, and aim to minimize the long-term operation cost by leveraging the spatiotemporal diversity of grid power price, water consumption and carbon emission. However, the intermittent nature of renewable energy and possible dynamic changing grid energy price was ignored.

- Two-stage [42] proposed a two-stage stochastic program, and developed a mathematical optimization model for energy-efficient while ensuring the desired level of renewable energy utilization and quality of service. Whereas both variable grid energy price and energy storage method were missing.

4.2 Framework Design

We start this section by depicting the REDUX2 system framework (see Section 4.2.1), which offers a high-level design from the perspective of power management in data centers. Next, we introduce the interface layer of REDUX2 system and shed light on the responsibilities of the core modules in REDUX2 (see Section 4.2.3).

4.2.1 The REDUX2 Platform

REDUX2 is a management system orchestrating UPS units and renewable energy for computing servers according to workload requirements in data centers. In what follows, we

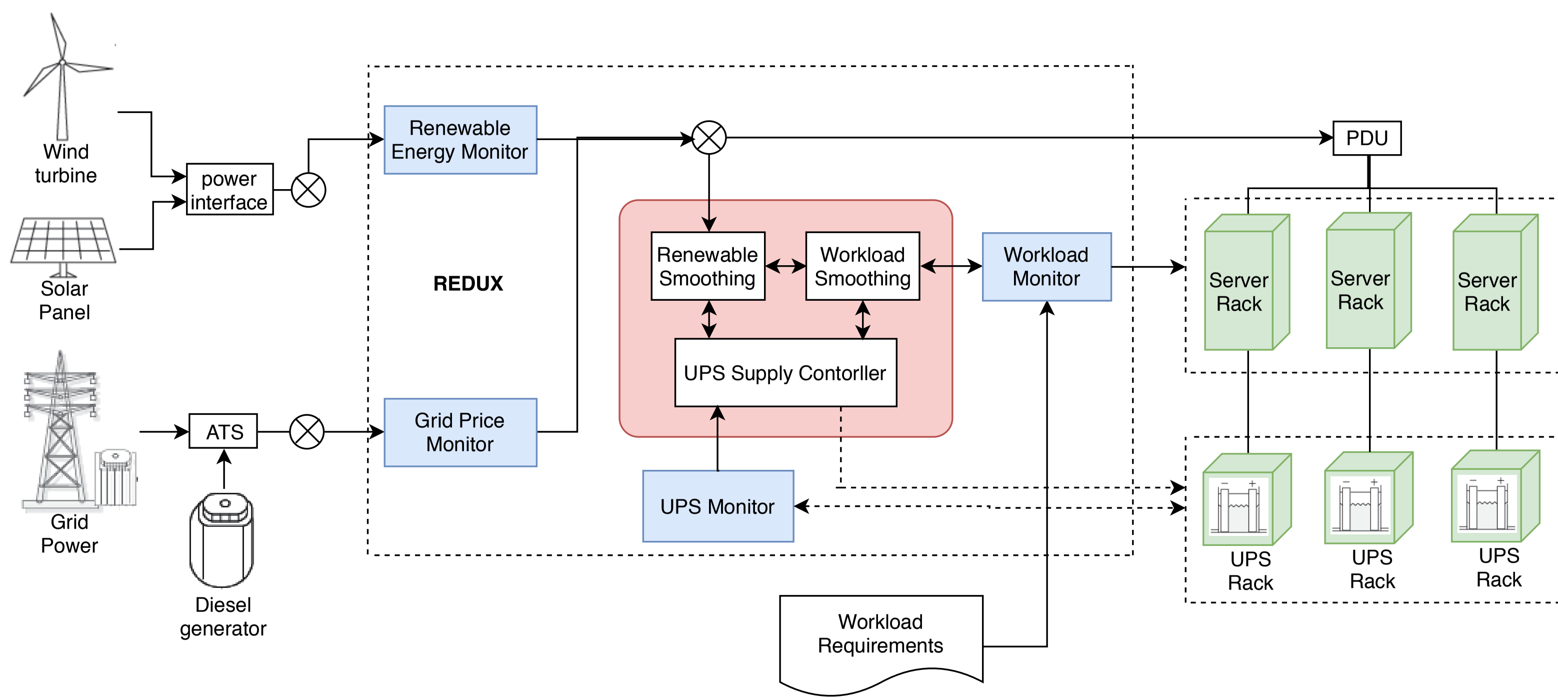

Figure 4.1: The framework of the *REDUX2* system. REDUX2 orchestrates renewable energy, UPS units, and server racks according to workload requirements.

outline the correlations between REDUX2 and the external entities such as renewable and grid energy supply, followed by the descriptions of the primary modules in REDUX2.

Fig. 4.1 plots the framework design of the REDUX2 system, where renewable energy (e.g., wind and solar), grid power, and diesel generators are seamlessly integrated (see the left-hand side of Fig. 4.1). REDUX2 employs a distributed UPS system, where a UPS device is attached to each server rack (see the right-hand side of Fig. 4.1).

REDUX2 persistently communicates with the five external entities, namely, renewable energy supplies, a power grid with diesel generator as backup, distributed UPS units, server racks and workload requirements. A wide range of energy supplies may include, but not limited to wind or solar energy, electrical grid and diesel generators. To alleviate the counter-varying load effect from the perspective of power grids, smart on-site diesel generators are installed in data centers as an energy buffering mechanism. Such an energy buffer aims at smoothing dynamically changing renewable energy to minimize the adverse effect of counter-varying load.

REDUX2 consists of two layers : an interface layer (see all light-blue colored modules in Fig. 4.1) and a backbone layer (see the modules inside the light-red colored box in Fig. 4.1), which are detailed in the following two subsections.

4.2.2 The Interface of REDUX2

The interface layer is comprised of four modules, including a renewable energy monitor, a grid price monitor, a UPS monitor, and a workload monitor. Seamlessly integrating these modules, REDUX2 keep track of (1) the necessary energy supply trace data from on-site renewable device or off-site grid networks, (2) an amount of workload to initiated in the current time slot or deferred from the previous time slot, and (3) energy storage amount in each UPS rack. On the other hand, we also need these monitors to update the necessary states of the specific data. For example, the renewable energy monitor and the grid price monitor will determine the state of renewable energy supply state and grid price state, respectively. Then the backbone layer can continue on the energy management process; The workload monitor will also update the workload states before the workload smoothing method take charge of workload arrangement tasks.

Furthermore, we introduce a primitive prediction mechanism in all the modules except for the UPS monitor. By considering the fluctuating patterns of renewable energy supply or grid power price, and the status of previous workload conditions, this mechanism helps the backbone layer of REDUX2 to achieve rational and improved decisions on both energy supply and workload management. Although we make no assumption on statistical distribution of workloads, this prediction method on workloads can help the core module in backbone layer by smoothing the workloads that are dramatically changing between subsequent time slots. Please refer to Section 4.5 for details on this mechanism.

In a nutshell, these monitors collect trace data from all the external entities, which feed data into the modules residing in the backbone layer.

4.2.3 The Backbone of REDUX2

The three core modules in the backbone layer communicate with five external entities (i.e., renewable energy supplies, power grid, UPS units, server racks, workload requirements) through the four interface modules in the interface layer.

The backbone layer of REDUX2 contains three modules, namely the Renewable Smoothing module, the Workload Smoothing module, and the UPS storage controller module. The algorithms of these three modules are presented in the Previous Section 3.3. The functionalities of the three core modules of REDUX2 are summarized as follows:

- **Renewable Smoothing.** The renewable smoothing module (see Algorithm 2 in Section 3.3.) is responsible for effectively dealing with cases where renewable energy fluctuates due to an environmental status. This module copes with the fluctuating renewable energy through a distributed UPS system.

- **Workload Smoothing** The workload smoothing module is focused on treating overload conditions, in which workload exceeds the underlying data center's computing and energy capacity, or cases when deferral is a better choice, depends on prediction. This module is invoked in Steps 3 and 10 of Algorithm 4 in Section 3.3.

- **UPS storage controller** The responsibility of the UPS storage controller module is two-fold. First, this module keeps track of the status of all UPS units in a data center. Second, the module is in control of making battery charging and discharging decisions. Refer to Algorithm 3 in Section 3.3 for the design of the UPS-usage controller module.

4.3 Problem Statement and Models

In this section, we formally formulate the problem statement of this project by building a time-slotted model. To facilitate the presentation of the model, the notations used throughout this section are summarized in Table 4.2.

Table 4.2: Symbol and Annotation

Symbol	**Annotation**
$E(t)$	total energy needed in time slot t
$E^W(t)$, $E^S(t)$	energy supply of wind and solar in time slot t, respectively
$s(t)$	the solar irradiance trace during time slot t
$v(t)$	the average wind speed trace during time slot t
$E^G(t)$	energy supply needed from power grid in time slot t
$E^{UPS}(t)$	net energy supply from UPS device in time slot t
$E^{UPS}_{storage}(t)$	currently stored energy in UPS device
E^{UPS}_{min}, E^{UPS}_{max}	lower and higher bound of UPS device storage capacity
P^W,P^S	price of wind and solar energy, respectively
$P^G(t)$	price of grid energy at time slot t, respectively
$W(t)$	the total workload need to be executed during time slot t
$w(t)$	the workload generated by users in time slot t
$w_d(t)$	the remaining and undeferable workload from time slot t
$C(t)$	total cost per unit of energy in time slot t

4.3.1 Modeling Energy Sources

In this subsection, we model multifaceted energy sources including renewable energy, grid power, and UPS systems. Given energy supply sources, we denote the total energy supply of the current time slot as $E(t)$. In a general scenario, $E(t)$ is derived from (1) grid energy drawn from off-site power grid $E^G(t)$, (2) renewable energy generated from on-site facilities $E^{ren}(t)$, and (3) energy previously stored in UPS devices $E^{UPS}(t)$. Thus, total energy supply $E(t)$ is expressed as

$$E(t) = E^G(t) \ + \ E^{ren}(t) \ + \ E^{UPS}(t)$$

In what follows, we construct the models of these three energy sources.

4.3.2 Grid Energy Model

The grid energy cost of a data center is determined by the product of two factors, namely, (1) the grid energy price and (2) the amount of grid energy consumed by the data center. Recall that the main goal of *REDUX2* system is to aggressively utilize renewable energy by employing UPS devices with high reliable condition. The amount of grid energy utilized in the current time slot, $E^G(t)$, will then reliant on (1) workload status of the current time slot, (2) price level of grid energy, and (3) the available amount of renewable energy combined with energy stored in UPS devices. Let $P(t)$ be the grid energy price during time slot t, which is a random variable determined by the electricity market [21]. We assume that the time slot (e.g., one hour) is small enough so that $P(t)$ remains stable during each time slot, unlike the system developed by Deng *et al.* [21]. We also assume the data center is supplied by a power grid source with infinite availability compares to the energy demand of the data center, meaning that there is no maximum bound to $E^G(t)$. Hence, the necessary off-site grid energy cost in time slot t can simply written as $P(t) \cdot E^G(t)$.

4.3.3 Renewable Energy Model

In order to model renewable energy $E^{ren}(t)$ in time slot t, we consider solar and wind energy - two popular renewable energy sources. Let $E^W(t)$ and $E^S(t)$ be the wind and solar energy amount generated in time slot t. We express $E^{ren}(t)$ as

$$E^{ren}(t) = E^W(t) \ + \ E^S(t).$$

It is reasonable to assume solar irradiance and wind speed remain stable within any time slot if the time window is small (e.g., 1 hour). Following renewable energy models developed by Richard, Komp, and Gipe [56] [29], we construct the solar energy model as:

$$E^S(t) = \alpha \ \cdot \ As(t) \ \cdot \ l,$$

where $s(t)$ is the solar irradiance trace during time slot t, A is the total active irradiance area of all solar panels, α represents the solar-to-electricity conversion efficiency rate, and l indicates the duration of a time slot. It is mindful that the total active irradiance area A is unlikely to change during the investigated time horizon; such an assumption is validated in the prior studies [56] [29].

The wind energy model is written as:

$$E^W(t) = \beta \ \cdot \ (1/2)Bpv^3(t) \ \cdot \ l,$$

where $v(t)$ is the average wind speed trace during time slot t, B is the total rotor area of all wind turbines, β represents the wind-to-electricity conversion efficiency rate, ρ is the air density, and l is again the duration of a time slot. Similar to active irradiance area A, rotor area of all wind turbines B remains unchanged during the course of the studied time horizon.

It is worth to notice that in both of these two models, the only changing variable according to different time slot t are only solar irradiance $s(t)$ and average wind speed $v(t)$.

Which means all other variables will act as constants as long as we focus our research in an appropriate the time horizon, or determined by experiment settings such as the duration of time slot l.

After modeling the above two renewable energy sources, we denote the unit prices of solar and wind energy as P^S and P^W, respectively. Unlike the grid power price, the price of renewable energy includes one-time capital expense and long-term operational costs. Renewable energy price is calculated by averaging the total cost of installment and operation over the total amount of energy generated during an entire available period [**?**]. Modeling and deriving renewable energy prices is beyond the scope of our research. Therefore, we set these prices as a constant in our experiment configurations. In the not-too-distant future, we will delve in the impact of dynamically changing renewable energy price on our system performance.

4.3.4 UPS Energy Model

Traditional UPS devices merely act as transitional fail-over mechanisms between utility and captive sources such as diesel generators. We pay particular heed to a powerful distributed UPS system that not only powers up a data center when grid power is interrupted, but also is capable of storing a noticeable amount of energy when gird energy price is cheap or renewable energy is abundant. Such a distributed UPS system provides power supply when grid energy price is high or renewable supply becomes unreliable.

Throughout the lifespan of battery, two affecting factors determine battery failure rates, namely, (1) the depth of discharge (DOD) and (2) discharge rates. DOD is gauged as a percentage of energy storage; a discharge rate is represented as discharge current specified as a C-rate. We assume the C-rate of UPS batteries during each time slot is fixed at a certain level. Then, we configure the DOD lower bound of UPS devices (e.g., 20%) to minimize the battery failure rate. Such a DOD lower bound is a secured level to boost battery reliability by tackling the running-down problem induced by power outages. On the other hand, it is mandatory to have batteries maintain a minimum energy storage amount in UPS devices for the reliability purpose.

To stay focused on this part of the dissertation research, we now follow the reasonable assumption imposed by Urgaonkar *et al.* [63], stating that (1) batteries lost no power either when recharging and discharging, and (2) the stored energy will not decreases except discharged (i.e., batteries are not "leaky"). So the batteries now have a finite and unchanged capacity of energy storage E_{max}^{UPS}. For the purpose of extending the life-span and improve the reliability of batteries, in each time slot we restrict the use of batteries in a way that one can either recharge or discharge the batteries, or take no action on batteries, but not together. In other words, the batteries can not be recharged and discharged at the same time (see, for example *et al.* [21]). So we let $E^{UPS}(t)$ denote the net energy supply offered by UPS devices during time slot t, with positive value as discharge and negative value as recharge. Finally, to maintain the function as backup power, we set E_{min}^{UPS} as the minimum energy storage requirement in the UPS device to a certain portion (20%, for example). Through the above denote, we have

$$E_{min}^{UPS} - E_{max}^{UPS} \leq E^{UPS}(t) \leq E_{max}^{UPS} - E_{min}^{UPS}.$$

Though with previous "not leaky" assumption, there are still other issues worth notice when designing UPS system. For example, an issue called double-conversion loss which incur with frequent discharge or recharge was addressed by Sato *et al.* [58]. By designing a three or multi-level circuits topology, they conclude that a high-density redundant designed UPS system will meet the high-efficiency and reliability requirements with the utmost critical system. Again to focus on the realm of this study, we stay with assumptions that all UPS devices are safe from inrush current phenomenon and has minor or no double-conversion loss.

As the cost of recharged energy is incorporated in the grid and renewable power cost, the battery power cost is simplified to a one-time capital expenditure only. Suppose a battery that is worth Q dollars has a finite non-negative integer N of recharge/discharge times, then we model the cost of UPS devices as $P^{UPS} = Q/N$ during the device lifespan. Though such cost becomes negligible compared to gird or renewable energy cost, we do consider the UPS cost in our experiments.

4.3.5 Modeling Workloads

Prior to the development of a workload model, we first measure the total computational workload $w(t)$ generated from data center users in time slot t. Since job-power profiles are missing in the job traces, we apply the power-profile function to derive workload $w(t)$ of each job during time slot t. Power demand of the individual job is then calculated by the job's allocated number of processors and running time during time slot t, followed by measuring power demands in terms of kWh. The details of the job power profile function can be found in section 4.5.

Workload $w(t)$ is now gauged in terms of power demand as kWh. All these workloads vary in a wide range, taking values from a non-negative finite set, with a maximum value denoted as w_{max}. Thus, we have

$$0 \leq w(t) \leq w_{max}.$$

To simplify the design while meeting our experimental needs, there is no necessity to keep track of the underling probability distribution or statistical characterization of $w(t)$. Nevertheless, we conclude that the overall workload generated during a time horizon T should match the total energy consumed by a data center. We formally express this trend as:

$$\sum_{t=1}^{T} E(t) = \sum_{t=1}^{T} w(t). \tag{4.1}$$

Because the quality of service (QoS) level offered a data center highly depends on whether the data center can handle the workloads in a timely manner, it is nature to build a mechanism to deal with cases in which heavy workload in a certain time slot exceeds the data center's computing capacity. To cope with such heavy workload conditions, REDUX2 smartly defers computing loads to the next subsequent time slot. It is worth noting that such a computing deferral strategy is applicable to the other three scenarios, namely, (1) high grid power price and (2) outage of renewable energy.

Following the assumption imposed by Goiri *et al.* [32] and Urgaonkar *et al.* [63], it is arguably true that a data center's energy demand is either satisfied in the current slot or delayed to the next time slot due to the above three scenarios. In other words, our scheduling policy makes jobs wait for a limited amount of time (e.g., one time slot) to acquire an improved QoS performance (see also a similar design philosophy by Zaharia *et al.* [69]). Now, we extend Eq. 4.1 to model workload deferral cases below.

$$\forall t \in T : \sum_{t'=1}^{t'=t} E(t') \leq \sum_{t'=1}^{t'=t} w(t'), \tag{4.2}$$

Eq. 4.2 implies the accumulated energy demand from workloads may greater or equal than the total energy supply. Eq. 4.2 is reasonable, because in order to maintain a relatively high QoS level for any time slot t', some heavy computing loads are deferred to the next time slot by the delay scheduling method.

In delay scheduling, the system strives to finish deferred workloads in time slot $t+1$ by giving a high priority to the deferred workloads. In doing so, deferred tasks are unlikely to be further delayed to avoid the starvation problem. Let $w_d(t)$ be delayed workload in time slot t. We set deferred workload as undeferable in slot $t+1$, placing the delayed workload at the beginning of an FIFO queue in the next time slot. We denote $W(t)$ as the total amount workload to be executed by the data center in time slot t. Hence, we express $W(t)$ as

$$W(t) = w(t) \ + \ w_d(t-1).$$

In a nutshell, the priority of delayed $w_b(t-1)$ is higher than that of the newly submitted workload $w(t)$. Ideally, jobs in high-priority $w_b(t-1)$ should be accomplished before executing the first job in $w(t)$.

Recall that $w(t)$ as well as $W(t)$ are measured in terms of energy consumption in the unit of kWh; $E(t)$ is defined as the total energy supply consumed by a data center handling all the workloads during time slot t. Given time slot t, we express the relationship between consumption $W(t)$ and supply $E(t)$ as

$$W(t) = E(t).$$

4.3.6 Modeling Total Cost

Now, we are positioned at construct a total-cost-optimization model that incorporates with all the constrains presented in Sections 4.3.1 and 4.3.5.

We define $C(t)$ as the total cost of energy supply in time slot t. Cost $C(t)$ is derived from grid-power cost, renewable energy cost, and the UPS-device cost. Thus, cost $C(t)$ can be calculated as

$$C(t) = P^{G}(t) \cdot E^{G}(t) \ + \ P^{ren}(t) \cdot E^{ren}(t) \ + \ P^{UPS} \cdot E^{UPS}(t).$$

The overall optimization goal of our system is to minimize the cumulative $C(t)$ through time horizon T under all the aforementioned constrains. We formally describe the goal as follows

$$Min \sum_{t=1}^{T} [P^G(t) \ \cdot \ E^G(t) \ + \ P^W(t) \ \cdot \ E^W(t) \ + P^{UPS} \ \cdot \ E^{UPS}(t)]$$

$$s.t. \quad W(t) = E(t)$$

$$E^G(t) \ + \ E^{ren}(t) \ + \ E^{UPS}(t) = E(t)$$

$$W(t) = w(t) \ + \ w_b(t-1)$$

$$0 \leq w(t) \leq w_{max}$$

$$0 \leq E^G(t) \leq E^G_{max}$$

$$0 \leq E^{ren}(t) \leq E^{ren}_{max}$$

$$E^{UPS}_{min} - E^{UPS}_{max} \leq E^{UPS}(t) \leq E^{UPS}_{max} - E^{UPS}_{min}.$$

It is clear that the above optimization goal has trade-off between power cost, renewable energy reflected in the constraints. By solving the preceding problem, we can obtain the corresponding power management and load scheduling strategies to optimize the system.

- Power Management Strategy: For a data-center with on-site renewable energy facilities and fluctuating grid power price, REDUX2 can determine the amount of power supplied by the grid at time slot t $E^G(t)$, and the amount of power recharged to the UPS devices $E^{UPS}(t)$.

- Load Scheduling Strategy: The workload smoothing module can smoothing the workload generated by users and decide whether to dispatch more or less the workload $W(t)$ to the data-center.

4.4 The Updated High-Level Control Algorithm

In this section, we present the updated high-level control algorithm in Section. We take a hierarchical design approach to the development of REDUX, in which a high-level control algorithm coordinates all the three core modules in the backbone layer (see the backbone layer in Section 4.2.3).

Algorithm 4 The high-level control algorithm in REDUX.

Require:
Renewable energy trace data $E^W(t)$ and $E^S(t)$
UPS energy storage $E^{UPS}(t)$ from previous time slot
Generated workload $W(t)$ and deferred workload $w_d(t)$
Price of all energy source P^W, P^S, P^{UPS} and $P^G(t)$
Ensure:
REDUX Cost $C(t)$ of All time slots
1: $t = 0$
2: **while** $t <= T$ **do**
3: $\quad workload\_shaving()$;
4: $\quad workload\_smoothing()$;
5: $\quad$ determine renewable energy supply state $S_{RE} = STA, FLU$ *or* OTA;
6: $\quad$ **if** $S_{RE} = FLU$ **then**
7: $\qquad ren\_supply\_smooth()$;
8: $\quad$ **end if**
9: $\quad$ update over-peak workload threshold then determine workload over-peak states $S_W = OVR$ *or* OFP;
10: $\quad$ **if** $(S_w = OVR)$ **then**
11: $\qquad workload\_shaving()$;
12: $\quad$ **end if**
13: $\quad ups\_storage\_control()$;
14: $\quad E_t^G = W_t - E_t^{RE} - E_t^{UPS}$;
15: $\quad calculate\_energy\_cost()$;
16: $\quad t++$
17: **end while**
18: **return** REDUX cost

Algorithm 4 depicts the pseudo-code of the high-Level control algorithm in REDUX, where the inputs include (1) all energy supply data, namely, wind energy $E^W(t)$, solar energy $E^S(t)$ and energy storage amount in UPS device $E^{UPS}(t)$, (2) workload data $W(t)$ generated by users and previous deferred workload $w_d(t)$ and (3) grid energy price $P^G(t)$ as variable and P^W, P^S, P^{UPS} as constants.

The high-level controller kicks off the power management task with the workload-shaving procedure (see $workload\_shaving()$; in Line 3). The purpose of this first workload shaving is the overloading of a data center with respect to power supplies and computing capacity.

More specifically, if the submitted workload is below the overloading threshold, the pending jobs will immediately be processed in the current time slot. Otherwise, these jobs will be delay scheduled by the high-level controller. Jobs resulting in overload should be scheduled and processed in the subsequent time slots. It is noteworthy that scheduling such jobs under the overloading condition imposes significant impact on the data center's overall performance metrics such as throughput and response time. The investigation of scheduling policies in the overloading case is out of scope of this REDUX2 development. We intend to address this concern as a future research direction.

The procedure of workload smoothing in Line 4 (i.e., $workload\_smoothing()$) aims to mitigate possible dramatic workload changes between subsequent time slots. Prior to kicking in this procedure, an exponential windowed average result of a previous workload should be maintained as a threshold. If the current workload is greater than the threshold (i.e., exponential window average), the workload smoothing method will simply update the current workload level with a smoothed average result. If the current workload is smaller than the threshold, the procedure will increase the current workload by handling deferred workload if there is any. In this case, the current workload level is increased until the level hits the threshold.

In the high-level control algorithm 4, the next task after workload smoothing is to determine the renewable-energy state S_{RE}(see Line 5), which has three candidate cases, namely, stable (i.e., STA), fluctuate (i.e., FLU) and outage (i.e., OTA). To categorize these states, if the renewable supply is at the minimum level, the renewable states will be defined as OTA (see the purple dots in Fig. 4.2). If the renewable-energy supply is beyond the maximum level, the renewable states will be referred to as STA (see the gray dots Fig. 4.2). For the other cases, we compare the renewable-energy supplies with their previous exponential windowed average. If the change in the renewable supply from the previous time slots is in a "stable" interval, we envision this current renewable supply as STA; otherwise, the state should be categorized as FLU (values unannotated in Fig. 4.2).

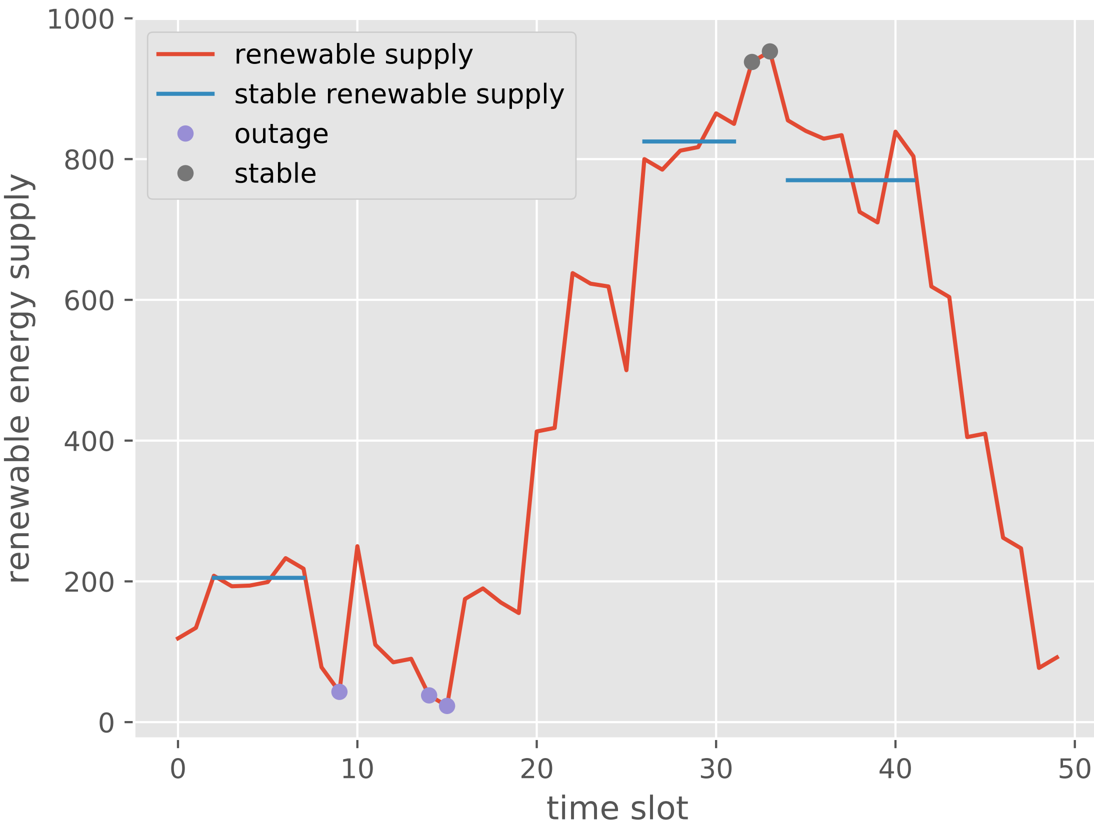

Figure 4.2: An example for the categorization of renewable energy supplies. The unannotated values are defined as fluctuate states.

After REDUX determines the state of the current renewable supply, REDUX further manages workload in accordance to the state. More specifically, if the renewable energy is in the fluctuation state, the renewable-energy-supply smoothing policy is invoked in Line 7. The smoothing procedure is governed by the current renewable supply data, a dynamically updating stable supply level, and UPS energy storage condition. Please refer to Section 4.4 for the details of this procedure.

After the fluctuating renewable supply is smoothed, the renewable and UPS energy supplies are finalized. To further improve the performance of REDUX with these measures in place, REDUX dynamically updates an over-peak workload threshold. Such an update is accomplished by providing the information pertaining to the current workload amount, renewable and UPS energy supply and grid energy price. From this threshold, REDUX can determine whether or not a second workload shaving is necessary by the workload's over-peak state S_W, either over-peak (i.e., OVR) or off-peak (i.e., OFP) (see Line 9). If the workload exceeds the over-peak threshold, then the workload-shaving procedure is kicked in to defer the extra load to the next time slot (see Line 11). Again, such second workload shaving makes a positive impact on the data center's overall performance as the first shaving.

The $ups\_storage\_control()$ procedure (see Line 13) is responsible for protecting UPS units against frequent charging and discharging, thereby prolonging lifetime of the distributed UPS system. This procedure determines a UPS energy level, under which the UPS units should be protected. Such levels largely depend on current workload conditions, grid energy price as well as renewable energy capacity. The functionality of $ups\_storage\_control()$ is two-fold: (1) updating the UPS status information and (2) making UPS discharging and charging decisions.

Recall that vital factors include workload W_t, renewable energy supply E_t^{RE} and net energy supply from UPS devices E_t^{UPS}. After all these factors are determined, REDUX computes the energy supply obtained from the grid network (see Line 14). The $calculate\_energy\_cost()$ function in Line 15 calculates the total energy cost occurred in time slot t. This function also enables us to quantitatively compare REDUX against other exiting power management strategies.

4.5 Experimental Settings

In this section, we extend experimental settings from the previous chapter: (1) We introduced a real-world workload trace named Parallel Workloads Archive (PWA). To further measure the power demanding of each job, we then implement the job-power profiling function to transform workloads info into power demands measured in kW/h. (2) We updated the setting of simulated data center with a representative scale size by resembling a system configuration prescribed in the PWA data set.

Before presenting experimental results in the next section, let us first briefly re-describe the experimental settings from the pilot REDUX project in the following Section 4.5.1. Then we present the power demand side as the extension from the REDUX to REDUX2 system (see Section 4.5.2). Finally, we demonstrate a brief comparison and analysis of the distribution pattern with or without the implementation of workload smoothing module in Section 4.5.3.

4.5.1 Power Supply

Same as the REDUX project, we test REDUX2 and its competitors using the electricity price of a real-world power grid in New York state during a period of three months from June 1 to August 31, 2006. Fig. 4.3 shows the variable nature of grid energy price we discussed before. The maximum and minimum prices in the trace are 1.82 and 0.26 $\$/Kwh$, respectively.

We still retrieve the solar supply data from the National Renewable Energy Laboratory's database of New York state during the three-month period also from June 1 to August 31, 2006. The solar supply data measured in KWh is plotted in Fig. 4.4. The solar and wind power price is set as constant at $0.09/Kwh and $0.07/Kwh, respectively.

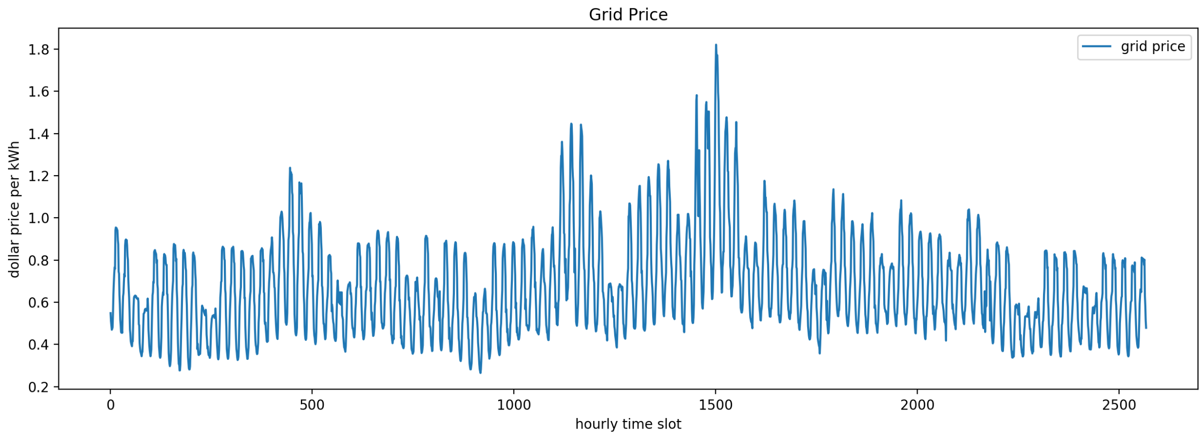

Figure 4.3: The electricity price of a power grid in the New York state during the period of three month ranging from June 1 to August 31, 2006. Unit: $\times 100\$/\mathrm{MWh}$

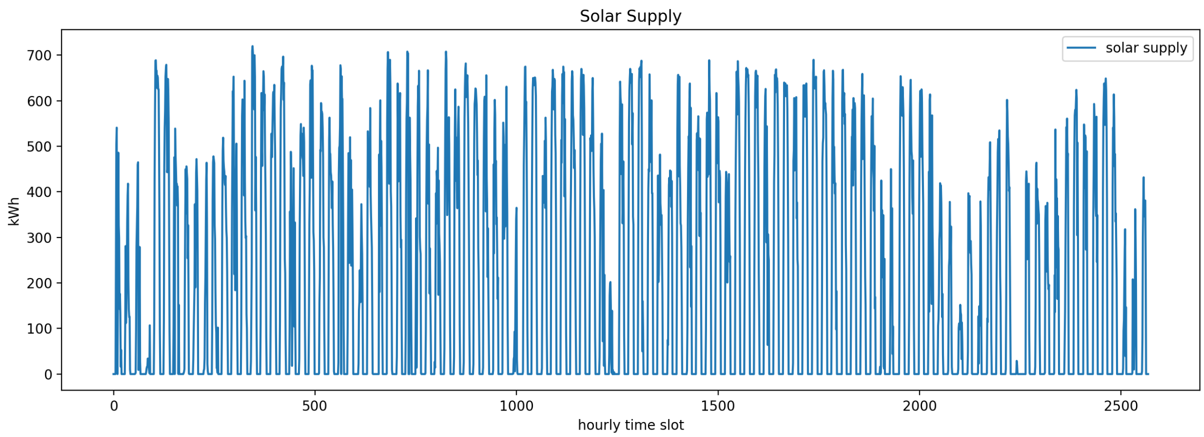

Figure 4.4: The solar supply of New York state during the three-months period ranging from June 1 to August 31, 2006.

As an update from REDUX, in REDUX2 project we introduce the distributed UPS system - 6 per rack (192 per cluster computing system) - and scaled up by a factor of 10 for the entire large data center. And the energy cost for recharging and discharging batteries is calculated as \$0.07/kWh from the UPS model we use.

4.5.2 Power Demands

As an important extension of experimental settings from REDUX to REDUX2, we conduct extensive experiments using real-world workloads recorded in the Parallel Workloads Archive (a.k.a., PWA) [25] [26] – a popular repository of parallel computing workloads. Traces archived in PWA are job logs collected from large-scale parallel computing systems. Each job information includes job ID, running time, the number of allocated processors, used memory resources and the like. The specific trace evaluated in this part of the dissertation study is LLNL Atlas, which contains 8-month (Nov 2006 to Jun 2007) accounting records from a large Atlas Linux cluster. The Atlas cluster is comprised of 1152 nodes in 32 server racks (36 nodes per rack), each of these nodes contains eight AMD Opteron processors clocked at 2.4 GHz and 16 GB of memory. The total number of processors in the cluster is 9216.

All the job trace data acquired from PWA represent the behavior of a single parallel system, which is relatively small compared with large-scale computing platforms housed in data centers. We expand the traces along the following two directions to address this concern. First, we scale up the size of the LLNL Atlas platform by a factor of 5 to mimic a large system that is comprised of 46,080 processors (5×9216) with a homogeneous configuration. Second, we scale up by a factor of five the number of required processors for each job to resemble the size of a data center.

The time slot length is set to one hour in our experiment settings to (1) accommodate large batching jobs and (2) match the measurement unit of energy consumption as kWh. Therefore, if a large job started during any time slot requests an execution time exceeding one hour, the REDUX system will (1) estimate power demand of the job through the current time slot (maximum one hour), at the beginning of the time slot; and (2) calculate the actual power consumption of the job during this time slot, at the end of the time slot.

We implement a job power profiling module to render a power profile for each job executed during time slot t. The power profiling module takes the number of allocated processors and running time during slot t of each job as inputs. Following a guideline documented in [68] and [64], we set the power demand of a node to a value anywhere between 20 to 60 $Watt$ using a normal distribution; This setting will lead to the following two deductions: (1) the profiling module then assigns the power demand of each processor in a range between 2.5 and $7.5W$ following a normal distribution; and more importantly (2) as the number of nodes of the entire data center was set as 46080, the maximum execution capacity of the whole data center in terms of energy consumption can then be calculated as $46080 \times 60Watt \times 3600Secs = 9.95 \times 10^9 J = 2764.8kWh$. The power profile of each job running in slot t is then multiplied by the power dissipation of all the allocated processors and the job running time. The final step of the profiling module is to accumulate all the jobs' power demands during slot t, thereby converting job trace data into power demand measured in kWh in slot t.

4.5.3 Predicting from Smoothed Workload

Recall that in this research we introduce a simple yet effective prediction model, where an exponential windowed average is continuously updated. These predicted results serve as a threshold or a baseline to guide the relative smoothing method carried out in both workload smoothing and renewable energy smoothing. With each updated threshold on hand, the REDUX will then determine whether shave and defer the extra workload form the current time slot to the next. Fig. 4.5 depicts an overall view of the smoothing method, which makes predictions on workload prior to any workload shaving. Intuitively, workload distribution after the exponential windowed average (see the red dots in Fig. 4.5) is more convergent to the total average. When we sort and plot both these data traces, this averaging method offers a flat sloped curve (i.e., see the red curve). We draw similar observations when we apply this prediction method to grid price.

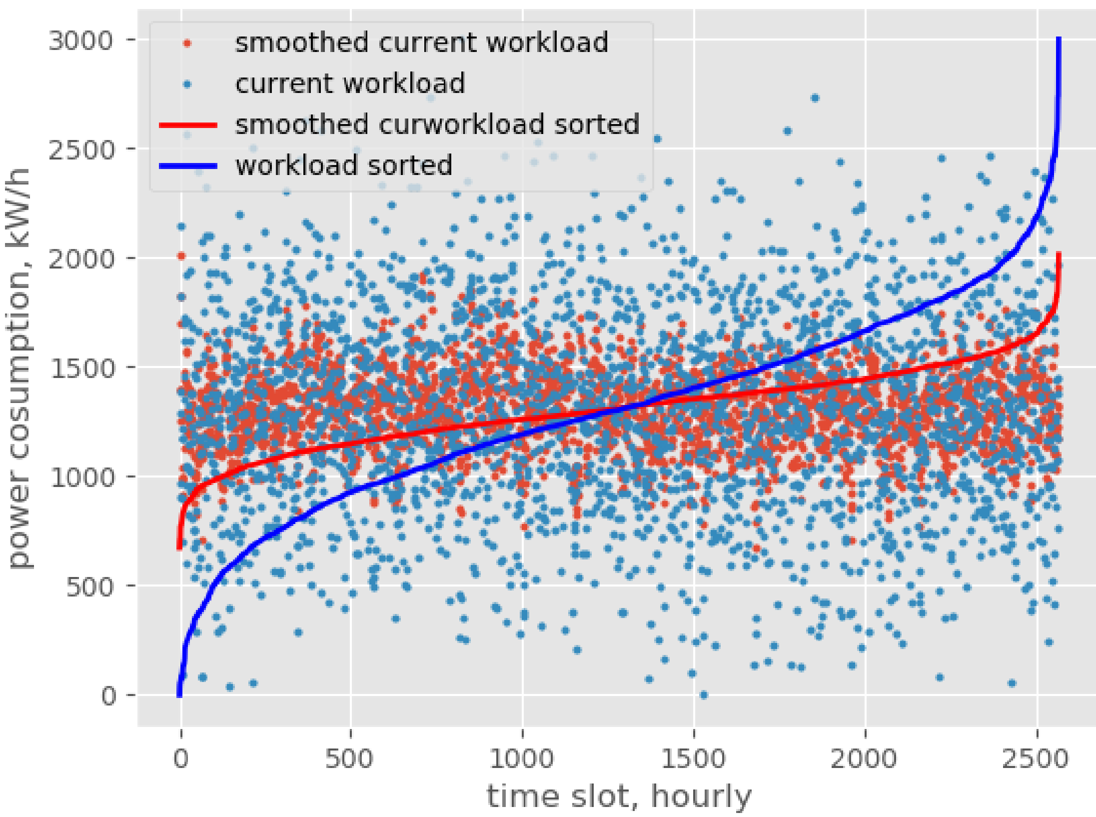

Figure 4.5: The predicted workload, serving as a smoothing threshold, shows convergence to an average workload

4.6 Results and Analysis of REDUX2

In the previous chapter (see section 3.4, we evaluated the performance of REDUX2 driven by real-world renewable energy and power grid price data and compared the accumulated energy cost of REDUX with a baseline solution called *noREDUX*, as well as two existing power management techniques - *GreenSlot* [30] and *ReUPS* [48].

We not only extend performance measuring methods from the previous chapter but also we substantially extend the simulations of the REDUX to measure the impact such as the UPS device capability and workload conditions on total cost.

4.6.1 REDUX vs noREDUX

In the first experiments, we compare REDUX with noREDUX - a baseline power manager where renewable energy resources and UPS units are not incorporated. More specifically, noREDUX is a scheme where renewable energy is just passively utilized as it generated, and the energy stored in UPS devices (if there exists) is not actively utilized.

Fig. 4.6 shows a few time slots are outliers, where total energy cost for a REDUX-enabled data center is higher than a noREDUX data center. This may because the deferred workloads from previous time slot has to be processed while noREDUX system do not have this mechanism. Nevertheless, REDUX judiciously cuts back, in a majority number of cases, the total cost of a data center.

Fig. 4.7 again reveals the superiority of the REDUX system against noREDUX systems by an accumulative metric. Regardless of REDUX or noREDUX, the cumulative energy cost linearly increases with time. The results show that REDUX significantly cut back the energy of the data center governed by noREDUX. For example, REDUX is able to reduce the energy cost of noREDUX by 24.5% by the end of the first month; such cost savings becomes 27.9% and 33.2% by the end of the second month and third month, respectively.

The energy-cost savings are expected because REDUX orchestrates renewable energy resources and UPS units to conserve energy cost when the electricity price of the power grid is expensive. By defer some workload to the following time slot while still maintain a desirable QoS, a data center with REDUX can then smartly control the arrangement of workload and distribution of energy resources. For example, there are multiple sharp peaks (e.g., see 480, 1200, and 1500 hours in Fig. 4.3) in the grid price trace data. REDUX makes

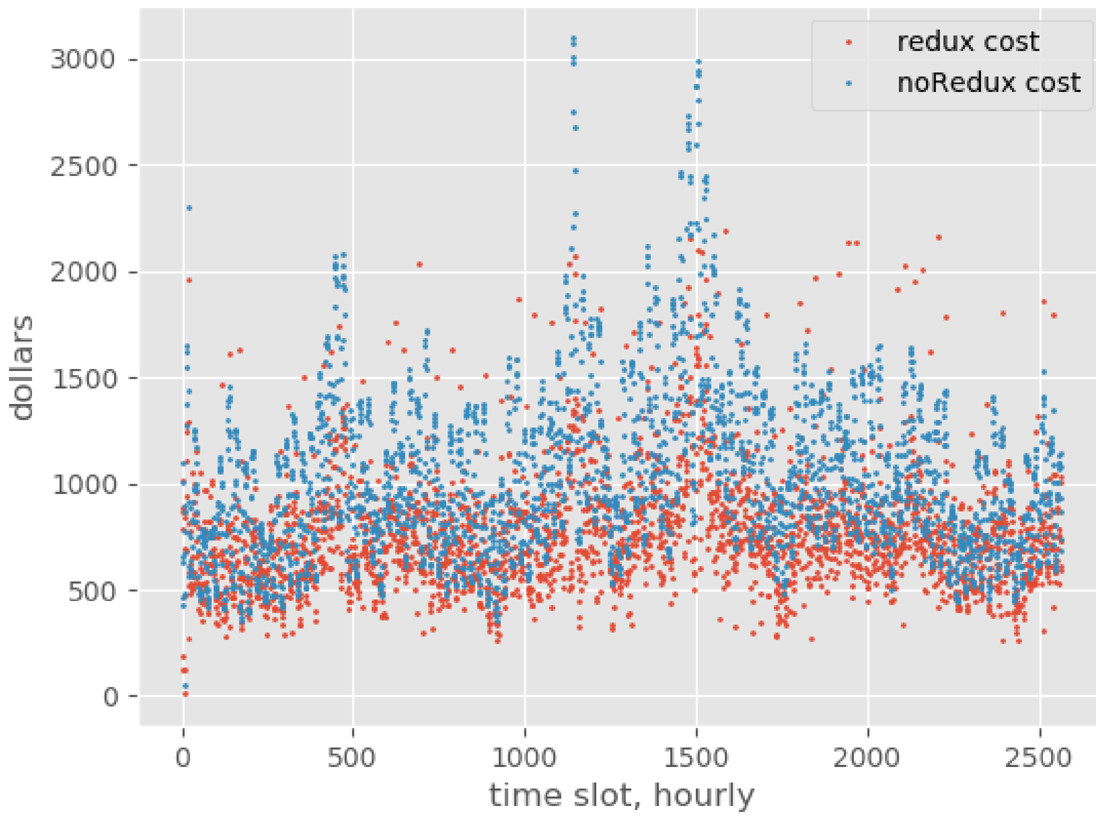

Figure 4.6: Total cost of a data center with REDUX during most time slot are lower than noREDUX

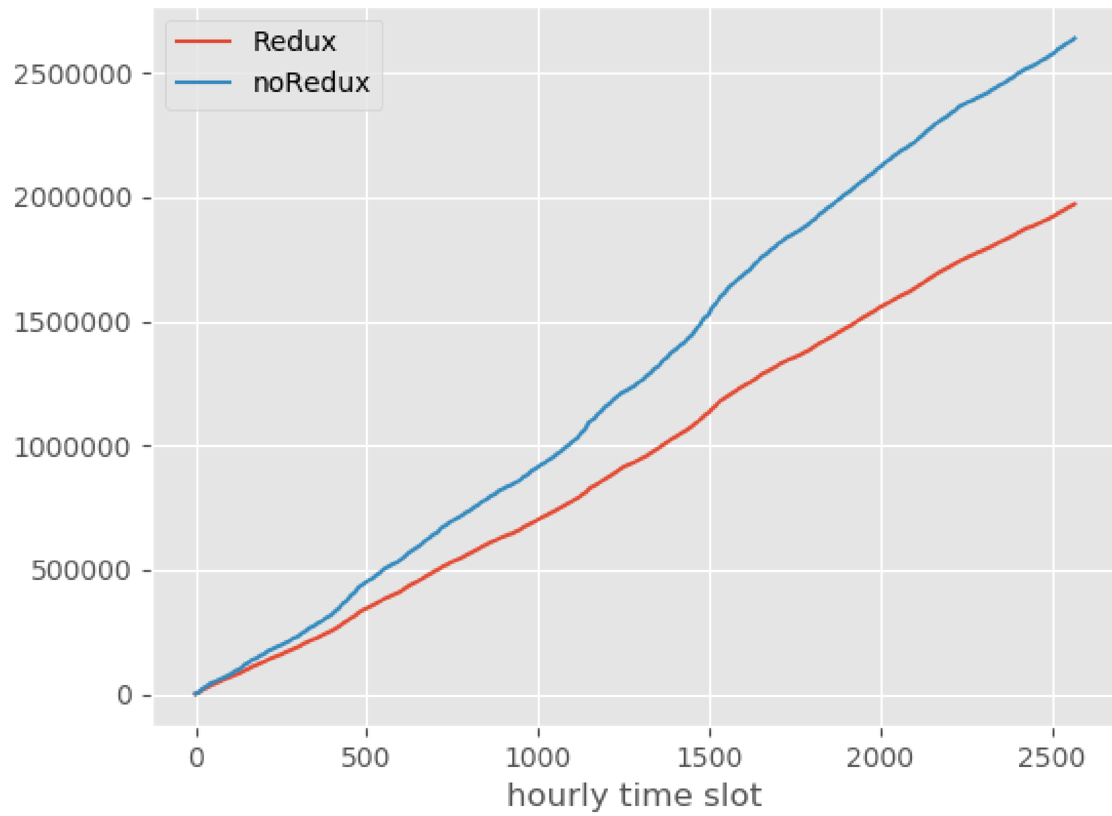

Figure 4.7: REDUX significantly reduces the energy cost of a data center managed by noREDUX - a traditional scheme.

judicious decisions to power the data center using either renewable energy or the distributed UPS system rather than the expensive power grid.

4.6.2 Impacts of UPS capacity

In the following experiments, we study the impacts of UPS capacity on energy efficiency of the REDUX system.

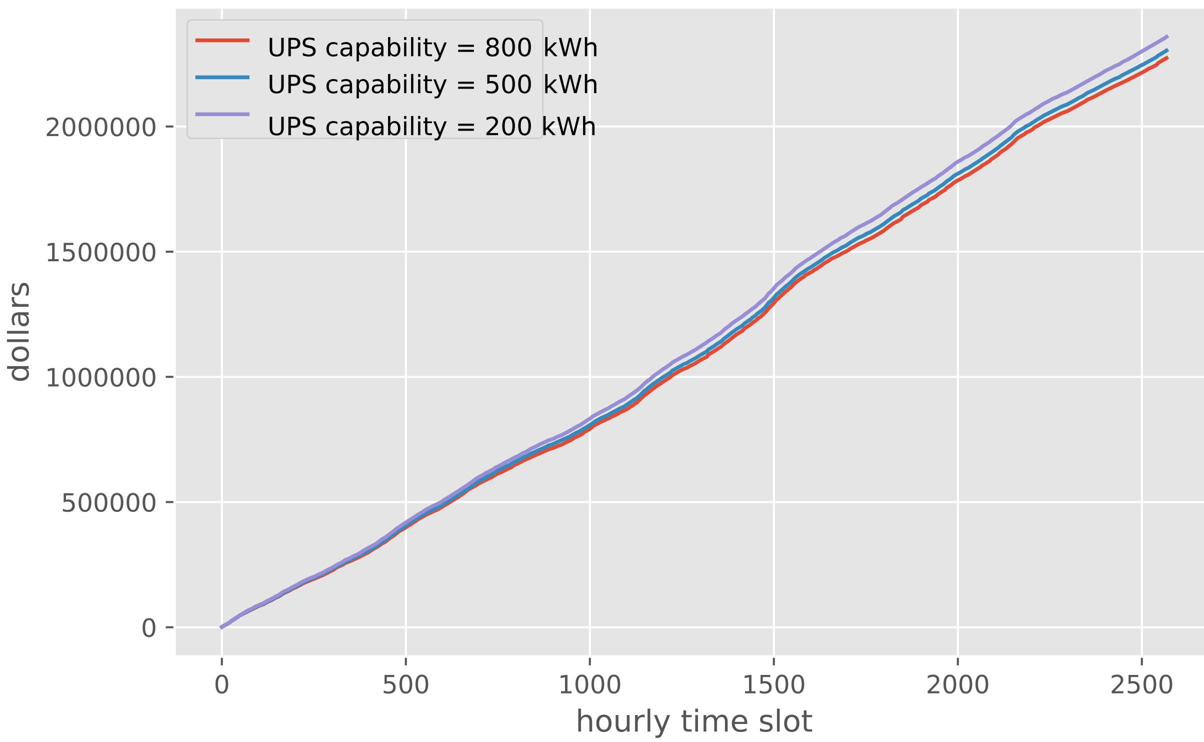

Figure 4.8: Total cost drops with an increased UPS capacity.

Fig. 4.8 unveils the impacts of UPS storage capacity on accumulated total cost. In this group of experiments, we test three UPS capacity levels while keeping the other parameters unchanged. Though the improvement made by large UPS capacity is marginal, the results still shows the advantage of large UPS devices on total cost savings in the long run. We conclude that given UPS devices with the high capacity, REDUX is offered ample opportunities to boost energy efficiency by storing low-cost energy for future usage.

4.6.3 Impacts of Workload Conditions

Now we investigate the impacts of workload conditions on energy efficiency of REDUX-enabled data centers. Fig. 4.9 illustrates the energy-supply distributions when the workload levels varies from 500 to 2500 with an increment of 500. As expected, when workload goes up, the total energy supply raises correspondingly. An intriguing observation is that the sensitivities of UPS, renewable, and grid power supplies on workload conditions are not on the same page. For example, the grid energy supply significantly surges when the workload increases from 500 to 2500. The renewable and UPS energy supply, on the other end of the spectrum, slightly go up and reach the peak when the workload is set to 1500. The UPS energy supply experiences the minimal level when the workload is as heavy as 2500.

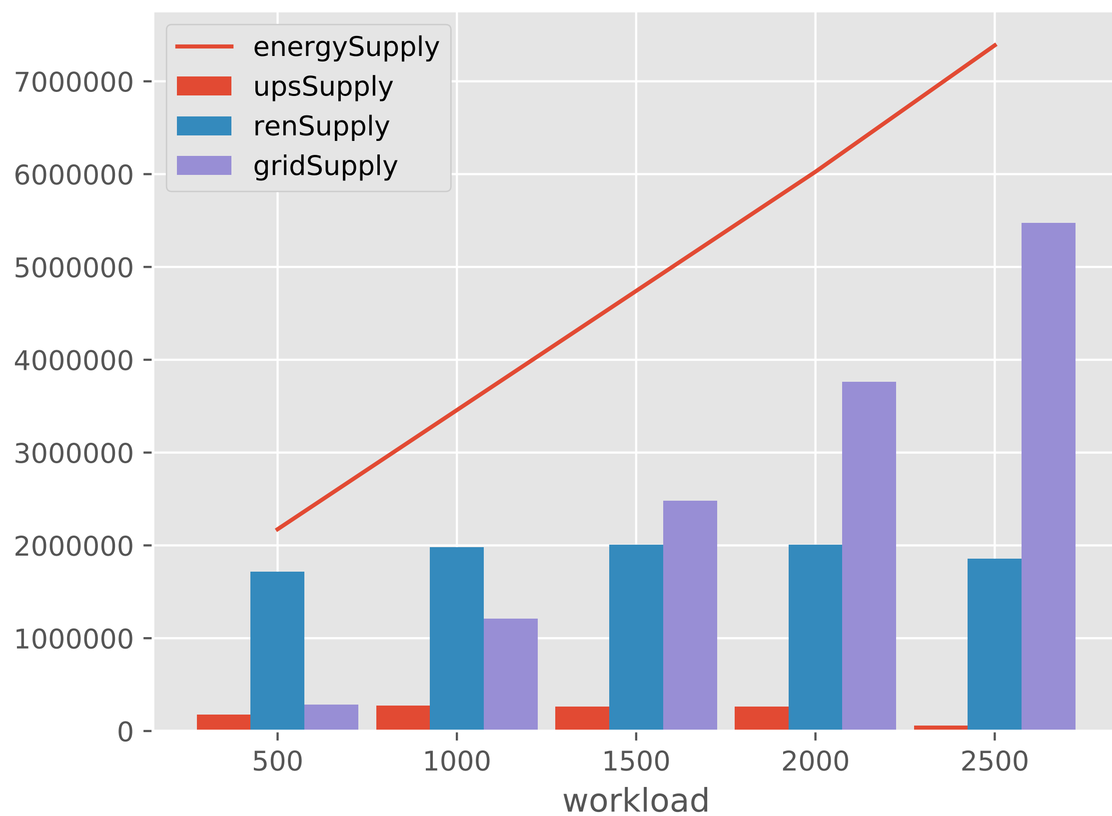

Figure 4.9: Energy supply comparation on different fixed workload condition

An important implication behind this group of experiments is that both the renewable energy and UPS supplies are exploited in the full potential to improve the energy efficiency of data centers when workload condition is maintained at the medium level.

4.7 Summary

In this chapter, we presented a resource manager called *REDUX2* to on-site renewable sources like solar coupled with a distributed UPS system to reduce energy cost of data centers. REDUX2 orchestrates UPS units to judiciously cope with the intermittency nature of renewable energy integrated with power grids. REDUX2 makes critical decisions on UPS charging and discharging with sufficient information on renewable-energy supply levels and time-dependent grid power prices. More specifically, REDUX2 makes prudent discharge of batteries to offer energy resources when renewable energy production is low or in the fluctuate condition. REDUX2 makes judicious decisions to charge batteries when renewable energy levels are high coupled with low loads.

Chapter 5

REDUX3: A Renewable-aware Scheduler for Energy Management in Data Centers

Leveraging on-site renewable sources like solar and wind provides ample opportunities on developing environmental friendly and energy-efficient data centers. Evidence demonstrate that renewable-aware job schedulers conserve energy by adjusting the arrangement of non-urgent workload according to renewable energy states. In our prior project [52], [53] and [51] articulated in Chapter 3 and 4, we have presented that the REDUX or REDUX2 system save electricity bills by judiciously managing inexhaustible on-site renewable energy, inexpensive stored energy from power grid or actively working energy storage system for a typical data center. The overarching goal of REDUX or REDUX2 is to curtail energy cost while maintaining high performance in data centers to address the following challenging issues: (1) the intermittent nature of renewable energy supply; (2) dynamically variate grid electricity price; and (3) efficient energy storage system for both smoothing energy supply and cheap energy cost.

In this chapter, we propose an unify designed energy management system with a renewable-aware scheduler called REDUX3. REDUX3 offers a smart way of both scheduling workload from data center users and managing the energy supply of data centers, which powered by the grid and renewable energy and equipped with distributed Uninterruptible Power Supply (UPS) system. As an integrated and smarter update from our previous work [52], [53] and [51], REDUX3 orchestrates renewable-aware workload scheduling to make back-fills or defer decisions for the non-urgent jobs, then properly allocate fluctuating renewable energy, UPS battery energy storage and grid power with dynamic price, aiming to minimize the long-term electricity bill for the data center. Compared with the existing strategies, REDUX3 demonstrates a prominent capacity of boosting renewable energy utilization.

The rest of the chapter is organized as follows. Section 5.1 describes the extended area from REDUX3 to REDUX and REDUX2, and present the brief discussions of related work listed in the comparison table 5.2 with our REDUX3 project. The framework and models of REDUX3 are detailed in Section 5.4. Section 5.5 articulates an array of heuristic algorithms at the heart of REDUX3. Section 5.7 describes the experimental settings and results. The detailed experimental results are analyzed and discussed in Section 5.8. Finally, Section 5.9 concludes this chapter.

5.1 Upgrading from REDUX2 to REDUX3

Energy efficiency is one of the key design principles of modern data centers. Inspired by the observations of renewable energy utilization and distinct pattern of job power profiles, we hypothesize that a renewable-aware scheduling mechanism running prior to a dynamic power manager has a potential to enhance the energy efficiency of data centers. In this chapter, we further develop the *REDUX3* system - an upgraded version of the REDUX2 energy resource manager (see Chapter 4) that enhance on renewable energy aware scheduling of energy demand side with workload from data center users.

We anchor this chapter in a renewable-energy-aware job scheduler to boost the energy-efficiency performance of REDUX. We refer to a system powered by the smart job scheduler as *REDUX3*, in which workloads are scheduled and dispatched to computing nodes in a data center. The rationale behind REDUX3 is sparked on a key observation of jobs from the perspective of users - each job has a distinct power consumption profile [68]. It is feasible to schedule jobs by considering expected power profiles accompanied by the status of renewable energy (available or outage) at the beginning of each time period. Our renewable-aware job scheduling mechanism enables REDUX3 to either (1) dispatch hefty load to the computing nodes with available cheap and green renewable energy or (2) defer non-urgent jobs when grid energy price is high amid insufficient energy stored in batteries. Compared with the legacy REDUX system, REDUX3 is adept at scheduling jobs in a holistic way to optimize energy efficiency of data centers fueled by renewable energy.

5.1.1 Contributions of REDUX3

In this chapter, we propose a workload-energy management system - *REDUX3* - to orchestrate renewable energy integrated with a power grid and a distributed UPS system. REDUX3 first examines jobs according to their power profiles with a help of renewable-aware scheduler to lay out a foundation for the subsequent executions in energy management; REDUX3 then adept at smoothing power supplies with the supply levels of renewable energy resources; By naturally scaling with data center size and eliminating a potential single point of failure, REDUX3 utilize per-server distributed batteries as an economical and secure battery backup; And finally, REDUX3 makes judicious decisions on UPS charging and discharging with vital information on renewable-energy supply levels and time-dependent grid power price.

REDUX3 makes three main contributions:

- *Unified Framework Design.* We design REDUX3 - a unified framework incorporating all the aspects of energy sources and workload. As a deep extension of our previous REDUX system [52], [53] and [51], REDUX3 takes full advantages of pre-execution renewable-aware workload scheduling, which governs job scheduling window and manages energy sources for computing servers in data centers.

- *Renewable-aware Job Scheduling* We devise the renewable-aware job scheduling with a dynamic scaling scheduling window as a vital part of the workload management. The scheduling policy leverages low-cost renewable energy in accordance with power profiles, priorities, and renewable energy states, and back-fill workload to idle nodes or defers the executions of non-urgent workload to further cut back the overall electricity cost.

- *Optimization.* We evaluate REDUX3 using the real-world workload traces and green energy data. We demonstrate that REDUX paves a way toward constructing modern data centers that are economically and environmentally friendly.

5.2 A Comparison Table

The topics related to energy efficiency of data centers is broadly and diversely discussed. Table 5.2 summarizes the major differences between our proposed REDUX3 and the existing power management techniques for data centers.

Table 5.1: Comparisons between REDUX3 and the existing energy management techniques

Energy Management	**Renewable-aware Workload Scheduling**	**Workload Scheduling**	**Renewable Energy**	**Grid Price Fluctuations**	**UPS Enabled**	**Unified Framework**
iSwitch	×	√	√	√	×	×
ReUPS	×	×	√	×	√	√
GreenSwitch	×	×	√	√	√	×
Geo-distributed	×	×	√	×	√	×
Two-stage	×	×	√	×	√	×
CQSim	×	√	×	×	×	×
GreenSlot	√	√	√	√	×	×
Green IT	√	√	×	√	×	×
REDUX (This Study)	√	√	√	√	√	√

5.3 A Comparison Table

The topics related to energy efficiency of data centers is broadly and diversely discussed. Table 5.2 summarizes the major differences between our proposed REDUX3 and the existing power management techniques for data centers.

In what follows, we briefly introduce the related systems and solutions listed in Table 5.2, and compare with our work in this chapter.

- iSwitch [45] did not considered any energy storage method like distributed UPS system, which will lead to waste of energy when renewable energy is sufficient or grid power price is cheap.

Table 5.2: Comparisons between REDUX3 and the existing energy management techniques

Energy Management	**Renewable-aware Workload Scheduling**	**Workload Scheduling**	**Renewable Energy**	**Grid Price Fluctuations**	**UPS Enabled**	**Unified Framework**
iSwitch [45]	×	√	√	√	×	×
ReUPS [48]	×	×	√	×	√	√
GreenSwitch [32]	×	×	√	√	√	×
Geo-distributed [71]	×	×	√	×	√	×
Two-stage [42]	×	×	√	×	√	×
CQSim [68] [64]	×	√	×	×	×	×
GreenSlot [31]	√	√	√	√	×	×
Green IT [35]	√	√	×	√	×	×
REDUX3 (This Study)	√	√	√	√	√	√

- ReUPS [48] Successfully integrated both renewable energy and a centralized UPS system, but ignores the fluctuation nature of grid power price, and lack of a hierarchical design of the framework.

- GreenSwitch [32] covered all intermittent renewable energy, fluctuate grid energy price and energy storage system, but lack both consider details on workload and a unified framework design.

- Geo-distributed [71] adopted the Lyapunov optimization technique to design a close-to-optimal online algorithm. However, the intermittent nature of renewable energy and possible dynamic changing grid energy price was ignored.

- Two-stage [42] developed a mathematical optimization model for energy-efficient while ensuring the desired level of renewable energy utilization and quality of service. However, it lacks consider of variable grid energy price and workload scheduling.

- CQSim [68] [64] intended to provide a generic job-power-aware scheduling mechanism to minimize electricity bills without impacting system utilization during both on-peak and off-peak pricing periods. The possibility of cheap and clean renewable energy were overlooked in the above studies.

- GreenSlot [31] proposed a scheduler for parallel batch jobs in a data center powered by a photovoltaic solar array. Grid power is only treated as a backup energy source, where off-peak cheap grid energy is never stored in any energy storage device.

- Green IT [35] implemented the "GreenVarPrice" version of the GreenSlot scheduler, which takes into account the variation of grid energy price. In these schedulers, there is the lack of a unified high-level framework design.

5.4 Framework Design

We start this section by depicting the advanced renewable-aware workload management system - REDUX3 at a high level view. Next, we shed some light on the design issues of REDUX3, which orchestrates two core mechanisms, namely (1) a smoothing mechanism (see Section 5.4.2) and (2) a renewable-aware workload scheduler (see Section 5.4.3). In Section 5.4.4, we elaborate the integration between the smoothing mechanism and the scheduler.

5.4.1 Overview

The *REDUX3* system is centered around two key components, namely, (1) renewable-aware workload scheduler and (2) REDUX workload-power manager [52], [53] and [51]. REDUX3 is a cross-layer power optimization system, which resides in both the front end as a workload scheduler, and the back end as a data-center energy resource manager. REDUX3 is adept at tuning power and managing workload in data centers. The scheduler embedded in REDUX3 is running in the batching mode, which is popular in the realm of big-data applications running in data centers.

Fig. 5.1 plots an overview of the REDUX3 framework design. The system inside the large box persistently communicates with the five external entities, namely, (1) renewable energy supplies, (2) power grid with diesel generator as a backup, (3) distributed UPS units, (4) server racks, and (5) workload submitted by users. Renewable energy (e.g., wind and solar), grid power, and diesel generators are seamlessly integrated into the system. On the

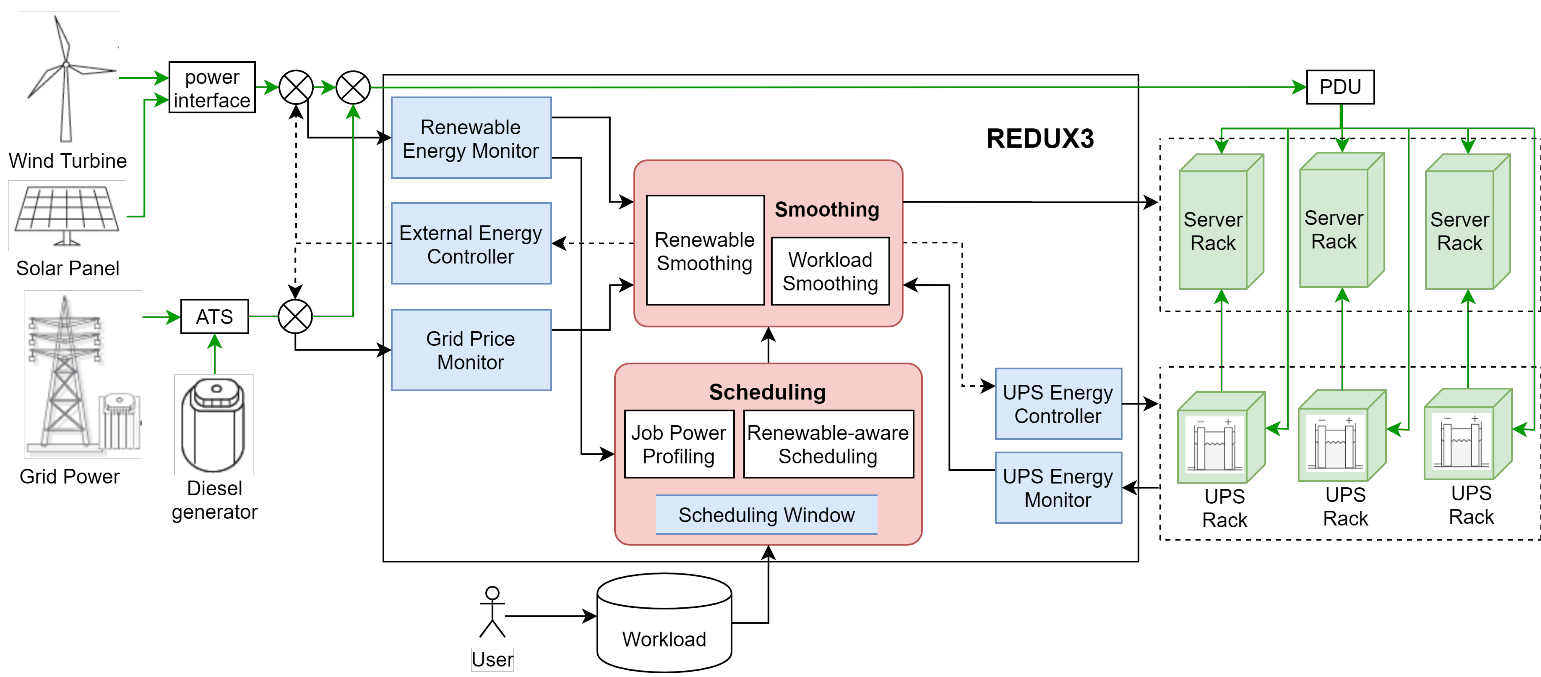

Figure 5.1: The framework of the *REDUX3* system.

left-hand side of Fig. 5.1, there are a green colored power circuit, a black colored monitoring data flows, and a dashed arrows of control signal flows. Through these flows, the monitoring data will feed both to the REDUX3 and the renewable aware workload scheduler, and the control signal flow will send to all external entities by the REDUX3 system except user workload. On the right-hand side of the framework, a distributed UPS system is attached to the server racks to improve efficiency, scalability, and reliability over a traditional centralized counterpart. The UPS system aims to buffer energy under light workload or power at a cheap price, and to consume stored energy in case of power spikes. Inspired by a prior work [40], we derive the required battery capacity based on the amount of power demand from smoothing processes and the corresponding energy stored in a battery for a given daily power profile. At the bottom of Fig. 5.1, jobs are submitted by users and queued in a waiting list before being fed into the renewable-aware scheduling system.

All the modules of the REDUX3 system falls into two categories. The first category includes control and monitor modules (see all light-blue colored modules in Fig. 5.1); the second category groups the modules forming the two core subsystems (see the modules in light-red colored in Fig. 5.1).

There are three monitoring modules and two controller modules for the external entities: (1) the renewable energy monitor, the grid price monitor and UPS monitor which collect trace data and update the status of all the external entities; and (2) the external energy controller and the UPS energy controller which manages the system according to received control signals from REDUX3. Seamlessly integrating the three monitoring modules, the system keeps track of energy supply trace data from on-site renewable devices or off-site grid networks, and energy storage amount in UPS racks. On the other hand, the monitoring modules facilitate the the two core subsystems with updated workload and power status. The controllers deliver control signals to the external entities, thereby tuning renewable and grid energy supply and determine recharging or discharging of UPS devices.

5.4.2 The Smoothing Mechanism

Following our previous work [52], [53] and [51], REDUX3 is a management system orchestrating dynamic grid energy price, UPS units and renewable energy for computing servers according to workload requirements in data centers.

As an upgrading from the previous chapters 3 and 4, REDUX3's smoothing mechanism embraces two core modules (1) a *renewable smoothing* module and (2) a *workload smoothing* module, communicate with the five external entities through the monitor and control modules (see Section 5.4.1). And the "UPS Supply Controller" from REDUX has been internalized as a passive controlling module in REDUX3 , as it only controls discharge/recharge decision and amount of the UPS devices. Through the assistance from the distributed UPS system, the renewable smoothing module is responsible for effectively dealing with cases where renewable energy fluctuates due to an environmental status. The workload smoothing module is focused on treating overload conditions, in which (1) workload exceeds the underlying data center's computing and energy capacity or (2) deferrals are a preferred choice when renewable energy is in outage state and grid price is relatively high.

5.4.3 The Renewable-aware Workload Scheduler

In the REDUX3 system, the renewable-aware workload scheduler keeps track of renewable energy status and workload power profile data in each time slot. The following three key components are seamlessly integrated in the scheduler.

The scheduling window module. This module is responsible for filling up a windowed list by picking jobs from the job waiting queue submitted during the time slot, thereby striving to maintain fairness among the jobs. In addition, the scheduling window module updates the states of the job list after the smoothing mechanism.

The renewable-aware scheduling module. The scheduling module first retrieves state of renewable energy from the monitor, and then judiciously makes final scheduling decides according to the renewable-aware scheduling policy. Basically, the scheduling module (1) back-fill workloads of next time slot from executing jobs if existed when renewable energy is abundant, or (2) defer non-urgent jobs to the next time slot at outage time slot of renewable energy.

Job power profiling module. The profiling module, managing power profile information, assigns a power profile to each job. After assigning profile information, this module records the total energy demand from the jobs residing in the scheduled window.

5.4.4 Put It All Together

In the REDUX3 management system, the renewable-aware scheduler first collect the state of renewable-energy from the renewable energy monitor module, then provides workload energy demands to the following energy management system. The smoothing mechanism then predicts energy supplies, followed by making a renewable-energy smoothing decision. Through these processes, the smoothing mechanism judiciously decides, in this time slot, the executable workload level serving as an energy-supply capacity for a data center. Before finishing each time slot, the smoothing mechanism and the workload scheduler perform independently and concurrently on the signal control and updating tasks.

- *Signal Control* The smoothing mechanism delivers control signals to all the external entities. Sample signals include the amount of energy needed to draw from gird network and renewable sources, recharging/discharging UPS batteries, and other information of job assignments to server racks.

- *Updating* After the renewable-aware workload scheduler updates job execution data in a scheduling window and removes these jobs from the list, the REDUX3 prepares remaining jobs to be executed in the next time slot.

5.5 Methodology

We advocate for a modularized design approach to constructing REDUX3, in which a high-level coordinating algorithm orchestrates the monitoring, scheduling and smoothing modules. We start this section by demonstrate principles on designing the workload measurement for the job scheduling in Section 5.5.1. We then articulating the high-level coordinating algorithm (see Section 5.5.2). Next, Section 5.6 sheds light on the design issues of the renewable-aware job scheduler.

5.5.1 Job Scheduling and Workload Measuring

After we built a solid foundation to measuring the total computational workload submitted by data center users within time slot t in Section 4.3.5, we now formulate principles on design workload measuring method with job scheduling ideas. Basically, workloads from data center users are defined by jobs coupled with arrival times, computing resource requests, and estimated running times, etc. Inspired by [68] and [64], we have these jobs submitted to the system through a newly designed renewable-aware job scheduler. Following a window-based optimization method in a priority wait queue, the scheduler assigns jobs to computing nodes governed by the following three design principles:

- The scheduling method should be aware of the availability of cheap energy sources such us sufficient renewable energy or low priced grid energy, thereby saving total energy cost at the first sight.

- Since jobs have versatile power consumption profiles, the estimated energy consumption of each job should be projected before the scheduling algorithm takes place.

- The impact on system utilization levels should be restricted to none or minor; job fairness and requests must be warranted. In case of any high priority job in the queue, idle computing resources should be minimized.

- The scheduler is configured to the batching mode, meaning that nodes assigned to a job become unavailable to other jobs until the current job is completed (i.e, no time-sharing).

Given the above principles, our job scheduling problem is stated as: How to schedule jobs in a windowed priority queue by maximizing the utility of available renewable energy to save total energy cost, while fulfilling job QoS requests and maintaining fairness? After solving this scheduling problem within a time slot t, the REDUX3 system will calculate each job's power demand during its execution and in terms of energy cost measured in $Watts$. The total power demand within time slot t (i.e., $W(t)$) is then measured in terms of kilo watts (kW) by the summation of these individual power demands. All these workloads vary in a wide range, taking values from a non-negative finite set with a maximum value denoted as W_{max}.

5.5.2 The Upgraded High-Level Control Algorithm

Algorithm 5 depicts the pseudo-code of the upgraded high-level algorithm in REDUX3. Similar with the previous chapters 3 4, the inputs of this algorithm include (1) energy supply values, namely, wind energy $E^{W}(t)$, solar energy $E^{S}(t)$, and energy storage $E^{UPS}(t)$ in UPS device, (2) the energy demand from workload $W(t)$, and (3) grid energy price $P^{G}(t)$ as variable, wind energy price P^{W}, solar energy price P^{S}, and UPS energy price P^{UPS} as constants. And we denote $E^{G}(t)$ as the energy draw from grid network after $E^{RE}(t)$ and $E^{UPS}(t)$ are decided during the time slot (see also Line 13 in Algorithm 5).

The high-level controller kicks off the power management task with determine the renewable energy state S_{RE} (see $S_{RE} = get\_renewable\_energy\_state()$; in line 3), which is categorized as, namely, *stable* (i.e., STA), *fluctuate* (i.e., FLU), and *outage* (i.e., OTA).

- *The Outage Case.* If the renewable supply is at the minimum level, the renewable state will be defined as OTA (see also the purple dots in Fig. 5.2).

Algorithm 5 The high-level control algorithm of REDUX3.

Require:

Renewable energy $E^W(t)$ and $E^S(t)$

UPS energy storage $E^{UPS}(t)$

Submitted workload $W(t)$

Price of all energy sources P^W, P^S, P^{UPS} and $P^G(t)$

Ensure:

Energy Cost $C(t)$ of time slot t

1: $t = 0$

2: **while** $t <= T$ **do**

3: $S_{RE} = get\_renewable\_energy\_state()$;

4: $renewable - aware\_job\_scheduling()$;

5: **if** $S_{RE} = FLU$ **then**

6: $ren\_supply\_smooth()$;

7: **end if**

8: $ups\_storage\_update()$;

9: $E^G(t) = W(t) \; - \; E^{RE}(t) \; - \; E^{UPS}(t)$;

10: $C(t) = P^G(t) \cdot E^G(t) + P^{ren}(t) \cdot E^{ren}(t) + P^{UPS} \cdot E^{UPS}(t)$;

11: $t + +$;

12: **end while**

13: **return** REDUX cost

- *The Stable Case.* If the renewable-energy supply exceeds the maximum level, the renewable state will be referred to as STA (see also the gray dots in Fig. 5.2). For the other cases, we compare the the renewable-energy supply with its previous exponential windowed average.

- *The Fluctuate Case.* If the change in the renewable supply from the previous time slots is in a *stable* interval, we envision this current renewable supply as STA; otherwise, the state should be categorized as FLU (see also values un-annotated in Fig. 5.2).

Taking both when $S_{RE} = STA$ and $S_{RE} = FLU$ as available states of renewable energy, the renewable-aware job scheduling procedure (see $renewable - aware\_job\_scheduling()$; in Line 4) then take place to apply the renewable-aware algorithm on a windowed priority queue, followed by transforming the job workload into energy demands derived from job power profiles. The details of this procedure is discussed in the following Section 5.6.

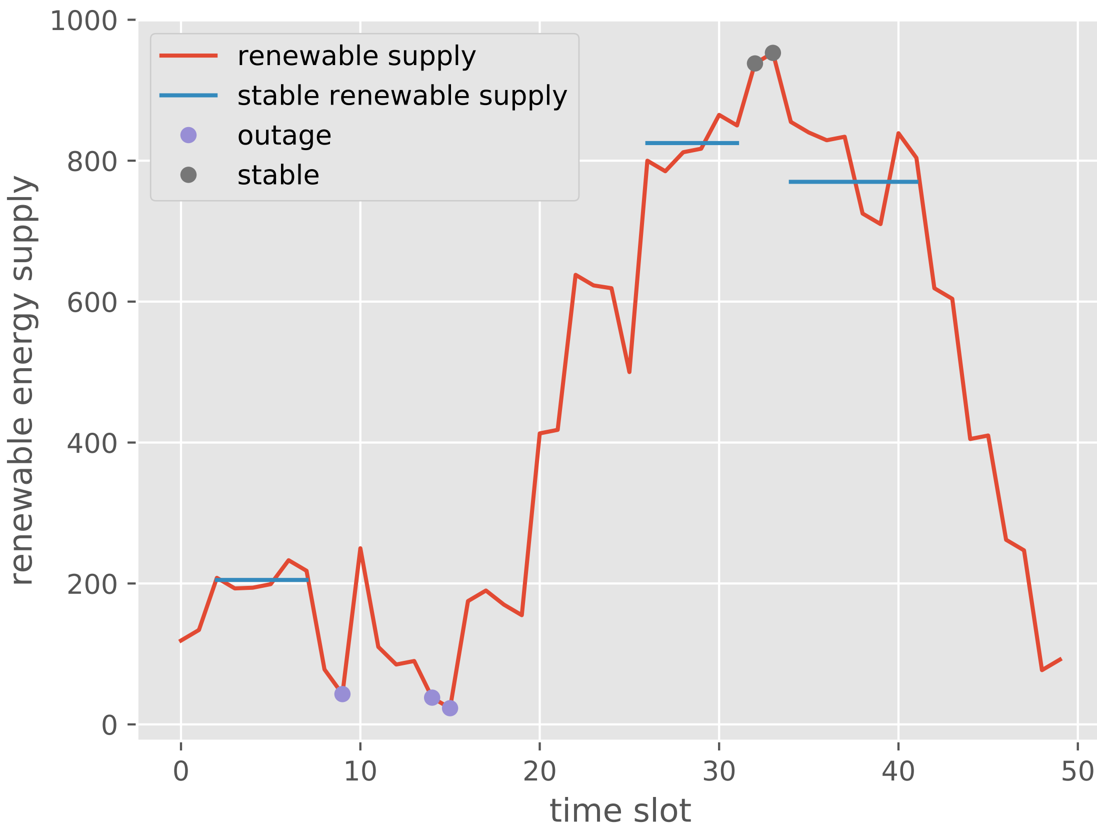

Figure 5.2: An example of the three state cases (i.e., OTA, STA, and FLU) of renewable energy supplies. The unannotated values are defined as fluctuate states or FLU.

After determining the state of the current renewable supply, REDUX3 further manages energy supply in accordance to the renewable-energy state. More specifically, if the renewable energy is in the fluctuation state, the renewable-energy-supply smoothing policy is invoked in Line 6. The smoothing procedure is governed by the current renewable supply data, a stable supply level that is dynamically updated, and a UPS energy storage condition.

The functionality of $ups\_storage\_control()$ procedure (see Line 8) is two-fold, namely, (1) making decisions on UPS discharging and charging and (2) updating the UPS status information. In responsible for protecting UPS units against frequent charging and discharging to maintain reliability of the distributed UPS system, this procedure stipulates a minimum UPS energy level, under which the UPS units should be protected.

Recall that the vital factors prescribed in our problem statement include workload demand W_t, renewable energy supply $E^{RE}(t)$ and net energy supply from UPS devices $E^{UPS}(t)$. After clearing out all these factors, REDUX3 computes the energy supply obtained from the grid network (see Line 9). The function in Line 10 calculates the total energy cost occurred in time slot t.

5.6 A Renewable-aware Job Scheduler

In this section we brush up on the description of renewable-aware job scheduling. Inspired by [68] and [64], our design embraces three key mechanisms, namely, (1) a scheduling window (see Section 5.6.1), (2) a scheduling algorithm (see Section 5.6.2), and (3) a job power profiling module (see Section 5.6.3).

In what follows, we detail the design and implementation issues of the scheduling window, the scheduling algorithm, and the job power profiling module in Sections 5.6.1, 5.6.2, and 5.6.3, respectively.

5.6.1 Scheduling Window

Maintaining fairness among jobs with various priorities and computing recourse requirements is a well-received concern amid the development of job schedulers. Demonstrated as a gray box in Fig. 5.3, we advocate for a window-based scheduling mechanism to facilitate the renewable-energy-aware scheduling algorithm without breaking fairness. In other words, a job submitted during time slot t first enters a general job waiting queue. Subsequently, the job is fed into the scheduling window if resources are permitted; any job that has already entered the scheduling window will never be kicked back to the waiting queue until being executed and marked as "finished". Although a scheduling policy may not allocate jobs in an FCFS order, this mechanism still manages to maintain relatively high fairness.

Another key design of the scheduling window lies in the renewable-aware dynamic scaling mechanism. Since any submitted job requests a number of processors during its execution, we derive and measure the size of the scheduling window as the total number of processors to be allocated by the data center. Selecting jobs from the window to be running on computing nodes should be governed by the metric of the renewable energy state. More specifically, renewable-aware and dynamic scaling is performed in the following fashion: (1) enlarging the scheduling window if renewable energy is sufficient or (2) shrinking the scheduling window when renewable energy is in the outage state. Guided by our empirical studies, we set this dynamic scaling ratio to 0.2 for the enlarging and shrinking process. The deriving of the most preferable level of this ratio depends on the aggressiveness of the energy-saving policy. The upper part of Fig. 5.3 demonstrates that the renewable energy state from renewable energy monitor dynamically scales the scheduling-window size.

5.6.2 Scheduling Algorithm

Our renewable-aware scheduling algorithm elects jobs from the scheduling window to optimize utilization of renewable energy. At the high level, the renewable-aware scheduling algorithm seeks to back-fill existing workloads within an enlarged scheduling window; the

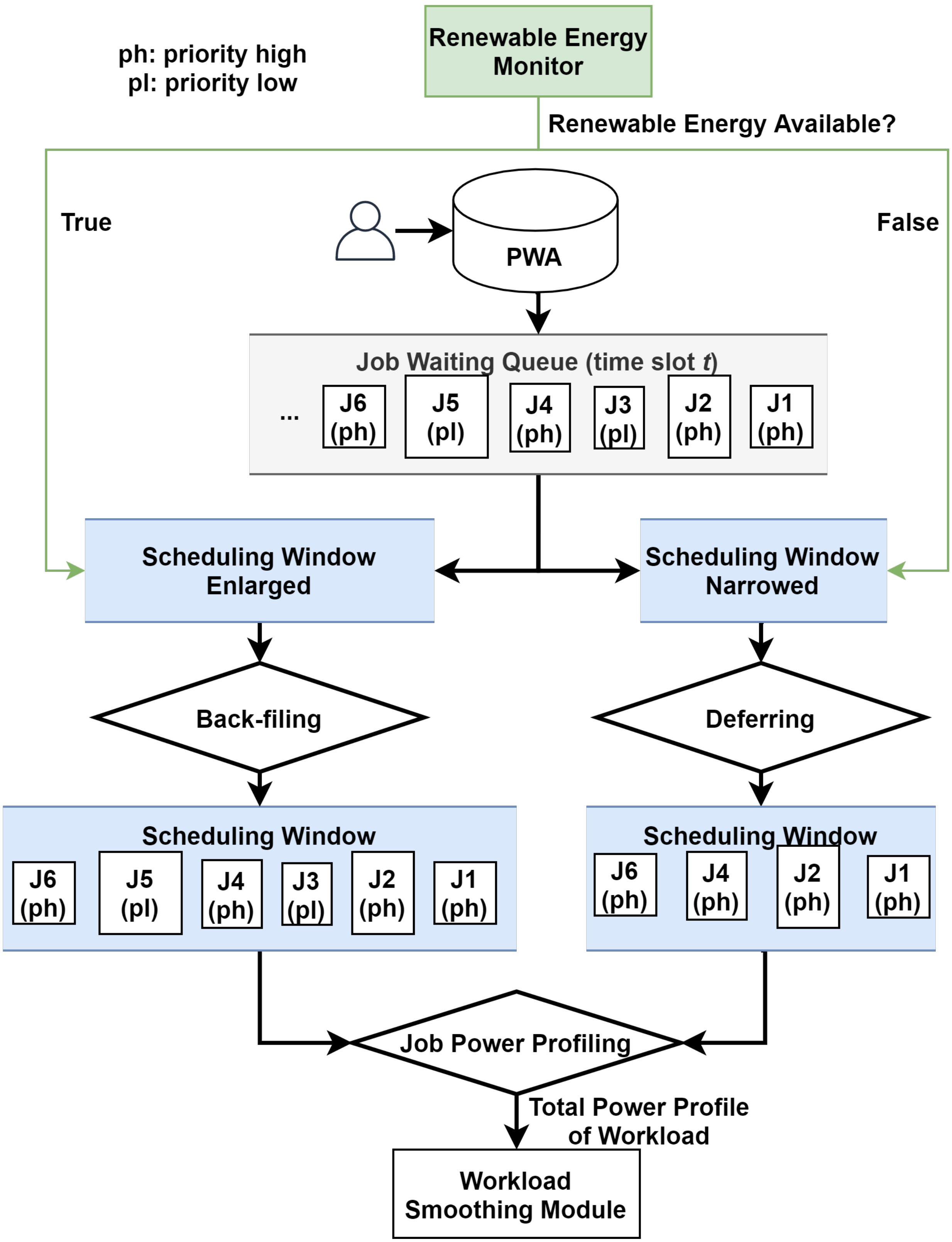

Figure 5.3: Overview of Renewable-aware Job Scheduling

scheduler defers non-urgent workloads from the narrowed scheduling window (see the lower part of Fig. 5.3).

At the beginning of each time slot, the algorithm fills up the dynamically scaled scheduling window (either enlarged or reduced) with jobs from the job waiting queue in an FCFS order. Next, either back-filling or deferring of workload is determined according to the state of renewable energy. Now we consider the following two cases.

- **Case 1.** If the renewable energy is abundant accompanied by an enlarged scheduling window, extra workload form existing jobs in the next time slot are back-filled into the scheduling window and dispatched to idle processors (see the LHS of Fig. 5.3). We avoid back-filling any new job from the job waiting queue because it is imperative to maintain an on-line processing status for the scheduling step, which is subject to a future study. Although archived in the PWA database, a new job from the next time slot has not yet been submitted for an online processing system.

- **Case 2.** With a narrow-scaled scheduling window coupled with the outage state of renewable energy, the scheduling algorithm switches to deferring low priority jobs to the next time slot (see the RHS of Fig. 5.3). Depending on the aggressiveness of the energy-saving policy, the REDUX3 system may either prevent all low priority jobs from entering the scheduling window in the first step or attempt to fill up the scheduling window with all high priority jobs first followed by low-priority ones.

Last but not least, as depicted in Fig. 5.3, each box with job number inside the job waiting queue represents one hour of workload of that job. This key design of the scheduling algorithm is driven by three reasons. (1) Any job's execution time spans over one time slot can be treated as multiple one-time-slot workloads to be fed into the job waiting queue at the beginning of each time slot, as long as the job keeps running on the same assigned nodes without migration. (2) Parallel computing among multiple nodes provides possibility of allocating a job's workload that belongs to the next time slot to the current time slot. (3)

This design can eschew overflow of any unfinished job when an enlarged scheduling window of the current time slot is scaled to a narrowed scheduling window in the next slot.

5.6.3 Job Power Profiling

To make an intelligent job scheduling decision, we ought to profile and model power consumption requirements of submitted jobs. Estimating power consumption of jobs running on clusters in a data center is nontrivial in that the running time and energy usage of a job are reliant on the allocated resources (e.g., the number of computing nodes and main memory capacity). For each job to be submitted to a data center, we profile the energy efficiency and performance of the job on a cluster. Such profiling data allow us to build a model to project the execution time and power consumption of the job given any assigned resources. In other words, the job's running time and energy usage are represented as the functions of the required number of nodes. Upon a job submission, such profiling and modeling data should be submitted along with jobs. In doing so, REDUX3 is capable of speculating each job's execution time and energy demands to make wise scheduling decisions. In our experiments, the profiling data are furnished in job traces containing job requirements (e.g., CPU time and energy demand).

Motivated by the findings in [68] and [64], we implement a job power-profile function leveraging a categorized normal distribution to estimate power consumption of a job from its power profile. More specifically, the power demand of the job is initially derived by the job's requested computing resources, followed by the categorized normal distribution method to finalized the job's power requirement value. The output of the power-profile function is then fed into the workload smoothing module of REDUX3 system as jobs' power demands during the current time slot. Power demands are gauged in terms of kWh. The detail settings of the job power-profile function can be found in Section 5.7.3.

5.7 Experiment Configurations

We systematically evaluate the design of REDUX3 by developing a simulated data center processing real-world workload. To mimic the real scenarios, we incorporate dynamic grid price as well as renewable energy supply traces into the simulator. Section 5.7.1 will introduce the renewable-aware scheduling simulator we developed. Next, we elaborate on the experimental settings of this project from the prospective of (1) configuration of job traces 5.7.2; Job Power Profile 5.6.3 and supply of energy sources 5.7.4.

5.7.1 ReCQSim: Renewable-aware Scheduling Simulation

We implement our renewable-aware scheduling method in ReCQSim, which follows the scheduling framework of the existing trace-based event driven scheduling simulator named *CQSim* [68] [24]. Tailored for the investigation of renewable-aware workload scheduling algorithms and takes renewable energy supply states into consideration during a scheduling process, ReCQSim is an integration of the job scheduling modules implemented in the CQSim simulator and renewable-aware modules from the REDUX3 system.

ReCQSim detects the state of renewable energy with help of the modules from REDUX3 at the beginning of each time slot. With the mechanism elaborated in Section 5.6, the renewable-aware scheduler makes a decision on back-filling when renewable energy is available, and priority defer when unavailable in the scheduling window. Next, the job power profile module computes workload of all the jobs in the scheduling window; Finally, the workloads are measured in the unit of kWh. The workload data coupled with all the job information are fed into REDUX3. More specifically, REDUX3 allocates the sorted jobs in the scheduling window into the available computing nodes of the data center. Then, REDUX3 acts as a management system orchestrating grid power, UPS units and renewable energy for computing servers according to workload requirements in data centers. It is worth mentioning that REDUX3 strives to maintain a desirable balance between energy cost and system performance. After each simulation round is accomplished, the scheduling window is updated with the current status of each job. The above simulation round is repeatedly executed in ReCQSim until all the trace data are played.

5.7.2 Job Trace and Size of Data Center

We replay real-world workload traces archived in a well established online repository [25]. All these workload logs - collected from the large-scale parallel systems in production - were scrubbed by the administrators to remove anomalous data that could skew the performance evaluation on different scheduling schemes. To gauge the performance of the renewable-aware scheduling algorithms, we use five key parameters in each trace file, namely, job ID, job submit time, job requested time, job requested number of processors, and the priority queue number to represent the priority level of the job. In order to demonstrate the effectiveness of our design on a broad range of workload settings and traces, we exterminate the following log files from *PWA* [25]: *UniLu Gaia*, *MetaCentrum2* [39], *CIEMAT Euler* [57], and *KIT FH2*. We choose to replay these traces, because (1) the traces encompass at least three-month workload data, which are aligned with the duration of the power supply dataset and (2) the traces were acquired very recently, implying that these large-scale systems are equipped with cutting-edge technologies.

Table 5.3: Comparisons between REDUX and the existing power management techniques for data centers.

Log Name	**From**	**To**	**Mo**	**CPUs**	**Jobs**
UniLu Gaia	May14	Aug14	3	2,004	51,897
MetaCentrum2 [39]	Jan13	Apr15	28	8,412	5,731,100
CIEMAT Euler [57]	Nov08	Dec17	110	1920	9263012
KIT FH2	Jun16	Jan18	19	24048	114,355

The aforementioned job trace data represent the behavior of a single parallel system, which is relatively small compared with computing platforms housed in data centers. To address this concern, we expand the trace data to resemble the scale of modern data centers simulated in REDUX3. More specifically, we scale up the job power consumption to match the size of a data center in the simulation project.

5.7.3 Job Power Profile

It's nontrivial to assess the system power profiles of job traces for parallel systems. While workload traces of parallel systems are widely available on the open repositories, there is the lack of power consumption data embedded in the traces. When workload traces consist of power data in some cases, the linkages between computing load and power consumption are obscure. Because the PWA trace excludes the power information connected to the tested workloads, we implement a job power-profiling module (see also Section 5.6.3) to render power profiles for jobs within time slot t.

The input of the power-profiling module includes the number of requested processors as well as expected running time during slot t of each job. Following guidelines documented in [68] and [64], we set the power demand of a computing node to a value anywhere between 20 to 60 Watts using a normal distribution. There are four processors in each node and; therefore, the profiling module randomly assigns the power demand of each processor in a range between 5 and 15 W following a normal distribution. Next, the power profile of each job running in slot t is multiplied by the power dissipation of all the allocated processors and the job running time. Finally, the profiling module accumulates the power demands of all the jobs running in slot t, thereby deriving power demand measured in kWh from the job trace data in slot t.

5.7.4 Energy Supply

Same as the REDUX2 project, we test the renewable-aware scheduler and the REDUX3 system using the electricity price of a real-world power grid in the New York state during a 3-month period from June 1 to August 31, 2016. We select this time period, because the solar and wind data were also collected and available for our simulation project. Fig. 5.4 reveals that the power grid price dynamically changes during the three-month time interval, in which the price noticeably spikes three times at around 480, 1200, and 1500 hours. The maximum and minimum prices in the trace are \$1.82 and \$0.26 $/Kwh$, respectively. According to the pricing policy of a hypothetical power company from which data centers can benefit from an off-peak price discount during midnight of a day. We use 2/3 of the previous recorded arithmetic average price as a threshold to determine the state of high or low grid price.

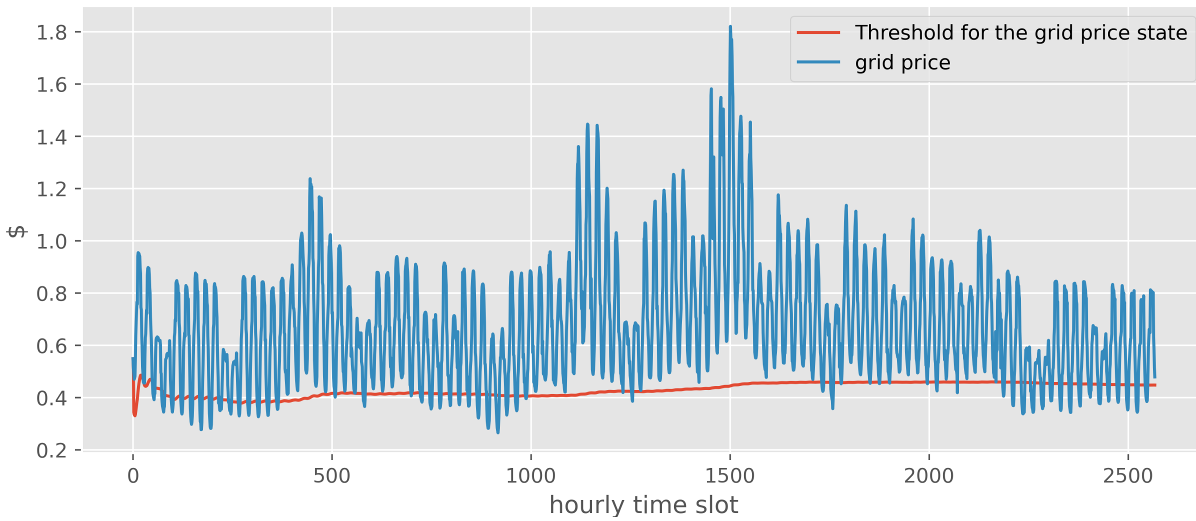

Figure 5.4: The electricity price of a power grid in the New York state during the period of three month ranging from June 1 to August 31, 2008. Unit: $\times 100\$/\text{MWh}$

We again retrieve the solar supply of the New York state the National Renewable Energy Laboratory's (NREL) database. To align the time window with the that of the above electricity price, we focus on the solar trend from June 1 to August 31, 2008. Fig. 5.5 plots the solar supply data measured in KWh. The solar and wind power price levels are fixed at \$0.09/Kwh and \$0.07/Kwh, respectively.

Inspired by a prior study [40], we assert that an array of distributed UPS devices with 12V lithium iron phosphate (LFP) batteries are installed. Attaching to each server provisioned at 60Ah, the batteries per rack have a maximum capacity fitting within server-size constraints. Because a distributed UPS system has the distinctive advantage of scalability, this setting can be configured according to servers' maximum energy demands. The impact on changing the relative scale of attached batteries to each server can be found in our early dissertation study on REDUX [52] and [53]. The energy cost paid to recharge/discharge batteries is calculated as \$0.07/kWh.

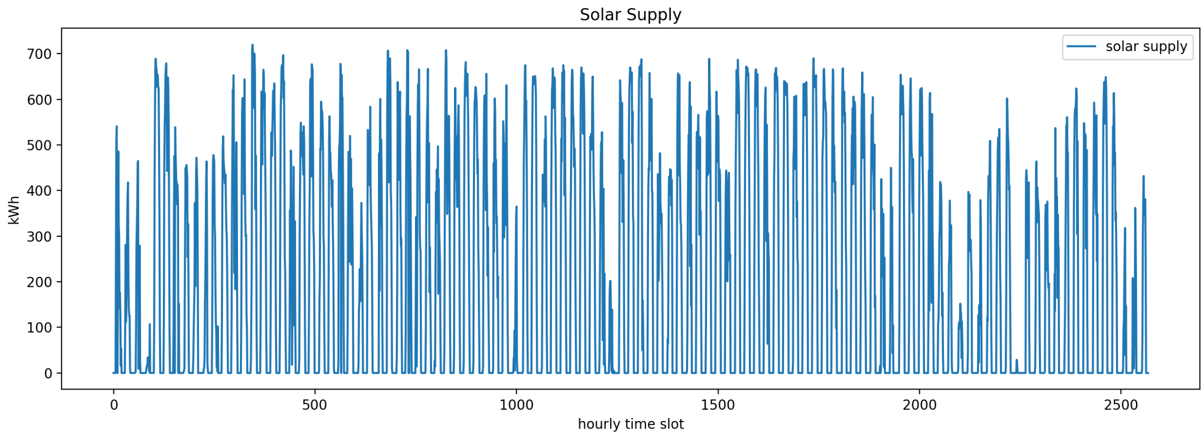

Figure 5.5: The solar supply of New York state during the three-months period ranging from June 1 to August 31, 2006.

5.8 Results and Analysis

This section is dedicated to the experimental results of the proposed renewable-ware scheduling strategy on energy saving and related analysis. In our empirical study, we first, in Section 5.8.1, present experimental results to demonstrate the benefits of applying the renewable-aware scheduler to various energy-efficiency management systems including the our baseline REDUX design. Next, Section 5.8.2 compares the renewable-aware scheduled REDUX3 system with our previous design of REDUX and REDUX2. In the following three sections, Section 5.8.3 make changes on the scale of data center to see if there is any impact exists. And finally, Section 5.8.4 enlarges the length of timeline and compare the energy saving result with 3, 6 and 9 months.

5.8.1 Energy Cost Saving

We first conduct the energy-cost-efficiency experiments driven by the representative workload trace $SDSC-SP2$. We compare the discrepancy of workload and accumulated energy cost between the renewable-aware REDUX3 and the baseline REDUX system. Fig. 5.6 depicts the workload difference between REDUX3 and REDUX systems during each time slot. A clear observation is that the kWh value of workloads scheduled by the renewable-aware design are more fluctuate than the workloads from the baseline REDUX system. This pattern is expected because parallel workloads can be either back-filled into a scheduling window amid sufficient renewable energy or deferred to the next time slot. The decision of deferring jobs is based on the job priorities when there is an outage in the renewable energy. Generally speaking, we expect a similar swing pattern between the renewable energy and the back-fill or defer operations from the renewable-aware scheduler.

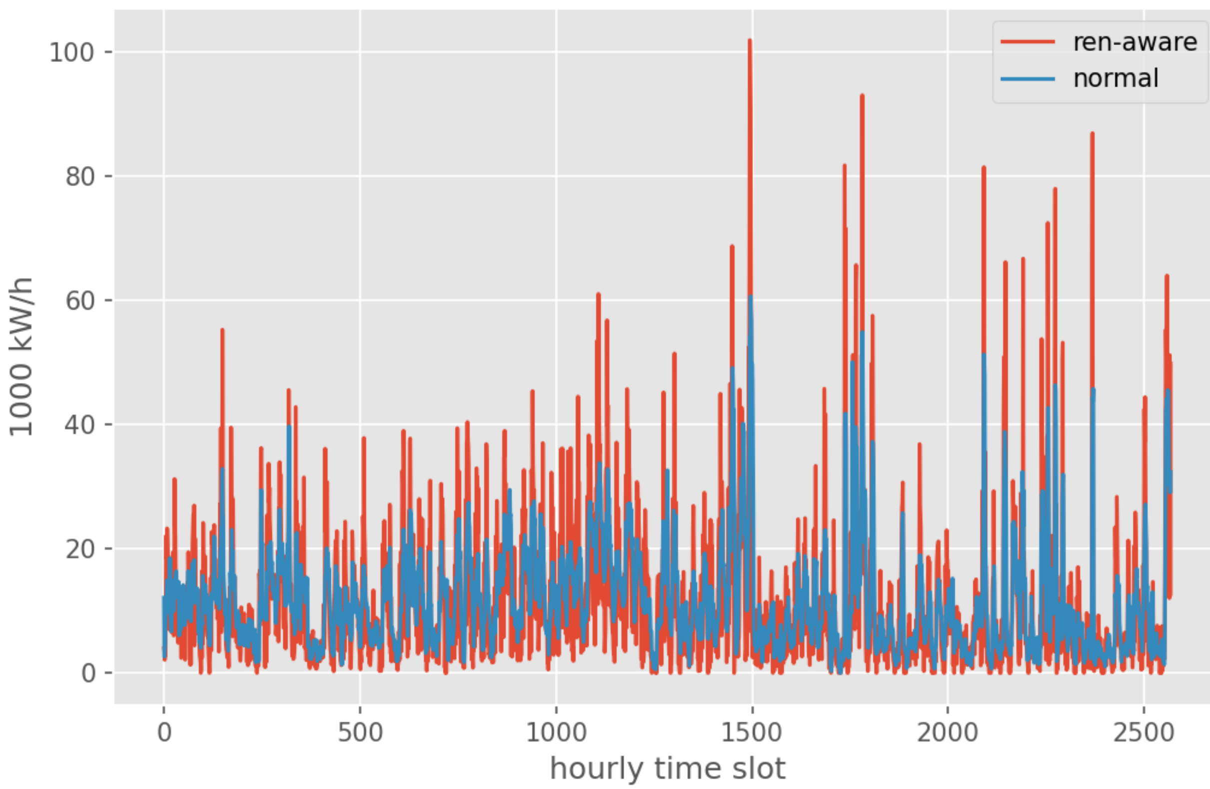

Figure 5.6: The difference of workload in *1000 kWh* between renewable-aware and normal workload scheduler across time slots

Table 5.4 summarizes three sample differences when renewable energy is available and experiencing an outage. It is obvious that the values (in $1000kWh$) of workload from renewable-aware scheduling is substantially higher when renewable energy is available, and greatly lower when renewable energy is at the outage state.

Table 5.4: Workload (in $1000kWh$) comparisons of renewable-aware REDUX3 and original REDUX system in representative time slots

Time Slot	**2370**	**1501**	**2092**	**1736**	**1495**	**2368**
Renewable Energy Available?	Yes	Yes	Yes	No	No	No
ren-aware	5.51	2.51	9.15	81.69	96.09	86.87
normal	45.73	46.92	50.45	39.90	53.88	44.40
difference	44.55	44.42	41.3	-41.79	-42.21	-42.48

Fig. 5.7 demonstrates the accumulated total energy cost (in million \$) of the REDUX3 and REDUX systems. We observe that the energy-cost gap between the two systems is widening along with the timeline. Intuitively, the renewable-aware REDUX3 has an edge over REDUX in terms of energy efficiency, offering cost savings at the rate of almost 8% at the end of the timeline. It is evident that the renewable-aware scheduler optimize the energy efficiency of REDUX3 during each time slot by (1) providing higher workload when inexpensive renewable energy is available and (2) lowering workload when expensive grid or UPS energy is the only choice. The detailed value of accumulated energy cost and relative saving rate against the REDUX system at the end of each month can be found in Table 5.5.

Time Slot	**1st**	**Cost Saving**	**2nd**	**Cost Saving**	**3rd**	**Cost Saving**
Ren-aware	4.82	0.36	12.57	1.33	16.23	1.74
Original	5.18	-	13.9	-	17.97	-

Table 5.5: Cumulative energy-cost (in million dollars) comparisons and savings between renewable-aware REDUX3 and original REDUX system at the end of each month

In the following subsections we explore each of the impacts of the renewable-aware REDUX3 design on energy cost.

5.8.2 Impacts of Workload Conditions

To assess the effect of workload traces on energy efficiency, we conduct the same energy-cost experiments driven by the three other traces extracted from the Parallel Workload Archive, namely $SDSCBLUE$, $SandiaRoss$, $UniLuGaia$. We pick these traces, because the traces are comprised of user run-time estimates data - an input of the power-profile module.

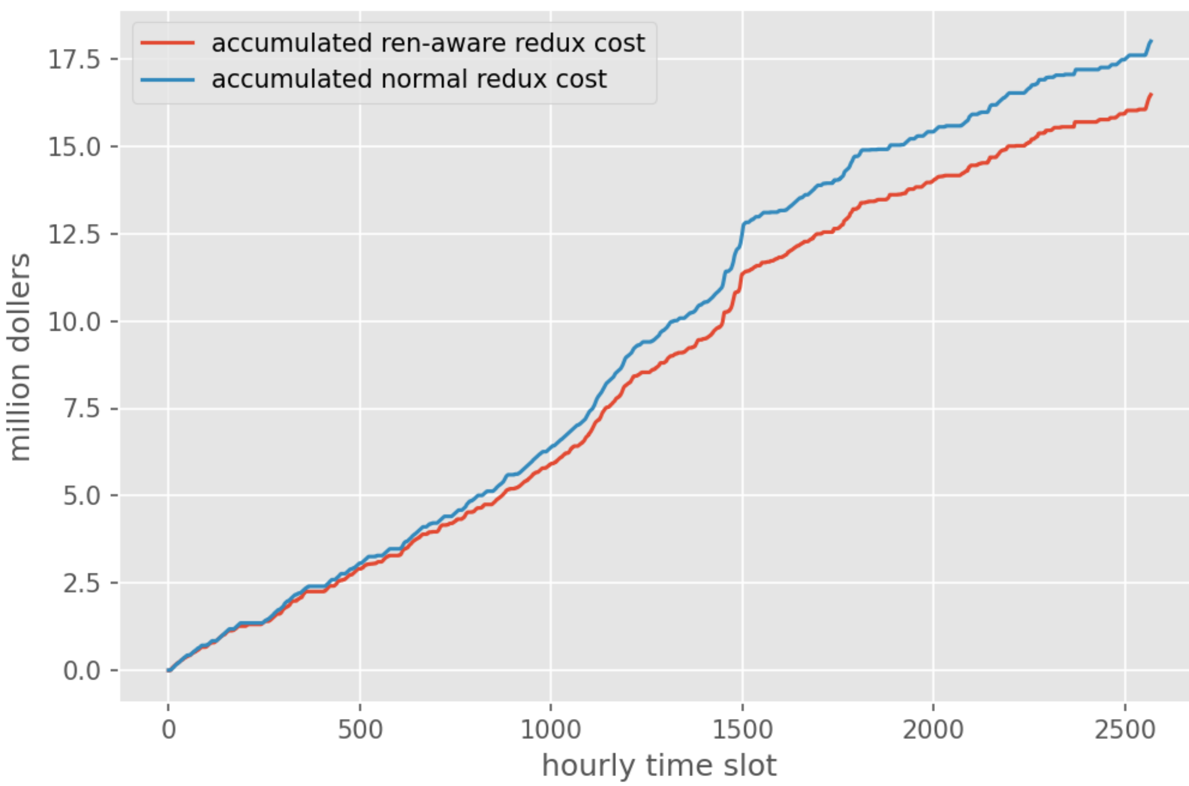

Figure 5.7: The accumulated energy cost of renewable-aware REDUX3 and original REDUX system

Table 5.6 lists key features of these workload data. Fig. 5.8 demonstrates that the renewable-aware REDUX3 system always prevails the original REDUX system on accumulated energy cost at time slots 1000, 2000, and 2600. The results indicate that REDUX3 boosts the energy efficiency across the tested workload traces. In particular, REDUX3 cuts back the energy cost by an average around 25% under the *UniLuGaia* workload.

Data Trace	Months	CPU	Job executed	Utilization Rate %
SDSC SP2	24	128	59,715	83.4
SDSC Blue	32	1152	243,306	76.7
Sandia Ross	37	1524	57,882	49.9
UniLu Gaia	3	2004	51,987	84

Table 5.6: General features of workload traces

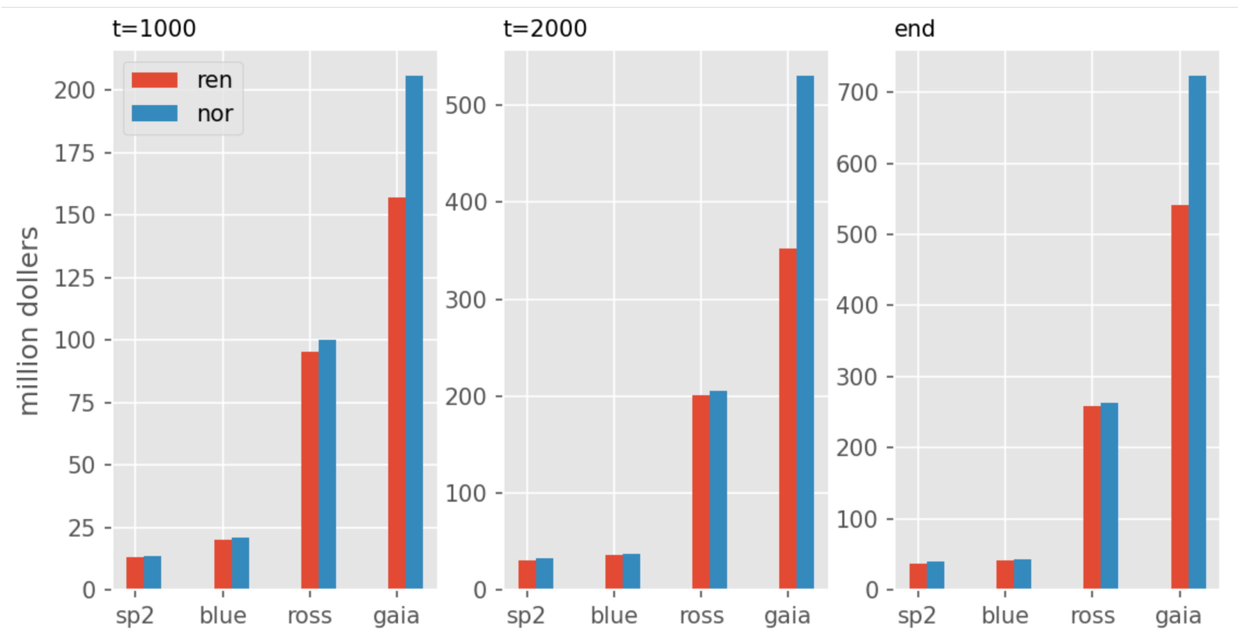

Figure 5.8: Comparison of Different Workload Traces

5.8.3 Impacts of Data Center Scale

A natural concern on the previous result is how various data center scales can affect energy efficiency. We here examine this possible impact by scaling up the total number of processors of the data center tested in our aforementioned experiments. We apply four opposite multipliers when calculating the workload of each time slot from trace data $SDSCSP2$.

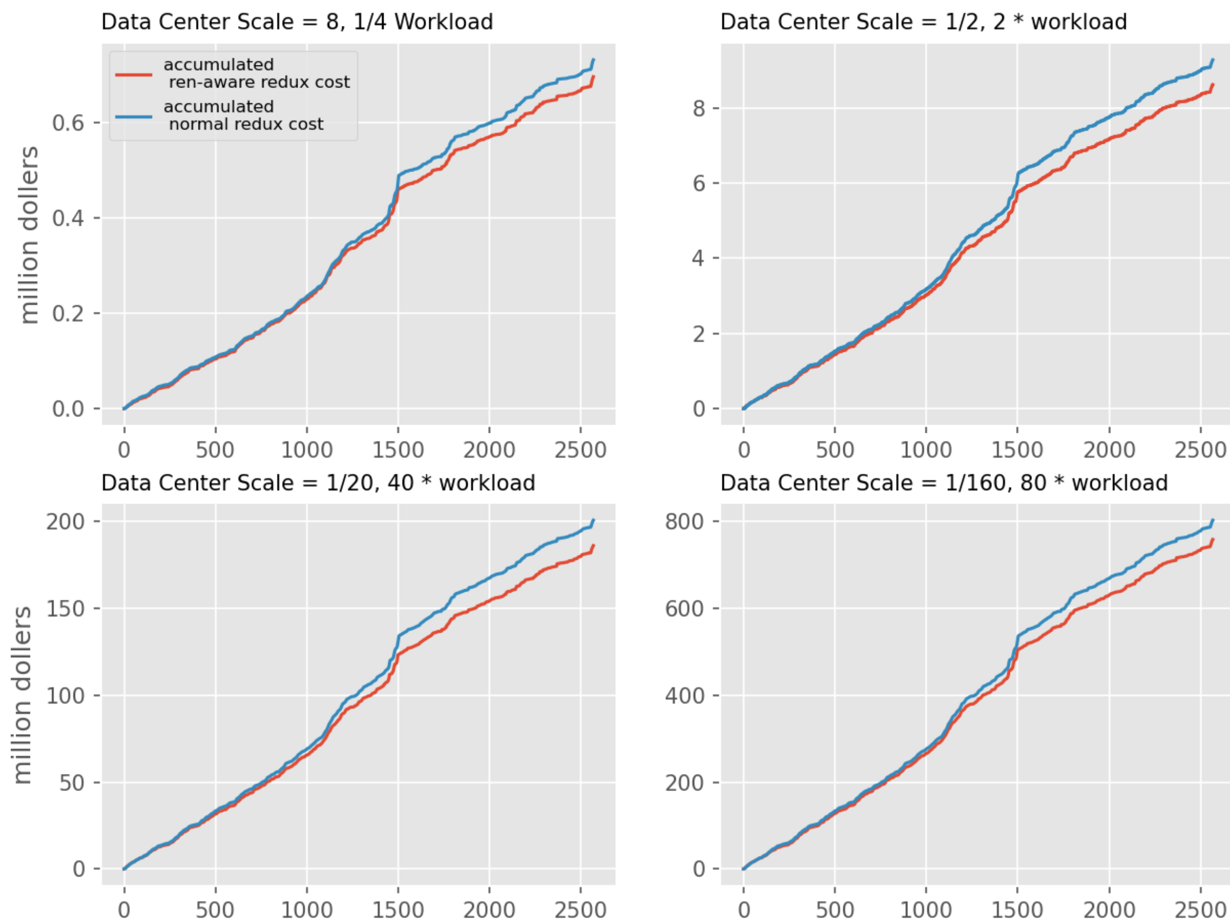

Figure 5.9: Comparison of Different Data Center Scale and Workload Multiplier

Fig. 5.9 depicts that the energy efficiency results of the four pair of scale multipliers share a similar pattern. We observe that the energy saving ratio obtained by REDUX3 becomes pronounced when data center is scaled with $1/2, 1/20$ with the workload multiplied by 2 and 40 of each time slot, respectively. The reason is two-fold. First, a mismatch between the scale of data center and workload may cause either a large number of idle processors or deferred non-urgent jobs. Second, the modules (scheduling window or job power profiling) and algorithm (defer or back-filling) of the renewable-aware scheduler may be negatively affected by non-preferred scale pairs.

5.8.4 Impacts of Timeline Length

In this set of experiments we prolong the length of timeline to six and nine months on the three previously visited workload trace. Together with the extended timeline, we collect nine-month grid price, solar and wind data to match with the augmented workload traces.

Data Trace	**3 Months**			**6 Months**			**9 Months**		
	Ren-aware	**Original**	**Cost Saving**	**Ren-aware**	**Original**	**Cost Saving**	**Ren-aware**	**Original**	**Cost Saving**
SDSC SP2	15.54	16.2	0.66	28.1	30.83	2.73	44.6	48.82	4.22
SDSC BLUE	23.53	24.2	0.67	36.96	39.68	2.72	59.54	63.63	4.09
Sandia ROSS	27.97	28.66	0.69	48.57	50.15	1.58	63.9	66.64	2.74

Table 5.7: Energy cost saving is enlarged gradually at the end of 3rd, 6th and 9th month

Fig. 5.10 unveils that the energy cost saving offered by REDUX3 becomes more noticeable when the timeline length is enlarged. For example, at the end of the ninth month with $SP2$, REDUX3 yields 4.22 million dollars of savings in energy cost consumed by the data centers. Similar gradually accumulated energy cost saving pattern from the other 2 workload trace can also be found in Table 5.7. The results confirm that the effectiveness of REDUX3 becomes remarkable during a long time period.

5.9 Summary

In this chapter, we presented a renewable-aware scheduler based data center energy efficiency system called *REDUX3* with on-site renewable sources coupled with a distributed

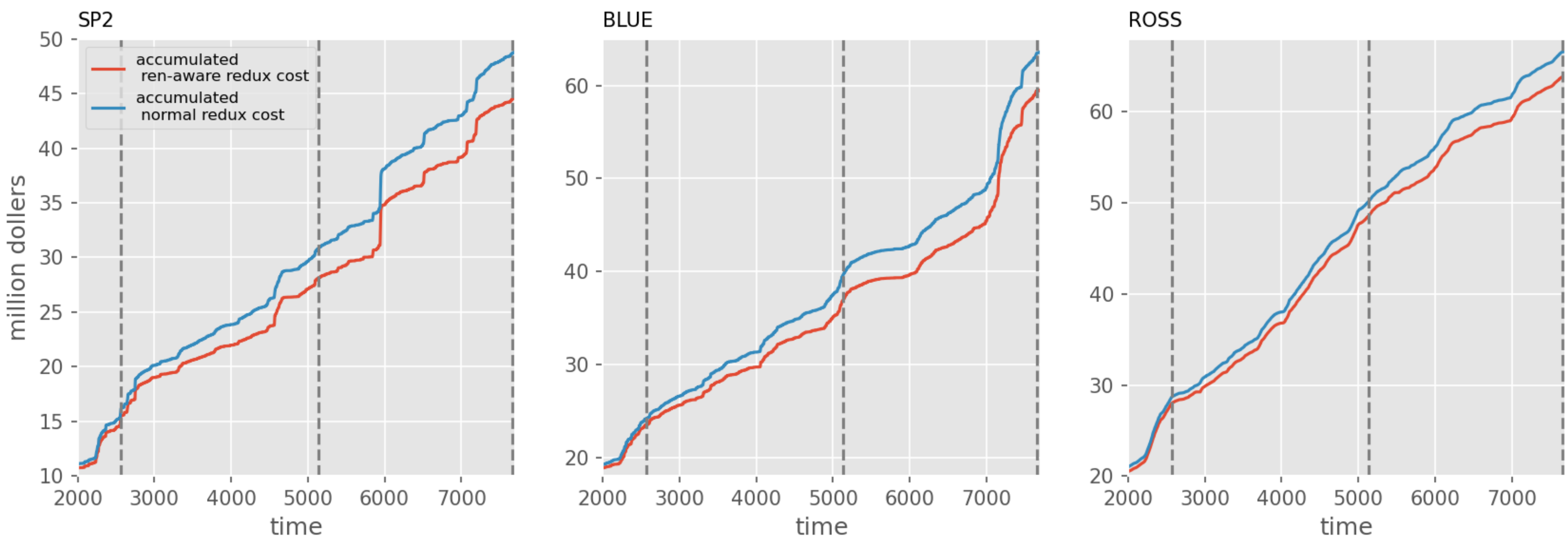

Figure 5.10: Extended timeline will expect higher energy efficiency

UPS system. The renewable-aware scheduler judiciously cope with the intermittent nature of renewable energy integrated with power grids. One promising feature of REDUX3 is to make data centers work in full capacity when inexpensive renewable energy is abundant and scale down workload when grid or UPS energy is the only choice. Compared with the prior solutions, REDUX3 conserves total energy cost while boosting renewable-energy utilization. We show that REDUX3 is an underpinning technique to build energy-efficient data centers.

Chapter 6

Conclusion

6.1 Main Contributions

6.1.1 Contributions of REDUX

In the first pilot study of this dissertation research (see also chapter 3), we proposed and devised a resource manager called *REDUX* to on-site renewable sources like solar coupled with a distributed UPS system to reduce energy cost of data centers. REDUX orchestrates UPS units to judiciously cope with the intermittency nature of renewable energy integrated with power grids. REDUX makes critical decisions on UPS charging and discharging with sufficient information on renewable-energy supply levels and time-dependent grid power prices. More specifically, REDUX makes prudent discharge of batteries to offer energy resources when renewable energy production is low or in the fluctuate condition. REDUX makes judicious decisions to charge batteries when renewable energy levels are high coupled with low loads. One salient feature of REDUX is to effectively slash energy costs of data centers powered by both renewable and conventional energy. Compared with the prior solutions, REDUX delivers a prominent capability of mitigating average peak workload and boosting renewable-energy utilization. The experimental results demonstrate that REDUX paves the way for constructing modern data centers that are economically and environmentally friendly.

6.1.2 Contributions of REDUX2

In the second part of the dissertation study (see also chapter 4), we presented a resource manager called *REDUX* to on-site renewable sources like solar coupled with a distributed

UPS system to reduce energy cost of data centers. REDUX orchestrates UPS units to judiciously cope with the intermittency nature of renewable energy integrated with power grids. REDUX makes critical decisions on UPS charging and discharging with sufficient information on renewable-energy supply levels and time-dependent grid power prices. More specifically, REDUX makes prudent discharge of batteries to offer energy resources when renewable energy production is low or in the fluctuate condition. REDUX makes judicious decisions to charge batteries when renewable energy levels are high coupled with low loads.

6.1.3 Contributions of REDUX3

As a third part of the dissertation research (see also chapter 5), we designed and developed a renewable-aware scheduler based data center energy efficiency system called *REDUX3* with on-site renewable sources coupled with a distributed UPS system. The renewable-aware scheduler judiciously cope with the intermittent nature of renewable energy integrated with power grids. One promising feature of REDUX3 is to make data centers work in full capacity when inexpensive renewable energy is abundant and scale down workload when grid or UPS energy is the only choice. Compared with the prior solutions, REDUX3 conserves total energy cost while boosting renewable-energy utilization. We show that REDUX3 is an underpinning technique to build energy-efficient data centers.

6.2 Future Work

6.2.1 Future Work for REDUX

There are two future research directions. First, we will unravel a plan to explore scheduling policies during peak load. We expect that the scheduling policies undoubtedly have a tremendous impact on energy efficiency and performance of data centers. Second, the smoothing algorithm implemented in REDUX is a pilot study towards optimizing renewable energy. To extend the workload shaving algorithm, we intend to propose a predictive scheme to make judicious workload-shaving decisions on renewable energy supplies.

6.2.2 Future Work for REDUX2

There are three valuable future research directions. First, we will unravel the most appropriate workload level that optimizes the energy efficiency of a data center along with predicted renewable energy sources. Second, we will unravel a plan to test our system using online big data processing workload conditions such as the *Google Cluster Workload Traces 2019*. Third, we intend to navigate alternative solutions for the optimization problem articulated in Section 4.3.6. More specifically, we will launch a concerted effort to incorporate reinforcement learning techniques into our problem statement. Finally, the smoothing algorithm implemented in REDUX2 is a pilot study towards optimizing renewable energy. To upgrade the UPS storage controller, we plan to incorporate a feedback-control strategy to optimize the UPS controller.

6.2.3 Future Work for REDUX3

We advocate for two future research directions. First, we intend to navigate alternative solutions for the optimization problem articulated in Section 4.3.6. More specifically, we will launch a concerted effort to incorporate Lyapunov optimization or reinforcement learning techniques into our problem statement. Second, the smoothing algorithm implemented in REDUX3 is an initial and bold step towards optimizing renewable energy usage. We intend to extend this algorithm by integrating prediction methods to make judicious smoothing decisions on renewable energy supplies.

Bibliography

[1] Accenture and WSP. Cloud computing and sustainability: The environmental benefits of moving to the cloud. Technical report, Accenture, 2010.

[2] B. Aksanli, T. Rosing, and E. Pettis. Distributed battery control for peak power shaving in datacenters. In *Green Computing Conference (IGCC), 2013 International*, pages 1–8. IEEE, 2013.

[3] B. Aksanli, J. Venkatesh, L. Zhang, and T. Rosing. Utilizing green energy prediction to schedule mixed batch and service jobs in data centers. *SIGOPS Oper. Syst. Rev.*, 45(3):53–57, Jan. 2012.

[4] M. Al Assaf, X. Jiang, M. Abid, and X. Qin. Eco-storage: A hybrid storage system with energy-efficient informed prefetching. *Journal of Signal Processing Systems*, 72(3):165–180, 2013.

[5] E. Azarkhish, D. Rossi, I. Loi, and L. Benini. Neurostream: Scalable and energy efficient deep learning with smart memory cubes. *IEEE Transactions on Parallel and Distributed Systems*, 29(2):420–434, 2018.

[6] E. Baccarelli, N. Cordeschi, A. Mei, M. Panella, M. Shojafar, and J. Stefa. Energy-efficient dynamic traffic offloading and reconfiguration of networked data centers for big data stream mobile computing: review, challenges, and a case study. *IEEE Network*, 30(2):54–61, 2016.

[7] A. Beloglazov, J. Abawajy, and R. Buyya. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future Generation Computer Systems*, 28(5):755 – 768, 2012. <ce:title>Special Section: Energy efficiency in large-scale distributed systems</ce:title>.

[8] A. Beloglazov and R. Buyya. Energy efficient resource management in virtualized cloud data centers. In *Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, CCGRID '10, pages 826–831, Washington, DC, USA, 2010. IEEE Computer Society.

[9] S. S. H. Bukhari, S. Atiq, and B.-i. Kwon. A sag compensator that eliminates the possibility of inrush current while powering transformer-coupled loads. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 5(2):891–900, 2017.

[10] S. S. H. Bukhari, S. Atiq, T. A. Lipo, and B.-i. Kwon. Line-interactive uninterruptible power supply system eliminating the inrush current phenomenon. *Electric Power Components and Systems*, 44(11):1203–1214, 2016.

[11] S. S. H. Bukhari, T. A. Lipo, and B.-i. Kwon. An online ups system that eliminates the inrush current phenomenon while feeding multiple load transformers. *IEEE Transactions on Industry Applications*, 53(2):1149–1156, 2016.

[12] J. M. Carrasco, L. G. Franquelo, J. T. Bialasiewicz, E. Galván, R. C. PortilloGuisado, M. M. Prats, J. I. León, and N. Moreno-Alfonso. Power-electronic systems for the grid integration of renewable energy sources: A survey. *IEEE Transactions on industrial electronics*, 53(4):1002–1016, 2006.

[13] D. Cheng, X. Zhou, P. Lama, M. Ji, and C. Jiang. Energy efficiency aware task assignment with dvfs in heterogeneous hadoop clusters. *IEEE Transactions on Parallel and Distributed Systems*, 29(1):70–82, 2018.

[14] D. Chiu, C. Stewart, and B. McManus. Electric grid balancing through lowcost workload migration. *SIGMETRICS Perform. Eval. Rev.*, 40(3):48–52, Jan. 2012.

[15] B.-G. Chun, G. Iannaccone, G. Iannaccone, R. Katz, G. Lee, and L. Niccolini. An energy case for hybrid datacenters. *ACM SIGOPS Operating Systems Review*, 44(1):76–80, 2010.

[16] D. Colarelli and D. Grunwald. Massive arrays of idle disks for storage archives. In *Proceedings of the 2002 ACM/IEEE conference on Supercomputing*, Supercomputing '02, pages 1–11, Los Alamitos, CA, USA, 2002. IEEE Computer Society Press.

[17] G. Cook. How clean is your cloud? Technical report, Greenpeace International, April 2012.

[18] M. Dabbagh, B. Hamdaoui, M. Guizani, and A. Rayes. Energy-efficient resource allocation and provisioning framework for cloud data centers. *IEEE Transactions on Network and Service Management*, 12(3):377–391, 2015.

[19] M. Dayarathna, Y. Wen, and R. Fan. Data center energy consumption modeling: A survey. *IEEE Communications Surveys & Tutorials*, 18(1):732–794, 2016.

[20] W. Deng, F. Liu, H. Jin, C. Wu, and X. Liu. Multigreen: Cost-minimizing multi-source datacenter power supply with online control. In *Proceedings of the Fourth International Conference on Future Energy Systems*, e-Energy '13, pages 149–160, New York, NY, USA, 2013. ACM.

[21] X. Deng, D. Wu, J. Shen, and J. He. Eco-aware online power management and load scheduling for green cloud datacenters. *IEEE Systems Journal*, 10(1):78–87, 2016.

[22] K. Ebrahimi, G. F. Jones, and A. S. Fleischer. A review of data center cooling technology, operating conditions and the corresponding low-grade waste heat recovery opportunities. *Renewable and Sustainable Energy Reviews*, 31:622–638, 2014.

[23] G. Falcao, L. A. Alexandre, J. Marques, X. Frazão, and J. Maria. On the evaluation of energy-efficient deep learning using stacked autoencoders on mobile gpus. In *Parallel, Distributed and Network-based Processing (PDP), 2017 25th Euromicro International Conference on*, pages 270–273. IEEE, 2017.

[24] Y. Fan, Z. Lan, P. Rich, W. E. Allcock, M. E. Papka, B. Austin, and D. Paul. Scheduling beyond cpus for hpc. In *Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing*, pages 97–108, 2019.

[25] D. Feitelson. Parallel workloads archive.

[26] D. G. Feitelson, D. Tsafrir, and D. Krakov. Experience with using the parallel workloads archive. *Journal of Parallel and Distributed Computing*, 74(10):2967–2982, 2014.

[27] M. Geveler, D. Ribbrock, D. Donner, H. Ruelmann, C. Höppke, D. Schneider, D. Tomaschewski, and S. Turek. The icarus white paper: a scalable, energy-efficient, solar-powered hpc center based on low power gpus. In *European Conference on Parallel Processing*, pages 737–749. Springer, 2016.

[28] M. Ghamkhari and H. Mohsenian-Rad. Energy and performance management of green data centers: A profit maximization approach. *IEEE Transactions on Smart Grid*, 4(2):1017–1025, 2013.

[29] P. Gipe. Wind power: renewable energy for home, farm, and business, chelsea green pub, 2005.

[30] Í. Goiri, R. Beauchea, K. Le, T. D. Nguyen, M. E. Haque, J. Guitart, J. Torres, and R. Bianchini. Greenslot: scheduling energy consumption in green datacenters. In *High Performance Computing, Networking, Storage and Analysis (SC), 2011 International Conference for*, pages 1–11. IEEE, 2011.

[31] Í. Goiri, M. E. Haque, K. Le, R. Beauchea, T. D. Nguyen, J. Guitart, J. Torres, and R. Bianchini. Matching renewable energy supply and demand in green datacenters. *Ad Hoc Networks*, 25:520–534, 2015.

[32] Í. Goiri, W. Katsak, K. Le, T. D. Nguyen, and R. Bianchini. Parasol and greenswitch: Managing datacenters powered by renewable energy. In *ACM SIGARCH Computer Architecture News*, volume 41, pages 51–64. ACM, 2013.

[33] S. Govindan, A. Sivasubramaniam, and B. Urgaonkar. Benefits and limitations of tapping into stored energy for datacenters. In *ACM SIGARCH Computer Architecture News*, volume 39, pages 341–352. ACM, 2011.

[34] S. Govindan, D. Wang, L. Chen, A. Sivasubramaniam, and B. Urgaonkar. Towards realizing a low cost and highly available datacenter power infrastructure. In *Proceedings of the 4th Workshop on Power-Aware Computing and Systems*, page 7. ACM, 2011.

[35] L. Grange, G. Da Costa, and P. Stolf. Green it scheduling for data center powered with renewable energy. *Future Generation Computer Systems*, 86:99–120, 2018.

[36] Y. Guo, Y. Gong, Y. Fang, P. P. Khargonekar, and X. Geng. Energy and network aware workload management for sustainable data centers with thermal storage. *IEEE Transactions on Parallel and Distributed Systems*, 25(8):2030–2042, 2014.

[37] S. K. S. Gupta, A. Banerjee, Z. Abbasi, G. Varsamopoulos, M. Jonas, J. Ferguson, R. R. Gilbert, and T. Mukherjee. Gdcsim: A simulator for green data center design and analysis. *ACM Trans. Model. Comput. Simul.*, 24(1):3:1–3:27, Jan. 2014.

[38] V. Hanumaiah and S. Vrudhula. Energy-efficient operation of multicore processors by dvfs, task migration, and active cooling. *IEEE Transactions on Computers*, 63(2):349–360, 2014.

[39] D. Klusáček, Š. Tóth, and G. Podolníková. Real-life experience with major reconfiguration of job scheduling system. In *Job Scheduling Strategies for Parallel Processing*, pages 83–101. Springer, 2015.

[40] V. Kontorinis, L. E. Zhang, B. Aksanli, J. Sampson, H. Homayoun, E. Pettis, D. M. Tullsen, and T. S. Rosing. Managing distributed ups energy for effective power capping in data centers. In *Computer Architecture (ISCA), 2012 39th Annual International Symposium on*, pages 488–499. IEEE, 2012.

[41] J. G. Koomey. Worldwide electricity used in data centers. *Environmental research letters*, 3(3):034008, 2008.

[42] S. Kwon. Ensuring renewable energy utilization with quality of service guarantee for energy-efficient data center operations. *Applied Energy*, 276:115424, 2020.

[43] Y. Lee and A. Zomaya. Energy efficient utilization of resources in cloud computing systems. *The Journal of Supercomputing*, 60(2):268–280, 2012.

[44] Y. C. Lee and A. Y. Zomaya. Energy efficient utilization of resources in cloud computing systems. *J. Supercomput.*, 60(2):268–280, May 2012.

[45] C. Li, A. Qouneh, and T. Li. iswitch: coordinating and optimizing renewable energy powered server clusters. In *Computer Architecture (ISCA), 2012 39th Annual International Symposium on*, pages 512–523. IEEE, 2012.

[46] C. Li, R. Wang, T. Li, D. Qian, and J. Yuan. Managing green datacenters powered by hybrid renewable energy systems. In *ICAC*, pages 261–272, 2014.

[47] J. Li, Z. Li, K. Ren, and X. Liu. Towards optimal electric demand management for internet data centers. *IEEE Transactions on Smart Grid*, 3(1):183–192, 2012.

[48] L. Liu, H. Sun, C. Li, Y. Hu, J. Xin, N. Zheng, and T. Li. Re-ups: an adaptive distributed energy storage system for dynamically managing solar energy in green datacenters. *The Journal of Supercomputing*, 72(1):295–316, 2016.

[49] T. H. Nguyen, M. Di Francesco, and A. Yla-Jaaski. Virtual machine consolidation with multiple usage prediction for energy-efficient cloud data centers. *IEEE Transactions on Services Computing*, 2017.

[50] L. Olatomiwa, S. Mekhilef, M. Ismail, and M. Moghavvemi. Energy management strategies in hybrid renewable energy systems: A review. *Renewable and Sustainable Energy Reviews*, 62:821–835, 2016.

[51] X. Peng, T. Bhattacharya, T. Cao, J. Mao, T. Tekreeti, and X. Qin. Exploiting renewable energy and ups systems to reduce power consumption in data centers. *Big Data Research*, 27:100306, 2022.

[52] X. Peng, C. Kauten, C. Zhang, J. Mao, T. Heckwolf, T. K. Al Tekreeti, and X. Qin. Redux: Managing renewable energy in data centers using distributed ups systems. In *2018 IEEE International Conference on Smart Cloud (SmartCloud)*, pages 46–53. IEEE, 2018.

[53] X. Peng and X. Qin. Energy efficient data centers powered by on-site renewable energy and ups devices. In *2020 11th International Green and Sustainable Computing Workshops (IGSC)*, pages 1–3. IEEE, 2020.

[54] E. Pinheiro and R. Bianchini. Energy conservation techniques for disk array-based servers. In *Proceedings of the 18th annual international conference on Supercomputing*, ICS '04, pages 68–78, New York, NY, USA, 2004. ACM.

[55] S. Ren and Y. He. Coca: Online distributed resource management for cost minimization and carbon neutrality in data centers. In *Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, SC '13, pages 39:1–39:12, New York, NY, USA, 2013. ACM.

[56] J. Richard and J. Komp. Practical photovoltaics electricity from solar cells, 2001.

[57] M. Rodríguez-Pascual, A. J. Rubio-Montero, J. A. Moríñigo, and R. Mayo-García. Execution data logs of a supercomputer workload over its extended lifetime. *Data in brief*, 28:105006, 2020.

[58] E. K. Sato, M. Kinoshita, Y. Yamamoto, and T. Amboh. Redundant high-density high-efficiency double-conversion uninterruptible power system. *IEEE Transactions on Industry Applications*, 46(4):1525–1533, 2010.

[59] M. Sharifi, H. Salimi, and M. Najafzadeh. Power-efficient distributed scheduling of virtual machines using workload-aware consolidation techniques. *J. Supercomput.*, 61(1):46–66, July 2012.

[60] A. Shehabi, S. Smith, D. Sartor, R. Brown, M. Herrlin, J. Koomey, E. Masanet, N. Horner, I. Azevedo, and W. Lintner. United states data center energy usage report. 2016.

[61] W. Tian, Q. Xiong, and J. Cao. An online parallel scheduling method with application to energy-efficiency in cloud computing. *J. Supercomput.*, 66(3):1773–1790, Dec. 2013.

[62] H.-W. Tseng, T.-T. Yang, K.-C. Yang, and P.-S. Chen. An energy efficient vm management scheme with power-law characteristic in video streaming data centers. *IEEE Transactions on Parallel and Distributed Systems*, 29(2):297–311, 2018.

[63] R. Urgaonkar, B. Urgaonkar, M. J. Neely, and A. Sivasubramaniam. Optimal power cost management using stored energy in data centers. In *Proceedings of the ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems*, pages 221–232. ACM, 2011.

[64] S. Wallace, X. Yang, V. Vishwanath, W. E. Allcock, S. Coghlan, M. E. Papka, and Z. Lan. A data driven scheduling approach for power management on hpc systems. In *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 656–666. IEEE, 2016.

[65] D. Wang, C. Ren, A. Sivasubramaniam, B. Urgaonkar, and H. Fathy. Energy storage in datacenters: what, where, and how much? In *ACM SIGMETRICS Performance Evaluation Review*, volume 40, pages 187–198. ACM, 2012.

[66] J. Whitney and J. Kennedy. Is cloud computing always greener? Technical report, Natural Resources Defense Council, October 2012.

[67] L. Xie, Y. Gu, X. Zhu, and M. G. Genton. Short-term spatio-temporal wind power forecast in robust look-ahead power system dispatch. *IEEE Transactions on Smart Grid*, 5(1):511–520, 2014.

[68] X. Yang, Z. Zhou, S. Wallace, Z. Lan, W. Tang, S. Coghlan, and M. E. Papka. Integrating dynamic pricing of electricity into energy aware scheduling for hpc systems. In *SC'13: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*, pages 1–11. IEEE, 2013.

[69] M. Zaharia, D. Borthakur, J. Sen Sarma, K. Elmeleegy, S. Shenker, and I. Stoica. Delay scheduling: a simple technique for achieving locality and fairness in cluster scheduling. In *Proceedings of the 5th European conference on Computer systems*, pages 265–278. ACM, 2010.

[70] M. Zapater, J. L. Ayala, and J. M. Moya. Leveraging heterogeneity for energy minimization in data centers. In *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (Ccgrid 2012)*, CCGRID '12, pages 752–757, Washington, DC, USA, 2012. IEEE Computer Society.

[71] G. Zhang, S. Zhang, W. Zhang, Z. Shen, and L. Wang. Distributed energy management for multiple data centers with renewable resources and energy storages. *IEEE Transactions on Cloud Computing*, 2020.

[72] Z. Zhou, Z. Lan, W. Tang, and N. Desai. Reducing energy costs for ibm blue gene/p via power-aware job scheduling. In *Workshop on Job Scheduling Strategies for Parallel Processing*, pages 96–115. Springer, 2013.