Leveraging Workload Relocation and Resource Pruning for Electricity Cost Minimization in Service Provider Networks

PhD Thesis

Muhammad Saqib Ilyas

2005-06-0024

Advisor: Zartash Afzal Uzmi

Department of Computer Science

Syed Babar Ali School of Science and Engineering

Lahore University of Management Sciences

Dedicated to dedication

Lahore University of Management Sciences

School of Science and Engineering

CERTIFICATE

I hereby recommend that the thesis prepared under my supervision by ***Muhammad Saqib Ilyas*** titled ***Leveraging Workload Relocation and Resource Pruning for Electricity Cost Minimization in Service Provider Networks*** be accepted in partial fulfillment of the requirements for the degree of doctor of philosophy in computer science.

Zartash Afzal Uzmi (Advisor)

Recommendation of Examiners' Committee:

Name	**Signature**
Zartash Afzal Uzmi	_____
Ihsan Ayyub Qazi	_____
Tariq Mehmood Jadoon	_____

Acknowledgements

Contents

List of Figures

List of Tables

Abstract

Service provider networks enable services that we rely on for many essential everyday tasks. These networks are shared by many users and must be able to handle the cumulative workload from all the users at any given time. These networks are composed of several network resources[1], each of which has a maximum workload handling capacity. Operators, therefore, dimension networks with enough network resources so that they may handle the expected peak of the cumulative customer workload. The customer workload is time-varying and has a large peak-trough ratio. Since network resources lack energy proportionality, networks always consume electricity at about the same level as the peak power consumption. This leads to wasted electric energy, which this thesis aims to reduce.

We propose saving electricity by using a two-pronged strategy. First, we reduce power consumption during low workload regimes by keeping as many network components off, or in power-saving state, as possible without compromising handling of current workload. Secondly, by smartly distributing workload among network components, we aim to maximize the number of network components turned off or in power-saving state. We term these strategies as resource pruning and workload relocation, respectively.

Both resource pruning and workload relocation control the state of network resources, i.e., on/off/power-saving and current workload assigned to each resource. The network resource state, in turn, determines the network's power consumption. We may consider the aggregate of the instantaneous state of all network resources as the network's state. Due to workload variations, no single network state can be optimal for a network at all times. Therefore, we formulate the energy efficiency improvement problem as a multi-interval optimal state trajectory problem, called RED-BL: Relocate Energy Demand to Better Locations. RED-BL

[1]such as servers, radio transceivers, network links and routers

computes the optimal states for a network over a time horizon by using workload estimates, future electricity prices and the cost of transition between network states during consecutive intervals.

We evaluated the benefit of RED-BL using real datasets obtained from geo-diverse data centers as well as cellular networks. Our results indicate that significant savings in electricity consumption and cost may be obtained by the application of RED-BL to these types of networks. In case of geo-diverse data centers, RED-BL can reduce electricity costs by as much as 45%. In case of cellular networks, the energy savings were as high as 22%.

Chapter 1

Introduction

1.1 Computer networks pervade

Services such as telephony, email and the world wide web (WWW) are a seamless part of our everyday lives. We communicate and collaborate using email, voice/video calls over the Internet and online social networks. We also use web-based systems to access teaching/learning material, course registration systems on campus and even pathological examination reports.

All of these services, and many more, are powered by several computer networks. These networks are run by commercial entities known as service providers or network operators. Examples of such network operators are:

- **Cellular Network Operators:** Cellular network operators deploy and run a cellular network infrastructure. Computing plays a significant role in cellular networks not only in the form of smart phones as end user devices but also in the form of servers that run critical functions within the network.

  The most common services offered by these operators are voice calls and short text messages. Whereas the predominant means for an end user to connect to the Internet was dialup over Public Switched Telephone Network (PSTN) until a few years ago,

nowadays cellular networks are quite commonly used to access Internet resources. In this context, the cellular network merely acts as an access mechanism to connect to the Internet.

- **Internet Service Providers (ISPs):** The Internet itself is an interconnection of ISP networks. End users connect to the ISP's network in order to gain access to resources on the Internet. The end user's connection to an ISP's network is typically based on a periodic paid subscription.

  An ISP's network is a mostly dumb but fast carrier of IP packets, also known as IP traffic, from one point to another. The intelligence of Internet applications and the information that the Internet is so popular for, are provided on the edges – by software running on the user's device or in the networks run by other service providers such as the ones that follow.

- **Geo-diverse data center network operators**: Today's web-based services such as Google Search and Facebook have such a large subscriber base that a huge number of servers is required to run these services. To enable such services, companies such as Facebook, Amazon, Microsoft and Google operate geo-diverse data centers. Servers in these data center networks run applications such as Google Search, GMail, Youtube, Twitter, Bing and Facebook.

- **Content Distribution Network (CDN) operators:** CDNs place multiple copies of Internet resources such as web pages across the globe. The role of CDNs is to lower the latency from a user to an Internet resource. For instance, if Google's home page were only located at a server in Mountain View, CA, the latency (the time it takes for a web browser to send a packet to the server) for users in Pakistan would be hundreds of milliseconds. Placing a replica of the Google home page close to Pakistan lowers the packet latency significantly, thereby allowing the web browser to display the page

much faster.

So far, we have seen that computer networks are critical resources that enable the services that we rely on in our everyday lives. In this thesis, we will focus primarily on cellular networks and geo-diverse data centers.

The deployment of these networks involves huge expenses. For instance, Google announced building a data center in Iowa at a cost of $400 Million [1]. Furthermore, according to [2], the capital cost of a typical cellular network site is $550,000[1].

The recurring operational cost of these networks is also quite high. For instance, in 2009, Facebook spent $50 Million on leasing the data center space, alone [3]. In the context of geo-diverse data centers, other contributors to operational expenses include staff salaries, maintenance costs, the cost of inter-data center network connectivity and the electricity bill. Optimizing operational costs is critical for network operators in order to offer cost-effective services and increase their profit.

1.2 Electricity costs for operational networks

For many computer networks, electricity costs account for a significant fraction of operational costs. For instance, electricity costs may be as much as 15% of operational costs in data centers [4]. For cellular networks, in European markets, the electricity cost is estimated to be around 18% of the operational costs [5]. This fraction is even higher in developing regions due to the shortage of grid electricity and the use of small-scale generation powered by diesel fuel. Even when ignoring the cost of diesel-generated electricity, the annual electricity cost for an operator in Pakistan with 7000 cellular sites in Pakistan can be roughly estimated at $9.19 Million[2]. Telecom Italia reported a consumption of 1.793 GWh in 2012 [7], which is

[1]This does not include spectrum licensing costs. Furthermore, an operator needs to deploy many sites. A site at about every 800 meters is common in urban settings

[2]Using a 1.5 kW draw for a single cellular site [6], Rs. 10 per kWh and Rs. 100 per US$. Note that the Rs. 10 per kWh is a gross under-estimation, given that it considers only the approximate current price of

significantly higher compared to our estimates for the Pakistani network operator and hence the electricity costs are also expected to be much higher.

1.3 The impact of energy inefficiency on operational networks' electricity costs

Ideally, a network's power consumption should be a linear function of workload such as indicated by the dotted line in Figure 1.1. The network should consume no power when there is no workload and in the presence of workload, its power consumption should be proportional to the network's utilization. However, for most operational networks, the power consumption is well-approximated as an affine function of workload [8, 9], as shown by the solid line in Figure 1.1. We describe this by saying that today's networks are not energy proportional, i.e., they operate at a large fraction of their peak power consumption under no load. There are two reasons for high power consumption in the absence of workload.

1. In some cases, the utilization of network hardware is nearly independent of workload. For instance, in order to allow subscribers to sense the availability of the network, it may be necessary to transmit beacon frames with no payload even when there is no user traffic. Thus, the network activity under no workload conditions is not significantly less than that under peak workload. A cellular network's radio components, for instance, must continue operating and drawing power to offer uninterrupted connectivity to subscribers, even when no call is in progress. In packet switching networks, many data link layer technologies continuously transmit frames with no payload, irrespective of traffic activity. Since the network hardware is continuously transmitting signals, its power consumption under no load is quite high.

the intermittently available grid power and does not factor in the cost of electricity produced using diesel generators during rolling blackouts. The cost of electricity generated through diesel generates is several times higher per kWh than the grid power.

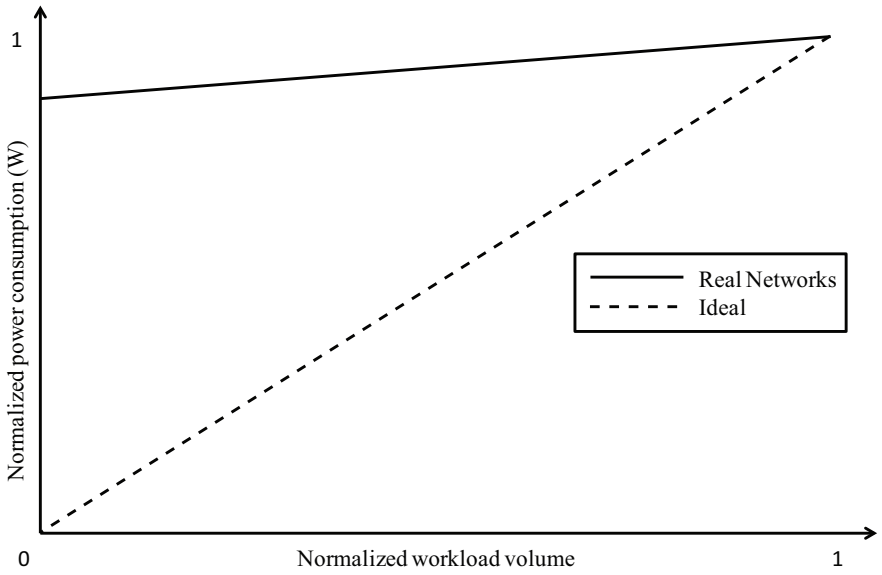


Figure 1.1: Lack of energy proportionality in computer networks

2. The components of the network may not be energy proportional. For instance, in data centers, server power consumption is a large fraction of the total power consumption [4] and server idle power consumption is a large fraction of their peak power consumption.

Lack of energy proportionality is of little concern if the network consistently operates in the high workload region because in this region, power consumption for a real network is close to ideal. However, most networks today have time-varying traffic which is in the low-workload region for considerable amount of time during a given day. Figure 1.2 shows the workload for call traffic at a cellular site in a large operational GSM network in Pakistan. It shows that call traffic has diurnal cycles and that traffic peaks for only a short period of time during a day. Furthermore, the workload peak is quite high compared to the trough. ISP and data center traffic also show similar trends [10, 11].

In order to meet peak expected workload amicably, networks are dimensioned according to the peak workload. Since the workload is far from the peak most of the time and networks are not energy proportional, most real networks consume a lot more energy than an ideal one. In other words, today's networks are energy inefficient. Another way to look at energy

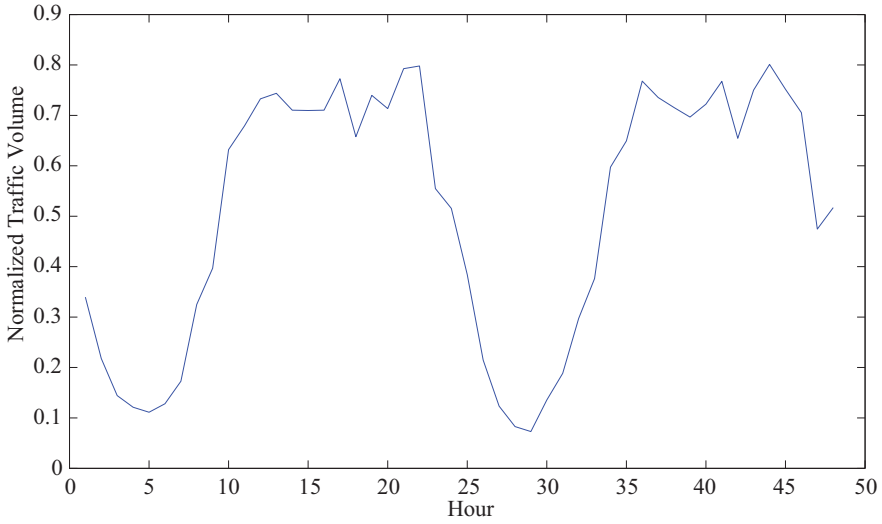


Figure 1.2: Call traffic for an operational cellular site over two days

efficiency is through the lens of performance per Watt – a useful metric for energy efficiency of a network. The performance per Watt for today's networks is quite low. Therefore, it is important to lower the y-intercept of the solid line in Figure 1.1 thereby reducing the electricity cost of today's networks without compromising on the performance that they deliver. Recent years have seen significant research focus on improving energy efficiency in both cellular and geo-diverse data center networks.

1.4 Prevalent electricity cost reduction techniques

The electricity cost for a network is given by:

$$\text{Electricity cost} = \text{amount of energy consumed} \times \text{unit price of electricity} \tag{1.1}$$

Recent work in reducing electricity cost for geo-diverse data centers and mobile cellular networks falls into two categories. The first category focuses on reducing the amount of energy consumed (thereby reducing the first term in equation 1.1), whereas the second

category focuses on using cheaper sources of electricity (thereby reducing the second term in equation 1.1). A taxonomy of the techniques that fall in these two categories is given in the next two sections.

1.4.1 Reducing the amount of energy consumed

1. **Hardware upgrades:** Due to ecological challenges, improved energy efficiency is generally a key requirement when developing new technologies and devices. For a given workload demand, an improvement in device energy efficiency lowers the amount of energy consumed. Thus, upgrading to more energy-efficient hardware is a way to reduce electricity costs. An operator would, however, opt for hardware upgrades in their network only after they have obtained the Return on Investment (ROI) of the initial deployment or if the deployed equipment has reached the end of life. The initial investment not only involves capital cost of equipment but other factors such as spectrum licensing as well. In the cut-throat competition prevalent in most of today's networks, the ROI is slow to achieve. This means that existing energy inefficient networks would stay that way for a considerable time into the future.

2. **Hardware virtualization:** With the advent of ever faster CPUs, it was observed that servers tend to operate at relatively low CPU utilization most of the time [12]. This was seen as an opportunity to statistically multiplex multiple servers onto a single physical machine by slicing the latter into multiple virtual servers. In this way, virtualization cuts capital costs for procurement of hardware. Since the virtual servers share the same resources (power supply, CPU, network interface, disks), if two servers are multiplexed onto a single physical server, the electricity consumption may be cut by as much as 50%. A more aggressive server consolidation may cut electricity costs by upto 80% [13].

3. **Resource Pruning (RP):** Since network resources must be deployed according to

peak demand while the workload peaks only for a short period of time, the excess resources may be deactivated (shutdown or put in power-saving mode depending on what is supported by the equipment) when the workload is low [14, 15, 16, 17, 8]. When evaluating the reduction in electricity costs through resource pruning, it is imperative to consider any costs associated with activation and deactivation of network resources. For instance, it is pointless to use RP if 1 kWh energy is saved whereas 2 kWh must be spent on deactivating the network resource. RP must only be applied if the electricity cost saved exceeds the overhead of activating or deactivating the network resource. It is important to note here that in case of data centers, in this thesis, we are only considering workload resulting from customer traffic. The workload does originate from within the data centers, such as background tasks for web indexing in a web search cluster are not considered. This is because we only had access to customer workload traces. Our assumption is that the peak and trough of workload are significantly different. If the background workload is high and arrives when the customer workload is low, this assumption may not be true.

1.4.2 Using cheaper electricity - Workload Relocation (WR)

Electricity prices exhibit geographic diversity [18, 19, 20, 21, 22, 23, 24], i.e., the price of electricity varies from one location to another. Figure 1.3 shows a week long trace of the *day-ahead* electricity prices at two locations in the US. Most of the time, we see that the electricity price is different at the two locations. The variation in electricity price is generally noticeable only at large distances. For instance, the electricity price anywhere within a city is generally the same[3], but electricity price at a given time may vary from one city to another. Most networks span large enough distances for geographic diversity in electricity prices to be apparent between a given pair of network sites.

[3]With the exception of factors such as different tariffs for domestic, commercial and industrial consumers

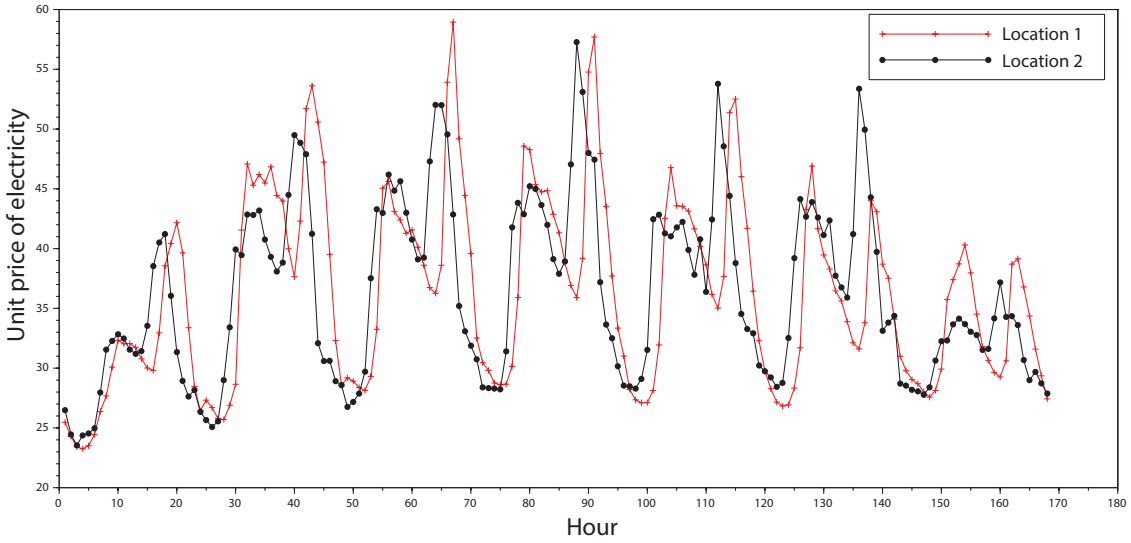


Figure 1.3: Hourly variation in electricity prices for two different locations in the US over a period of one week

If the network workload is quite flexible in terms of where it is handled, we say that the workload is geo-flexible. By relocating geo-flexible workload originating at a location with high electricity price to another location with lower electricity price we may cut the electricity cost [19, 20, 21, 22, 23, 24]. We call this technique Workload Relocation (WR). This is only applicable if one of the conditions hold:

- The network site B to which traffic is redirected replicates the site A from which traffic is redirected. For instance, data center B has all the content that data center A did.
- Content or applications are replicated when traffic needs to be redirected. This may be possible using lightweight container technology such as Kubernetes [25] or Docker [26].

Network workload exhibits varying levels of geo-flexibility. In geo-diverse data centers, for instance, the workload is highly geo-flexible, i.e., a client's request may be handled close by or even hundreds of miles away. On the other hand, the workload in cellular networks has very low geo-flexibility, i.e., a call must be handled at a cellular base station within a few hundred meters from the caller. Therefore, the amount of electricity cost savings

achievable by exploiting geo-diversity in electricity prices through the use of WR is expected to be higher in data center networks than in cellular networks. However, while exploiting geo-diversity in electricity prices to save electricity cost in data center networks, if the client latency to the serving data center increases, revenue losses may occur [27, 28, 29].

Electricity prices also exhibit temporal diversity [18], i.e., the relative order of electricity prices at different locations keeps changing. In Figure 1.3, sometimes location 1 is cheaper than location 2 and sometimes location 2 is cheaper than location 1. This means that mapping of workload to locations must be periodically updated. The granularity of these updates depends on how frequently electricity prices change. Deregulated electricity markets, such as the ones in the USA, exhibit price changes at two different time scales (15 minutes for real-time electricity prices and an hour for day-ahead prices).

1.5 Energy efficiency improvement using WR and RP

An operator may use RP to improve the network's energy efficiency, thereby reducing electricity cost for a given workload. Furthermore, an operator may use WR to exploit geo-temporal variations in electricity price to reduce electricity costs. A joint electricity cost optimization using WR and RP together is expected to provide greater electricity cost savings than either WR or RP alone. As we shall see in later chapters, this is indeed the case.

A precise statement of the above optimization problem is: Given a forecast of workload and the electricity prices at each network site over a future time horizon, the problem is to determine, for every instant in the future time window:

- How many resources should be on at each network site?
- How much workload should be assigned to each network site?

Since each network resource has a discrete workload capacity[4], the future time horizon may be viewed as consisting of several discrete intervals where within an interval, a particular integer number of resources is sufficient to handle the workload during that interval. For convenience of modeling and analysis, the future time horizon may be viewed as being composed of discrete intervals of equal duration during which the workload does not change.

For every discrete interval in the future horizon, the workload may be distributed among the network sites in different proportions. Thus, there can be many different feasible states for each interval. This leads to an optimal state trajectory representation of the electricity cost minimization problem. The cost of electricity associated with a particular distribution of workload among network sites can be modeled as the state cost in the optimal state trajectory representation. State transition costs due to shifting of workload or turning on/off of network resources between states in two consecutive intervals may also exist.

The optimal state trajectory problem is a combinatorial discrete optimization problem. We have modeled it as a Mixed Integer Program (MIP) and solved it using CPLEX solver which is part of the CPLEX Studio tool that is freely available for academic purposes. We have also proposed heuristic algorithms to approximately solve the optimization problem.

1.6 Our thesis

Based on the similarity in workload characteristics and the dependence of power consumption on workload, we opine that a generalized power optimization problem may be formulated that is applicable to many different types of networks. In this thesis, we focus specifically on geo-diverse data center networks and cellular networks. Our generalized electricity cost optimization formulation would use workload relocation and resource pruning in tandem to reduce electricity costs.

[4]A transceiver in a GSM cellulr network can handle upto 8 simultaneous calls. Similarly, a server is capable of amicably handling a certain integer number of clients for a particular application.

1.7 Contributions

This thesis makes the following contributions:

- We present a generalized model for electricity cost optimization called Relocate Energy Demand to Better Locations (RED-BL), pronounced Red Bull. RED-BL jointly uses workload relocation and resource pruning. We believe that RED-BL is applicable to different types of networks.

- We apply RED-BL to optimize electricity costs in geo-diverse data centers as well as cellular networks. For both of these networks we show that minimizing electricity costs using WR and RP is NP-Hard.

- We use real data traces to obtain optimal solutions to RED-BL for both data center and cellular networks for realistic problem sizes. While we were able to solve these problems within reasonable time, given the NP-Hard nature of the optimization problem, it may take much longer to optimally solve the problem using some other datasets (even those of the same size). To handle the intractability of the problem, we also propose some heuristics for approximate solutions.

- Prior efforts in this area had mostly ignored the costs associated with activation and deactivation of network resources. To the best of our knowledge, we are the first to incorporate these in our optimization problem.

- A network with significant over-provisioning may handle most of the workload at cheaper locations while the more expensive locations may be temporarily pruned from the network. In other words, geographic diversity in electricity prices incentivises over-provisioning. We study the benefits of increased over-provisioning and find diminishing returns with over-provisioning.

1.8 Organization

The rest of the document is structured as follows. In Chapter 2, we compare two different types of networks and describe how similar they are in terms of workload handling and power consumption. In chapter 3, we derive a generalized power consumption model, applicable to different types of networks and formulate RED-BL, a generalized electricity cost optimization problem. We present an evaluation of RED-BL on geo-diverse data centers and cellular networks in chapters 4 and 5, respectively. In chapter 6, we draw the conclusions and provide additional future directions.

Chapter 2

Background - Power consumption models in service provider networks

In this chapter, we first describe the power consumption models for geo-diverse data centers and mobile cellular networks. Then, we draw similarities between these two networks to come up with an abstract power consumption model. This generalized power consumption model motivates the formulation of a generalized electricity cost optimization framework in the next chapter.

2.1 Networks of geo-diverse data centers - enablers of cloud computing

Traditionally, computational resources are procured, deployed and managed. Authors needing to electronically typeset manuscripts would buy a PC with word-processing software. Organizations that need to deploy an Enterprise Resource Planning (ERP) system would buy servers and install the required software on these. This involves upfront expenditure on equipment and software licensing costs as well as recurring expenses resulting from fac-

tors such as renewal of software licenses, salaries for staff to smoothly run the software and training to customize and operate the newly acquired software.

For much of our non-IT needs, such as electricity, water and gas, we are used to the utility model where we don't own the resources. We pay a per-use subscription charge to some organization that owns and manages the resources for us. Due to economies of scale, utilities are economical for the provider and the end user. It was envisioned that computing could also be offered as a utility, to be consumed and (sometimes) paid for, as needed. This is the vision of cloud computing. Since our computing needs are met somewhere out there, where we don't know how it is all managed and run, we call it the cloud.

Based on the flexibility and sophistication of the service being offered, cloud computing has various service models, as described below:

- **Infrastructure as a Service (IaaS)**: In this model, the consumer gets access to one or more servers hosted and managed by the service provider. The consumer is responsible for installing the Operating System (OS) and any other software according to their requirements. Amongst all cloud computing models, this one offers the greatest flexibility to the end user. The end user can choose which OS they want to run. They can deploy their own customized applications on the server. This also means that the great responsibility of server software management lies with the service subscriber. Amazon Web Services (AWS), Microsoft Azure and Google Compute Engine are examples of IaaS.

- **Platform as a Service (PaaS)**: In this model, the service provider offers a complete computing platform with a pre-installed OS. The computing platform generally also includes other pre-installed software such as a database management system (DBMS) and programming environment. The service consumer can deploy their required applications on the platform as long as it meets the application's software requirements.

Microsoft Azure offers PaaS model services. Hosted web servers are another example of PaaS.

- **Software as a Service (SaaS)**: A service user often wishes to be least concerned with software installation, configuration and maintenance. The user just wishes to access an application remotely and seamlessly. The cloud computing model for such cases is SaaS. Web-based email services such as GMail are a common example of SaaS.

In order to offer cloud computing, cloud operators deploy large facilities called data centers. A data center may have tens of thousands of servers to provide computing as a utility. Cloud operators typically deploy multiple data centers at different geographic locations. Figure 2.1 shows the locations of Google's data centers across the globe as of August 10, 2014 (according to www.google.com/about/datacenters/inside/locations/).

The geographic distribution of data centers is done for two reasons, namely resilience and lowering client latency. In terms of resilience, having multiple diverse sites helps because if one site goes down, another site may take over. Also, multiple remote sites are less likely to be affected simultaneously by power outage or a natural disaster. Furthermore, typically an operator has data centers in different continents, thereby ascertaining that no matter where a client may be, there is a data center relatively close by.

2.1.1 Structure of a data center

Today's data centers are organized hierarchically [30], in the form of a tree. A single-rooted tree data center architecture[1] is shown in Figure 2.2. A typical data center hosts tens of thousands of servers [31]. The servers are installed in vertical racks. Apart from servers, the racks host other equipment as well. In addition to built-in hard drives in the servers,

[1]Multi-rooted and flat tree architectures have gained popularity, but since servers account for three times as much electricity consumption than the network [4], for the present discussion, we consider only the single-rooted tree topology.

Figure 2.1: Google data center locations - Source: http://bit.ly/YhZqAF

some dedicated storage nodes are also installed in the racks. A high speed Ethernet switch provides interconnection between the devices installed in the rack and connectivity to the rest of the data center and beyond. Power supply and distribution units for the equipment are also installed in the rack.

A group of racks, called a pod (or a cluster), are interconnected by means of aggregation switches. An aggregation switch allows servers in different racks to communicate with one another. All the pods within a data center are interconnected by core switches. This allows servers in different pods to communicate with each other. The core switches are interconnected through one or more border routers. These border routers are the gateways for traffic coming in to and going out of the data center.

All of the equipment is quite tightly packed in a data center. The equipment generates a lot of heat and to prevent thermal damage to it, cooling must be provided. This is generally done by air-cooling, i.e., heat is transported away from the equipment by circulating cool air around it.

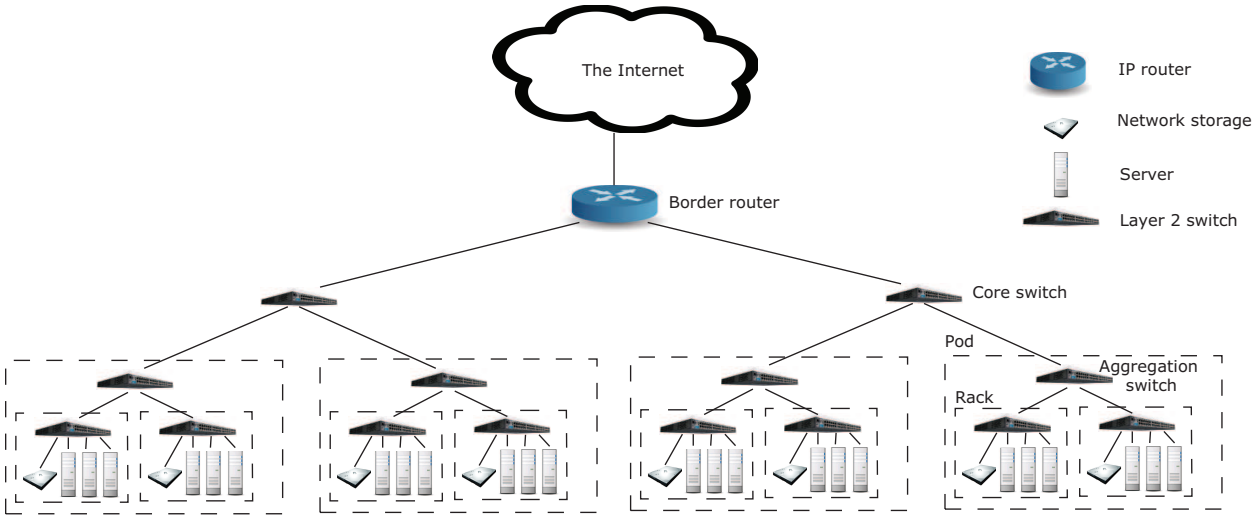


Figure 2.2: A single-rooted data center's architecture

2.1.2 Interconnection of data centers

In a geo-diverse data center setting, servers in a data center must often communicate with servers in other data centers. For instance, a web search engine operator may split the web index amongst the data centers. In this case, servers in different data centers may search for keywords independently and subsequently combine the results. The coordination between servers in different data centers would be done by means of network traffic on inter data center links. These links serve to carry various types of traffic, some of which are given below:

- **Consistency traffic:** An operator often maintains replicas of some of the servers in their data centers. These replicas are maintained for achieving one or both of the following three objectives: high throughput, resilience and low-latency to clients. Multiple replicas that handle client requests simultaneously can help to increase the effective throughput, i.e., the number of requests handled per second. Sometimes a secondary replica does not actively handle client requests until the primary server fails. When such an event occurs, the secondary replica takes over and offers continued

services to the clients. Lastly, by geographically distributing several replicas of a server and routing client requests to the nearest replica, the client latency to the server can be minimized.

If the content hosted on a server is static, all of its replicas are always automatically consistent, i.e., the user will have the same experience irrespective of the replica that serves its request. However, most applications have data that changes over time. For such applications, whenever content on one server changes, the change must be reflected to all other replicas. For instance, a customer's website may be hosted at two different data centers and whenever a change is made to one copy of the website, the same changes must be reflected at the replica as well. This action requires network traffic between the data centers that host the replicas.

- **Background traffic due to load-balancing:** Some traffic on the inter-data center links may be a result of the effort to achieve load-balancing amongst the data centers. As an example, consider a web-based email service provider that has partitioned user inboxes over the data centers. One objective of this partitioning may be that roughly the same amount of storage space is used at each data center. Since user inbox sizes are dynamic with different growth/shrink rates[2], the partitioning of users amongst data centers must be dynamic as well. The operator must periodically determine a new user partitioning that results in roughly equal storage space being consumed at each data center. In order to achieve this balanced storage size, some users' inboxes must be moved from one data center to another, over the inter-data center links.

- **Background traffic due to distributed computing:** Handling client requests in a distributed fashion requires network traffic overhead due to different servers commu-

[2]User inbox size grows when new emails are received and shrinks when emails are deleted. Furthermore, the inbox may grow and shrink at different rates, depending on factors such as number of mailing lists that a user subscribes

nicating user requests, intermediate results and final responses. For instance, a web search provider may distribute the web's index over multiple data centers. When a search query is being processed, the request must be sent to all data centers over the inter-data center links and the responses from the data centers, received over these links, must be aggregated.

2.1.3 Handling of client requests

We observed in chapter 1 that electricity cost depends not only on how much workload is handled, but also where it is handled. Thus, in order to develop a model for electricity cost in a geo-diverse data center, we need to first understand how workload from all over the globe is distributed amongst the data centers. In this section, we will use as an example a client request for viewing a web page hosted by a geo-diverse data center operator.

To access a web resource, the user types a uniform resource locator (URL) in the web browser's address bar. The URL typically contains the Fully Qualified Domain Name (FQDN), such as www.google.com, corresponding to the web server that hosts the requested resource[3]. Since a single server would hardly be sufficient to handle all traffic for a popular web site, several servers are actually mapped to the same FQDN. However, the web browser must connect to exactly one of these servers to fetch a particular web resource. Figure 2.3 briefly describes how this web server's IP address is picked.

The process to view the web page starts with what is known as DNS resolution. Details of DNS resolution are given in [32, 33], which we have summarized here. When the user directs the web browser to fetch a web page by typing a URL in the address bar, the browser invokes the local Domain Name System (DNS) resolver on the client machine which attempts to determine the IP address corresponding to the DNS name of the remote host specified in

[3]It is also possible to specify the IP address of the web server directly in the URL.

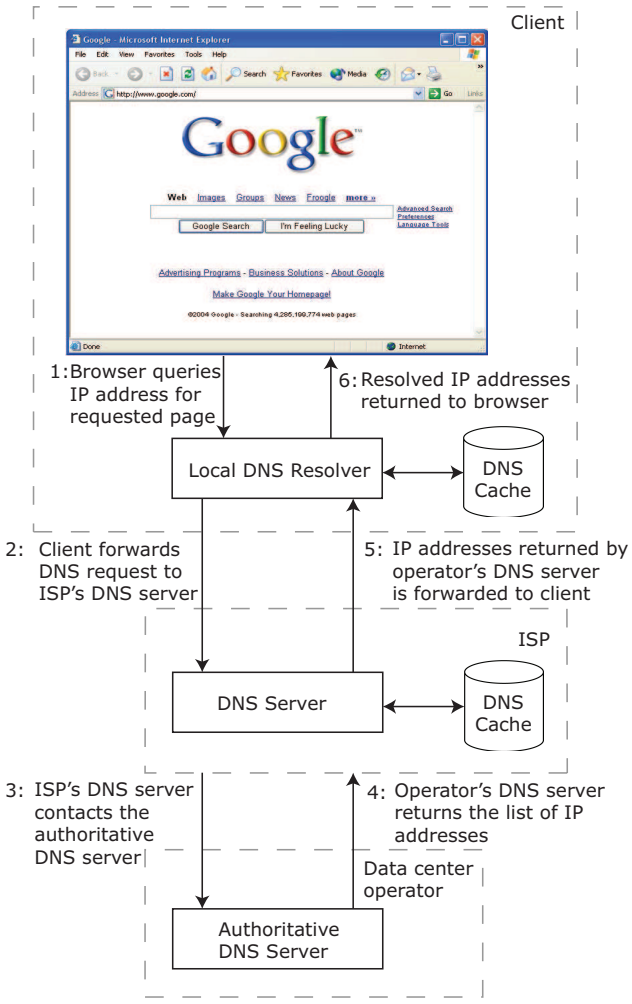


Figure 2.3: Resolving the IP address for a server hosted in a data center

the URL. The local DNS resolver communicates with the DNS server for the client's ISP[4]. The DNS query eventually reaches the authoritative server for the remote host's domain. In our example, this would be operated by the data center operator. The DNS server for the data center operator resolves the DNS name by returning a list of IP addresses corresponding to the servers that are associated with the DNS name specified by the client. When the client receives the DNS response containing a list of IP addresses, it picks one of the IP addresses to fetch the web page. For simplicity, many web browsers always pick the first IP address in the list. If the web browser is unable to connect to that IP address, it tries the second one on the list and so on. For consecutive responses to DNS queries for the same FQDN, DNS servers often rotate the list of IP addresses in each response so that if the client always picks the first IP address on the list, the servers are likely to receive equal workload.

Notice in Figure 2.3 that caches are available at various DNS resolvers in order to improve the latency of DNS resolution. These caches will keep the list of IP addresses corresponding to recently queried DNS names until the timeout specified by the authoritative DNS server occurs.

The data center operator has a large pool of IP addresses (IP address space) for their layer 3 devices. This IP address space is segmented over the geo-distributed data centers. The IP address picked by the client from amongst the list received from the operator's DNS server belongs to one of these data centers. The client must now send its Hyper Text Transfer Protocol (HTTP) [34] request to the server corresponding to the chosen IP address. To this end, the client establishes a Transport Control Protocol (TCP) connection with the server using the selected IP address. An HTTP request is sent over this TCP connection and a response is received. During this communication, packets from the client leave the client's network interface and are routed by the ISPs towards the data center where the required web

[4]Some people configure other DNS servers, such as Google's Open DNS Servers on their machines. In such cases, the local DNS resolver would communicate with those other DNS servers.

server is hosted. These packets enter the data center at the border router which forwards these packets to the server with the destination IP address. En-route, these packets traverse the core, aggregation and top of rack switches. The HTTP response packets are forwarded from the web server back to the border router which routes it back to the client machine. Hence, the requested web page is displayed in the client's web browser.

2.1.4 Data center power consumption model

The electric load within a data center may be categorized into (i) IT load and (ii) non-IT load. The IT load includes servers, network equipment and network storage. The non-IT load includes power supplies, un-interruptible power supplies (UPS), air-conditioning and lighting.

Fan et al. collected power usage characteristics of upto fifteen thousand servers hosted in a data center for a period of about six months [9]. These servers were running different classes of applications. They found a strong relationship between server power consumption and it's CPU utilization where the power consumption at no-load was non-zero. They also found that as CPU utilization increases, the corresponding increase in power consumption can be well approximated as linear. Thus, server power consumption can be modeled as an affine function of CPU utilization. As more and more client traffic arrives at servers in a data center, the average CPU utilization increases. Therefore, power consumption for elastic load in a data center can also be represented as an affine function of workload. The total power consumption for an operator is the sum of the power consumption over all data centers.

2.2 Cellular networks

Mobile telephone systems have enabled not only untethered access to traditional telephony services but also new types of services. We make phone calls, send text messages and can

even connect to the Internet using our mobile phones. Just as Internet connectivity services are provided by ISPs and Internet applications are powered by data center operators, mobile phone services are provided by mobile network operators (MNOs).

Over the years, mobile networks have been deployed based on different technologies. First generation cellular networks (1G) were based on Advanced Mobile Phone System (AMPS). AMPS networks were deployed starting in 1978. The AMPS system also evolved into Digital-AMPS (D-AMPS) networks. Two technologies were part of the second generation (2G) cellular networks, namely Global System for Mobile communication (GSM) and Code Division Multiple Access (CDMA). Anticipating the increased demand for mobile access to data services such as Internet access, vendors introduced General Packet Radio Service (GPRS) as an add-on to GSM networks. GPRS offers data rates between 56 kbps and 114 kbps. A GSM network with GPRS is sometimes referred to as 2.5G. GPRS bit rates are insufficient for many high bandwidth applications such as video calls, video streaming and video conferencing. To enable such services, broadband mobile services were introduced in third generation (3G) networks such as High Speed Downlink Packet Access (HSDPA) and Universal Mobile Telephone System (UMTS). The increasing trends in the use of high-bandwidth applications in mobile networks has spawned the fourth generation (4G) cellular networks such as Mobile WiMAX and Long Term Evolution (LTE). In this thesis, we consider only GSM cellular networks.

2.2.1 Structure of a cellular network

Mobile phone networks are also referred to as cellular networks because the area covered by the operator is logically divided into several small areas called cells. A cell in an urban setting is typically upto a few hundred meters in radius, whereas in suburban or rural settings, the cell radius may be as large as a few kilometers. A *cell site*, typically situated in the middle of a cell, enables subscribers in that cell to connect to the mobile network. A cell site, often

referred to as a Base Transceiver Station (BTS)[5] or simply a base station, hosts a number of transceivers (TRXs), radio antennas, power amplifiers and other allied equipment.

A cell is typically divided into three sectors, resembling 120 degree pie-slices, each covered by different directional antennae. Each sector may have multiple TRXs. A typical cell configuration is one with six TRXs in each cell, referred to as a 6+6+6 cell.

GSM uses a combination of frequency division multiple access (FDMA) and time division multiple access (TDMA). Each sector is assigned multiple frequency channels. Each frequency is also time-shared between users. Each GSM frame has a duration of 4.6 ms, which is divided into 8 time slots of 0.577 ms each. The transmission within a time slot for a given frequency in termed as a burst. The recurrence of a particular burst across all GSM frames is termed as a physical channel. In other words, there are 8 physical channels per frequency. A few of the physical channels in each sector are reserved for control signaling purposes and the rest can be used for user traffic such as voice calls. Since the number of physical channels represents traffic capacity, and the number of physical channels is proportional to the number of TRXs in a sector, the number of TRXs represents a cell's traffic capacity.

A government regulator such as Pakistan Telecommunication Authority (PTA) allocates a frequency band to each of the operators providing cellular service in the host country. The allocation is such that each operator gets a different frequency band. The operator distributes their allocated band to cells in their network. The channels allocated to an operator are much fewer than the number of cells in the network. Therefore, a given channel must be reused in an operator's network. Frequency reuse is done in such a way that any two cells that share the same frequency channel are sufficiently far apart so that the radio signal from any one of the cells does not noticeably interfere with that in the other. In fact,

[5]A single cell site sometimes hosts multiple BTSs, for instance, when multiple network operators share a single site

each cell is typically divided into three sectors (resembling 120 degree pie-slices), therefore, the frequency allocation is done on a per-sector basis. Nonetheless, for a high-level view, the set of frequencies allotted to all sectors in a cell can be considered as allotted to the cell itself. Each TRX at a cell site operates at a distinct frequency[6].

Frequency assignment and frequency hopping are examples of activities that require co-ordination. For systematic coordination of radio resources, Base Station Controllers (BSCs) are deployed in the network. The operator's coverage area is geographically split into zones, each of which is controlled by a single BSC. The BTSs communicate with the BSC via back hauls such as E-1 or microwave links. Since a BSC only controls a subset of BTSs, it can only perform local coordination. However, there is a need for global coordination in radio resource allocation. For instance, cells near the boundary of adjacent zones covered by two different BSC must not be assigned the same frequency channels to avoid interference. Therefore, in order to ensure global coordination in radio resource allocation and control, the BSCs are also interconnected using back haul links.

A mobile station (MS) often receives radio signals from multiple BTSs nearby. The MS picks the BTS from which it receives the strongest signal as its serving BTS. A MS will do all communication such as call reception and placement through the serving BTS. When a subscriber moves around in the operator's coverage area, the signal from the serving BTS might weaken. In such an event, the MS requests the network to allow it to change the serving BTS to the one from which it currently receives the strongest signal. This call hand-off is coordinated by a BSC.

[6]Given two communicating parties at fixed locations, if the transmitted signal power is kept constant, the received radio signal strength would differ depending on the frequency used. Also, this frequency selective behavior of the radio communication medium keeps changing with time, i.e., if frequency A receives better propagation compared to frequency B at time t_1, the same will not necessarily be true at time $t_1 + \epsilon$. This means that we can't statically pick the best frequencies to use for a particular cell by considering, for instance, the type of terrain. In order to make decent communication conditions available to all callers, on average, GSM networks also use frequency hopping, whereby the frequency allocation to cells are changed periodically.

Another key component of a GSM network is the Mobile Switching Center (MSC) that is responsible for call routing both within the GSM network and beyond (to a landline phone, for instance). Since the focus of our thesis is power consumption in the network and 50% [35] - 80% [36] of a cellular network's electricity consumption is due to the BTSs, we will not dwell on the MSC and other components of the cellular network any more than necessary.

2.2.2 Call placement

To place a voice call, when a user enters the phone number digits on the MS and presses the send button, the MS requests the BTS to acquire a channel to communicate with the MSC. Once this channel is successfully acquired, the MSC authenticates the MS, sets up encryption so that the call data is secure. The MS then sends the dialed digits to the MSC. The MSC instructs the BSC and MS to switch from signaling mode to voice mode and attempts to route the call to the called number. A downlink channel is assigned so that the ringing tone and called party voice can be heard on the calling MS. If there is an error in call establishment, an error message is heard on the calling MS over the same channel. In short, two channels are required to support a voice call in the sector where the call originates.

A sector with n TRXs has $8n$ physical channels. It appears that the sector should be able to support $4n$ calls because each call requires two physical channels. However, the actual maximum number of simultaneous calls is different from this number for two reasons:

- Some channels are reserved for control purposes, the exact number of such channels varies from operator to operator.

- GSM supports two different types of codecs, namely the full-rate codec and the half-rate codec. The full-rate codec corresponds to a caller using a burst in every GSM frame during a call, whereas the half-rate codec corresponds to a caller using a burst in every alternate GSM frame. By default, the full-rate codec is used for every call.

However, when traffic congestion rises above an operator-configured threshold, the network attempts to admit every new call using a half-rate codec, if the corresponding MS supports it. If the traffic rises further and crosses a second threshold as configured by the operator, the network also re-assigns current calls to use a half-rate codec depending on the corresponding MS support. This enables a BTS to support more than $8n$ simultaneous calls during times of congestion.

2.2.3 BTS power consumption model

BTSs account for most of the power consumed in a cellular network. Louhi [35] claimed that BTSs contribute 50% of overall network power consumption, whereas Oh et al. [36] put this number at 80%. For this reason, most of the prior work related to power consumption in cellular networks focuses on BTSs.

Lorincz et al. performed a measurement study of BTS power consumption under real-traffic conditions. They concluded that the BTS power consumption may be approximated as an affine function of call traffic [37]. Thus, as traffic varies during a given day, instantaneous power consumption would follow a similar curve as the traffic.

2.3 A comparison of data center and cellular networks

From our discussion of geo-diverse data centers and cellular networks, we note that these networks have several similarities and some subtle differences apropos their power consumption. Keeping their similarities and differences in mind, abstractions can be identified to generically represent these networks. These similarities and differences are summarized in Table 2.1. Both of these networks are a collection of resources that handle workload. The resources are data centers and BTSs in case of geo-diverse data centers and cellular networks, respectively. The workload is clients' computing requests such as web page views in case of

	Geo-diverse data centers	**GSM cellular networks**
Traffic Characteristics	Diurnal patterns	Diurnal patterns
Network capacity	Dimensioned according to peak workload	Dimensioned according to peak workload
Energy proportionality	No	No
Power consumption	Affine function of workload	Affine function of workload
Geo-diversity in **electricity prices**	Yes	No
Geo-flexibility in resource selected for workload handling	High	Low

Table 2.1: A comparison of data center and cellular networks

data centers, whereas it is voice calls in case of cellular networks. The workload in both types of networks exhibits diurnal patterns [11, 8]. The network in both cases is provisioned according to peak workload demand. Since the network resources are not energy proportional, this means that in low-workload regimes, the network is heavily over-provisioned. Thus, both of these networks are energy inefficient. The power consumption for both networks is an affine function of the workload.

At any given time, if x_i is the utilization of resource i, the power consumption for resource i may be given as:

$$P^i = P_{min} + x_i(P_{max} - P_{min}) \tag{2.1}$$

where P_{min} and P_{max} are the power consumption of a network resource under no-load and full-load respectively. Assuming that the network has a total of m identical resources, then the network's total instantaneous power consumption may be given as:

$$\sum_{i=1}^{m} \{P_{min} + x_i(P_{max} - P_{min})\} \tag{2.2}$$

Despite the similarity in the power consumption model for both networks, there are some differences as well. One such difference is that the workload in case of geo-diverse data centers may be handled at any of the data centers (assuming that all content is completely replicated on all data centers). However, a voice call may only be served through one of a few candidate BTSs that are near the caller. Since data center workload is geo-flexible and data centers are far apart (and thus under different electricity tariffs), the candidate resources for a given workload exhibit electricity price diversity. Meanwhile, nearby BTSs are unlikely to be under different tariffs.

2.4 Discussion

In this chapter, we have seen that there are certain similarities between geo-diverse data centers and cellular networks apropos their power consumption. We have used these similarities to identify abstractions for a general model of power consumption in these networks. Using these abstractions, we identified a generic power consumption model applicable to both geo-diverse data centers and cellular networks. In the next chapter, we will use this model to formulate a generic electricity cost minimization problem.

Chapter 3

A generalized model for electricity cost optimization

In Chapter 2, we observed that there are several similarities and a few subtle differences between geo-diverse data center and cellular networks in terms of their power consumption models. We proposed a generic view of geo-diverse data centers and cellular networks as a collection of resources that are used to handle client workload. We also saw that their power consumption may be modeled as an affine function of the workload. In this chapter, we will use this generic model to formulate a generic optimization problem that minimizes electricity cost for both these networks. Our generic power consumption model and the corresponding electricity cost optimization problem incorporates the subtle differences, as discussed in section 2.3, between the data centers and cellular networks.

3.1 Optimization problem model

In order to develop a generalized optimization problem for minimizing electricity cost, we use an illustrative example shown in Figure 3.1. The example uses a test tube to represent

a network site and marbles to represent a unit workload. The network site would be a data center in the context of geo-diverse data center operator, whereas it would be a BTS in the case of a cellular operator. Similarly, the workload unit would be a client request in the data center context, whereas it would be a call in a cellular network setting. The operator's goal is to assign workload to network resources and, if needed, periodically update this assignment in response to variations in workload.

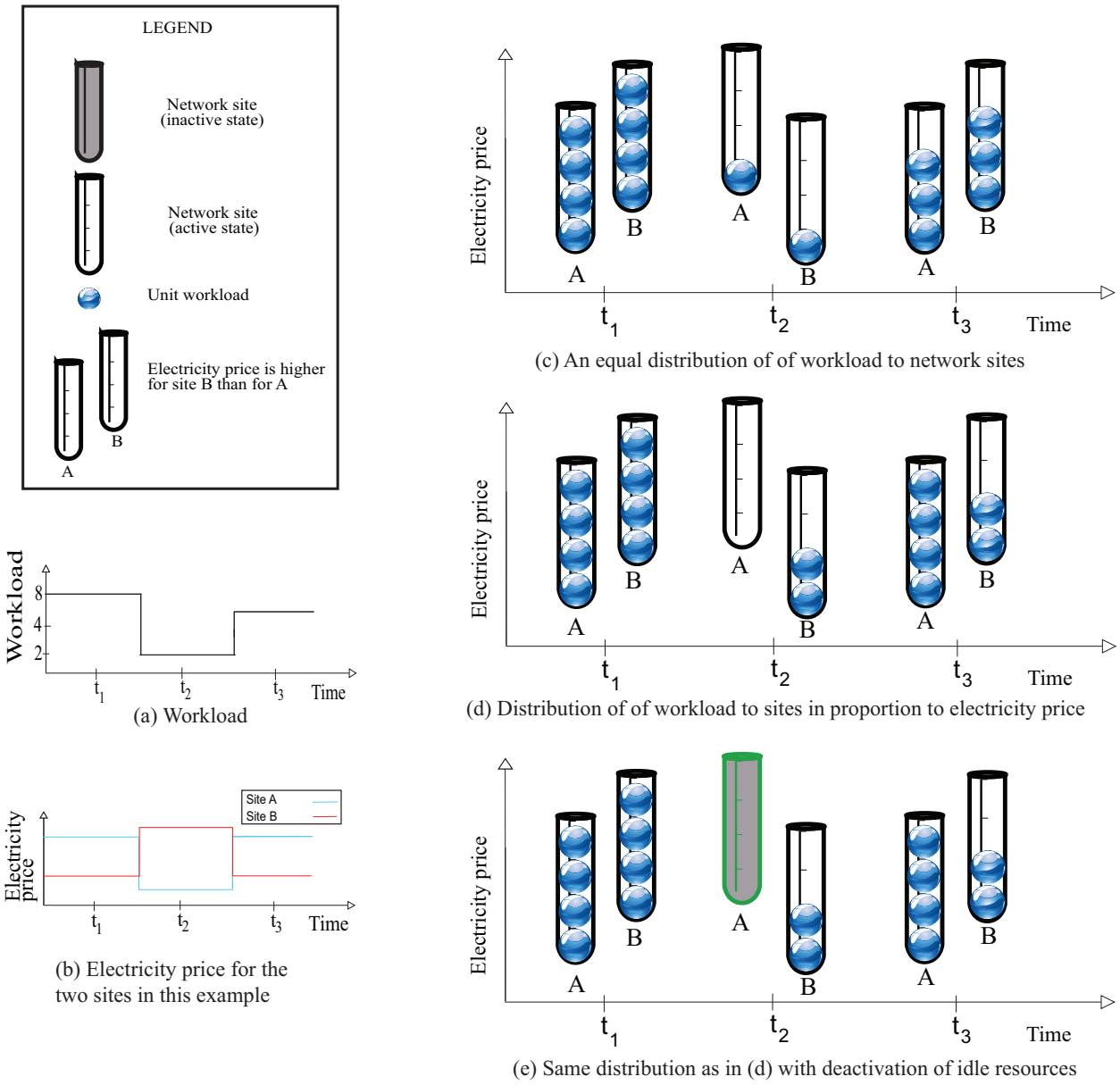
We consider the largest possible quantum of time for which the electricity prices remains fixed and term each such quantum as an *interval*. We assume that workload for several consecutive intervals is known and term this sequence of intervals as a planning window. The example in Figure 3.1 demonstrates three different ways (shown in parts c, d and e) of mapping this workload to two network sites situated at different locations, over a planning window consisting of three intervals centered at t_1, t_2 and t_3. In each interval, we assume that the workload is geographically split such that half of it originates near each of the two sites. For this example, we consider temporal variation in workload as shown in Figure 3.1 (a). Meanwhile, Figure 3.1 (b) shows the geo-temporal variation in electricity prices for the two network sites.

One possible operational strategy is to map each workload unit to the nearest available site as shown in Figure 3.1 (c). In a sense, this is the default strategy in cellular networks, whereby a call is handled by the BTS from which the mobile station (MS) receives the strongest radio signal[1]. In geo-diverse data center settings, this mapping strategy is also often the default strategy because it minimizes the access latency for all clients[2].

The above workload-site mapping strategy pays no attention to geo-diversity in electricity

[1]Signal from the physically nearest BTS may be weakened considerably due to natural or man-made obstructions. In such cases, the nearest BTS may not be the one from which the strongest signal is received. Hence, we take "nearest" to mean the BTS from which the MS receives the strongest signal

[2]Network latency has been shown to have a strong correlation with the physical shortest path distance between two locations on the globe [38]. So, the commonly understood physical measure of "shortest" applies in this case.



(a) Workload

(b) Electricity price for the two sites in this example

(c) An equal distribution of of workload to network sites

(d) Distribution of of workload to sites in proportion to electricity price

(e) Same distribution as in (d) with deactivation of idle resources

Figure 3.1: An example of mapping variable workload to capacity-limited network sites with geo-temporal diversity in electricity prices. Three consecutive intervals t_1, t_2 and t_3 are considered. Workload and electricity prices may only change between two consecutive intervals. (a) Workload considered in this example. (b) Electricity prices for the locations at which the two network sites are situated. (c) A uniform mapping of workload to network sites does not exploit electricity price diversity. (d) Mapping workload to network sites in order of their current electricity price. Due to lack of energy proportionality, only slight savings in electricity cost are possible. (e) Deactivating idle resources alongwith the resource mapping strategy of (d) may result in significant electricity cost savings.

prices. We can exploit geo-diversity in electricity prices to reduce the electricity cost over the planning window by mapping more workload to network resources at sites with cheaper electricity. To this end, we must change the way workload is mapped to resources as the electricity prices at various locations change. We term such changes in workload-site mapping as Workload Relocation (WR).

Figure 3.1 (d) shows a mapping strategy that uses WR to map as much workload as possible to resources at sites with cheaper electricity. In interval t_1, since the cumulative workload equals the total network capacity, both network sites will be operating at full capacity. Accordingly, there is no opportunity to reduce electricity costs using WR or RP. In interval t_2, on the other hand, as shown in Figure 3.1 (d), we may use WR to move all workload to network site B, which is situated at the location with the cheapest electricity price, thereby reducing electricity cost for that interval as compared to the default workload-mapping strategy. Similarly, in interval t_3 WR may be used to shift workload such that the cheaper site A is loaded to full capacity and the remaining workload is mapped to site B.

Due to lack of energy proportionality in networks, the power consumption of idle sites is a large fraction of their peak power consumption. Hence, consolidation of workload to cheaper locations offers a limited benefit in terms of reducing electricity cost compared to the default workload-mapping strategy. To avail considerable savings in electricity cost, one must use resource pruning (RP), i.e., deactivate idle resources. Notice that in the default workload-site mapping strategy of Figure 3.1 (c), there is no opportunity to deactivate idle resources in any of the three intervals. However, the purely-WR strategy of Figure 3.1 (d) may be augmented with RP, as shown in Figure 3.1 (e), to achieve maximal savings in electricity cost. The strategy in Figure 3.1 (e) not only shifts workload to the cheapest possible resources, but also deactivates as many resources as possible.

In claiming that Figure 3.1 (e) shows the maximal savings in electricity cost, we have assumed that activation and deactivation of network resources is free of cost. However, such

costs may exist in practice and in some networks it may even be significant compared to the total electricity cost of network operation. In such cases, care must be taken when defining the optimal strategy for network operation.

Unused fractions of a network site continue to consumer power on idle. Electricity cost savings from the WR and RP strategies may be improved if unused fractions of network resources could be deactivated. For instance, the power savings of the strategy shown in Figure 3.1 (e) can be improved by deactivating half of the resources at site B during intervals t_2 and t_3.

3.2 Optimization problem formulation

On a high-level, routine network operation involves distributing workload to network sites and periodically updating the fraction of workload mapped to each network site. For simplicity of modeling and analysis, we assume that the mapping of workload to network sites, henceforth referred to as *workload mapping* or simply *mapping*, is updated at the beginning of intervals of fixed duration.

We denote the status of site i during interval j by p_i^j, which may be a binary variable if the site may be either active or inactive. It may be possible to configure a site at a finer granularity, and thus, in general, p_i^j may have a value between 0 and l. For instance, if a data center consists of 10 pods, each of which may be independently (de)activated, p_i^j may have any integer value between 0 and 10. If p_i^j has a value equal to c, then exactly c pods are active while others are inactive.

We denote the workload being handled by site i during interval j as x_i^j. We refer to the status and amount of workload mapped to site i at time j collectively as the site's state, denoted s_i^j. The aggregated state of all network sites during interval j may be termed as the *network state* during the corresponding interval, denoted by S^j. The routine network

operation can thus be modeled as determining a sequence of network states for a time horizon, called a *planning window*, consisting of a set of consecutive intervals of equal duration. In the context of our thesis, the objective is to determine the state trajectory that is optimal in the sense that it minimizes the electricity cost over the planning window.

3.2.1 The objective function

The optimal state trajectory problem attempts to determine a sequence of states such that the sum of state costs and the cost of transitions between states in consecutive intervals is minimized. Mathematically, we may present the objective function as:

$$\sum_{j=1}^{n} C(S^j) + T(S^j, S^{j-1}) \tag{3.1}$$

Here $C(S^j)$ represents a function that evaluates the cost of being in state S^j. As per equation 2.2, $C(S^j)$ may be given as:

$$C(S^j) = \sum_{i=1}^{m} \left\{ P_{min} + x_i^j (P_{max} - P_{min}) \right\} \tag{3.2}$$

Furthermore, $T(S^j, S^{j-1})$ represents a function that computes the cost of transitioning from state S^{j-1} to state S^j during consecutive intervals $j-1$ and j. The transition costs may be a result of factors such as overhead electricity consumption when turning network resources on or off. We will show in the next chapter how activation and deactivation overheads may be computed using the values of the indicator variables p_i^j and p_i^{j-1}.

3.2.2 The constraints

The state trajectory problem must be subject to a number of problem-specific constraints. Some constraints are common to both geo-diverse data centers and cellular networks.

- **Site capacity must be respected:** During all intervals, we must ensure that the workload mapped to each network site does not exceed its capacity.

- **All workload must be handled:** During all intervals, the sum of workload mapped to all network sites must equal the offered workload for that interval.

- **Network site status is an integer value:** The status of a network site must be represented as an integer variable. This may be a 0/1 variable if the site may have only two states, on or off, for instance. If a resource may be configured in one of $l + 1$ power-saving states, then this variable may take on integer values between 0 and l.

  It might appear that the binary site status is redundant given that we are also keeping track of the workload mapped to the site as part of its state. A site may be considered off if it has zero workload and on otherwise. However, there is no linear function that can calculate the cost of resource activation and deactivation given the workload mapped to it in two consecutive intervals. One must introduce an auxiliary variable that represents resource status in order to keep the optimization formulation linear[3].

Several network-specific constraints must also be formulated. These constraints arise from subtle differences between different types of network. For instance, while any client request can be handled at any data center, a given call may be handled by only a few BTSs that are in the immediate vicinity of a caller. Such constraints will be specified in later chapters.

[3]Introduction of non-linearity in an optimization problem increases its computational complexity.

3.2.3 Comments on the problem formulation

The decision variables in our problem formulation are the state of network sites for each interval in the planning window. A site's state has two parts: the amount of workload it handles and its status (on or off). The site status needs to be a discrete (binary or integer) variable. The amount of workload may also be a whole number in some network types. For instance, in a cellular network context, the workload mapped to a site represents the number of active calls being handled by a BTS, which is a whole number. It is also possible to formulate the problem such that the workload mapped to a site during an interval is a fraction between 0 and 1, representing the fraction of total workload during that interval mapped to the particular site. In the first formulation, the site state is purely discrete, whereas in the second formulation, the site state is composed of a discrete as well as real-valued parts. In the former case, our optimization problem is an integer program (IP), whereas in the latter, the problem is a mixed integer program (MIP). Both IP and MIP are NP-Hard and must be solved using techniques such as branch and bound [39] or other heuristics.

If functions $C(.)$ and $T(.)$ as well as all constraints are linear and convex, the formulation is termed as an integer linear program or mixed integer linear program. Since the branch and bound technique repeatedly solves constrained and integer-relaxed versions of the IP (or MIP), linearity of the objective functions lowers the computational complexity. Fortunately, the nature of energy consumption in networks is such that the power consumption function $C(.)$ is linear and convex. For this reason, in our thesis, we strive to make the transition cost function $T(.)$ as well as all problem constraints as linear.

3.3 Summary

In this chapter, we have presented an abstract formulation of the general optimization problem for minimizing electricity cost in service provider networks. Two concrete instances of the optimization problem are discussed in the next two chapters. In order to solve both of those concrete instances, we used the CPLEX solver available with the ILOG CPLEX Studio. CPLEX solver uses the branch and bound heuristic. Our primary focus in this thesis is to investigate the potential that the use of RP and WR offers for electricity cost optimization. Therefore, we focus on solving both concrete instances of the optimization problem *exactly*. As we shall see in the next two chapters, we were able to solve problems of realistic size using simple desktop PCs within a reasonable amount of time. While our primary focus is not on proposing heuristics for approximate solutions to the problem, in the next two chapters, we do propose an approximation algorithm for the optimization problem for each of networks considered in this thesis.

Chapter 4

Case study I: geo-diverse data centers

4.1 Prelude

Geo-diverse data centers enable robust and low-latency cloud services. The electricity cost for this huge infrastructure is a significant fraction of the operational cost (15%) [4] as well as capital cost [18]. Due to increasing demand for cloud services and increasing electricity prices, it is essential for data center operators to cut their electricity bill [40, 41].

In the previous chapter, we presented an abstract and generalized electricity cost minimization problem that is applicable to geo-diverse data centers as well as cellular networks. In this chapter, we derive a specialized instance of the general problem, discuss its solution (exact as well as heuristic) and perform a sensitivity analysis. We will show that the RED-BL problem as applied to geo-diverse data centers is NP-Hard.

Due to geo-diversity in electricity prices and data center locations, workload relocation may be used to cut the electricity cost for geo-diverse data centers. However, the electricity cost savings resulting from workload relocation are somewhat limited due to lack of energy proportionality in today's data centers [11]. Therefore, it was proposed to dynamically scale the active infrastructure in response to temporal variations in workload [11, 24]. This

capacity scaling is expected to incur some overhead electricity cost which may be modeled as the *state transition cost* in the state trajectory problem.

Theoretically, one can benefit from capacity scaling by shutting down some equipment when it is not needed. However, some equipment in data centers may not be shut down because it is critical in nature. We term such equipment as inelastic load and the rest of the equipment as elastic load. Resource pruning of elastic load in data centers minimizes the electricity consumption, but conventional wisdom suggests that restarts affect equipment lifetime and operators are generally reluctant to adopt this approach. On the other extreme, elastic load may be left in an idle state, but the reduction in electricity consumption would be quite small. In between these two extremes would be Dynamic Voltage and Frequency Scaling (DVFS) techniques. The transition cost would be zero for the idling scheme since no state-change overhead is incurred. The transition costs would be quite high if elastic load is de-activated (and activated later when needed), whereas DVFS would account for transition costs somewhere between the two extremes [16]. In this thesis, we experiment with the entire spectrum of transition costs in order to generalize our results.

To the best of our knowledge, prior work has largely taken a micro-scale view of this problem by scaling the data center capacity at the granularity of states of individual servers within a data center [19, 20, 21, 22, 23, 24] and, in some cases, has altogether ignored transition costs resulting from capacity scaling. These approaches lack scalability to multi-data center scenarios or are sub-optimal. In this thesis, we address the challenges of scalability as well as incorporation of transition costs into the optimization problem.

In the present work, we approach a scalable solution to this problem by treating all the elastic electric load in a data center as an aggregate and determining this aggregate's state for each interval in a planning window. For every interval in the planning window, our coarse-granularity formulation provides the average utilization of the servers within a data center. Relaxing a discrete optimization problem to a continuous one typically introduces

an approximation error. The magnitude of this error is expected to be small for large scale problems such as geo-diverse data centers. Determination of the approximation error's magnitude is beyond the scope of this paper and is left as future work.

To motivate the significance of transition costs in the dynamic scaling of geo-diverse data center capacity and our scheme for incorporating the transition costs to the optimization problem, we will use a simple example shown in Figure 4.1. It depicts an example instance of the state trajectory problem for a planning window consisting of three intervals represented along the horizontal axis. For each interval, we show three sample states represented using rounded rectangles. Each state is labeled, in bold letters, with a name in the left half and the corresponding electricity cost in the right half. Moreover, transition between states in consecutive intervals is shown using arrows and the corresponding transition cost is shown as a label over the arrow. We consider three data centers in this example, represented using circles numbered 1, 2 and 3. In Figure 4.1, the relative height of the circles in a given interval represents the diversity in electricity prices. For instance, in interval 3, data center 3 has the lowest electricity price. In a particular state, the workload mapped to each data center is represented using shading within the circle. For simplicity of demonstration, we assume that the cumulative workload is fixed at a value that equals 1.3 times a single data center's workload capacity.

In the absence of transition costs, the optimal state trajectory could be obtained by making a *greedy* choice of state in each interval (the path S2→S6→S8 in Figure 4.1) [18, 24, 23]. This is clearly the lowest possible sum of state costs without considering any transition costs. With transition costs included, however, the greedy solution yields a total cost of 42. We refer to such a strategy as Relocate Energy Demand to Cheaper Locations (RED-CL).

One may also consider a *static* deployment configuration where an operator selects the data centers that have the lowest average electricity price over the planning window. This corresponds to the path S1→S4→S7, with the sum of state costs equal to 42. Since the

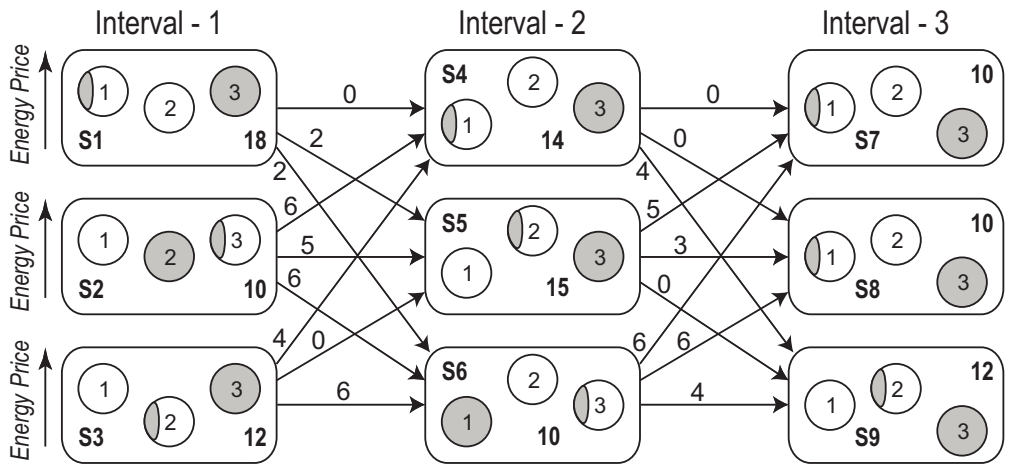


Figure 4.1: A motivating example that depicts the workload-mapping problem for three consecutive intervals involving three data centers of equal capacity. For this example, the workload is assumed to be constant equal to 1.3 times the capacity of a single data center, in all three intervals. Of many possible states in each interval, we show just three example candidate states along with the electricity cost for being in those states. Cost of transition from one state to another in the next interval are also labeled on the arrows representing the state transition.

workload mapping does not change, there are no transition costs, and hence the total solution cost is also 42. In general, depending on the magnitude of transition costs, the static solution could be better or worse compared to the greedy solution.

The optimal solution from Fig. 4.1 is the path S3→S5→S9, with a total cost of 39. For this state path, the sum of state costs is 39, which is higher than the corresponding component for the greedy solution. However, the sum of transition costs is 0 resulting in an overall lower total solution cost than both the static and the greedy strategy. This simple example illustrates that it is important to consider the costs associated with relocating demands in operational data centers.

4.2 Related work

Li et al. determined the electricity cost optimal mapping of workload to geo-diverse data centers by controlling the state of the individual servers within each data center [19]. The state of the servers and their electricity consumption was controlled using Dynamic Voltage and Frequency Scaling (DVFS) and Dynamic Cluster Server Configuration (DCSC). Since their optimization problem formulation used Mixed Integer Programming (MIP) with decision variables per server, their approach is effective for small-scale problems such as an individual data center or a fraction thereof. This limitation is evident from the small number of servers used in the simulation-based evaluation in [19]. They used the Soccer World Cup 1998 web server workload traces and electricity prices at four different locations to evaluate their proposal. One way to scale their approach to large distributed data centers is to use coarse granularity in their problem formulation. For instance, instead of controlling the state of each server independently, all servers in a single rack could be configured in the same state at a given time.

Some researchers have also proposed algorithms for the data center electricity cost optimization problem. In the context of a web-search query processing system hosted on a geo-diverse data center network, Kayaaslan et al. presented a bin-packing type of algorithm for shifting search query workload between data centers in [42]. Buchbinder et al. proposed online algorithms for relocating Map Reduce jobs between geo-diverse data centers to reduce the electricity bill while considering the cost of inter-data center bandwidth [43]. Their proposed algorithms consider the uncertainty in electricity prices and workload estimates while mapping the jobs to data centers. They evaluated their algorithms using electricity prices from 30 locations across the US and workload data from a 10000 node Map Reduce cluster. Bhaskar et al. proposed online algorithms for mixed packing and covering, a problem which may be applied to optimally map workload to geo-diverse data centers [44]. For configuring

servers in a single data center with a view of minimizing electricity costs, Lin et al. presented offline as well as online algorithms for dynamic scaling of server computational capacity [20]. Urgaonkar et al. proposed an online optimization algorithm while proposing to disconnect data center devices from mains and running on UPS when the electricity prices are high [45]. Their proposed scheme recharges the UPS units when the electricity prices are lower.

Investigations of infrastructure scaling strategies to conserve power in a single data center are reported in [21, 46, 47, 14]. Chen et al. proposed three different solutions that either shut down or frequency-scale servers in a web hosting data center with the objective of minimizing electricity and maintenance cost while ensuring SLA compliance [21]. The first two of the proposed algorithms were based on a queuing theoretic and control theoretic analysis, respectively, while the third one was a hybrid scheme. While scaling the deployed capacity, their proposed scheme considers the cost of turning the servers on and off in terms of the resulting wear and tear. Mazzucco et al. presented similar strategies in [46]. Oh et al. considered a virtualized environment and proposed solutions for optimally placing Virtual Machines (VMs) on servers and map workload to the VMs such that electricity costs are minimized [47]. In [14], Chase et al. presented policies for resource allocation in a hosting center alongwith a switching infrastructure for routing requests to servers.

Most of the prior work in this area considers applications with short request-response type jobs. In [15], however, Chen et al. considered connection-intensive applications such as video streaming, Internet gaming and instant messaging in the context of energy cost aware load dispatch.

Rao et al. consider data center operation in a futures electricity market and the possibility of hedging against uncertain electricity prices under a smart grid environment in [48]. The authors used workload data from Google search cluster and evaluated a scenario of an operator with data centers at two different locations.

Another related theme of research is greening of data centers. Some examples of work

that reports the results of efforts towards *green* mapping of workload to data centers are: [49, 50, 51, 52, 53, 54, 55, 56]. On a related note, Sucevic et al. studied various approaches for shutting down end-hosts to minimize the total electricity consumption on participant hosts in a peer-to-peer file download system [57].

All of the above work deals with problems that can be categorized broadly as optimal scheduling problems. Such problems arise in many different domains and prior work in such domains is relevant. For instance, System on Chip (SOC) [58], electric power systems and smart grid [59, 60, 61, 62], WiFi access points [63], wide area networks [64], cellular networks [8] and high performance computing [65, 66, 67, 68, 69, 70].

4.3 Instantiating the generalized optimization formulation

Consider a geo-distributed data center infrastructure comprising m interconnected data centers. At any given time, the workload is distributed amongst the data centers in the network. For ease of modeling, we assume that changes to the distribution of workload amongst data centers may be done at the start of discrete intervals of duration λ. We use x_i^j to denote the fraction of workload during interval j that is mapped to data center i. We consider workload that is normalized over its peak, i.e., the workload values for any interval are between 0 and 1. The workload capacity of data center i, denoted c_i, is also normalized on the same scale. We assume that the network of data centers is over-provisioned so that $\sum_{i=1}^{m} c_i > 1$.

Let the sum of elastic load's peak and idle power consumption over all data centers be P_{max} and P_{min}, respectively. Assuming that the data centers are homogeneous, an individual data center's workload capacity is directly related to its peak (or idle) power consumption. Thus, the maximum power consumption for data center i is $c_i P_{max}$ and the corresponding minimum is $c_i P_{min}$.

Data center power consumption is an affine function of the average CPU utilization of the servers [9]. Therefore, the average power consumption at data center i during interval j is:

$$P_i^j = c_i \left(P_{min} + \frac{x_i^j \left(P_{max} - P_{min} \right)}{c_i} \right) \tag{4.1}$$

Dividing both sides of the above equation by P_{max} gives the normalized power consumption for data center i during interval j:

$$\hat{P_i^j} = f c_i + x_i^j \left(1 - f \right) \tag{4.2}$$

where f is the ratio of P^{max} to P^{min}. If we set $x_i^j = 0$, i.e., data center i is not computing any workload during interval j, then the second term in equation 4.2 goes to zero and the power consumption reduces to the first term in equation 4.2 only, which we refer to as *idle power consumption*. The second term in equation 4.2 indicates the workload-dependent *computational power consumption*, which is independent of the data center capacity. Since electricity cost is the product of energy consumed and the unit price of electricity (e_i^j), the state cost $C(.)$ from equation 3.1 for interval j may be given as:

$$C(j) = \sum_{i=1}^{m} c_i e_i^j \left(p_i^j \lambda \left(f + (1 - f) \frac{x_i^j}{c_i} \right) \right) \tag{4.3}$$

Here, decision variable p_i^j is 1 if the elastic load in data center i is active during interval j, or 0 otherwise. Multiplication by p_i^j ensures that computation and idling costs are accounted for in interval j, only if the elastic load in data center i is active during that interval.

Let σ and δ be the average power consumption, over a single interval, required to activate

or deactivate, respectively, all of the elastic load at a unit capacity data center. Then, the power consumption for activation of the elastic load at data center i is σc_i and the corresponding electricity cost[1] is $c_i e_i^j \sigma$. Here, we are assuming that the elastic load at a data center may be activated within a single interval. The value of λ that we used in our experiments is equal to one hour, which is a sufficiently large interval for server activation. Deactivation cost for elastic load may also be derived in a similar manner. Let b_i^j be a binary variable which is 1 if data center i is activated in interval j and 0 otherwise. Similarly, let s_i^j be a binary variable indicating if data center i is deactivated in interval j, then the transition cost of entering a given state in interval j may be given as:

$$T(j) = \sum_{i=1}^{m} \left(b_i^j \sigma + s_i^j \delta \right) \tag{4.4}$$

Hence, the RED-BL optimization problem formulation may be given as:

$$\text{minimize} \sum_{j=1}^{n} \sum_{i=1}^{m} c_i e_i^j \left(p_i^j \lambda \left(f + (1-f) \frac{x_i^j}{c_i} \right) + b_i^j \sigma + s_i^j \delta \right) \tag{4.5}$$

subject to:

$$x_i^j \leq c_i \ \ \forall i, \forall j \tag{4.6}$$

$$\sum_{i=1}^{m} x_i^j = w^j \ \ \forall j \tag{4.7}$$

$$p_i^j, b_i^j, s_i^j \in \{0, 1\} \ \ \forall i, \forall j \tag{4.8}$$

[1]Multiplication with the duration of an interval, i.e., λ is not necessary, because σ is defined as the per interval cost.

Parameter	Description
m	Number of data centers
n	Number of intervals in a planning window
λ	Duration of an interval in hours
f	The ratio between a data center's peak and idle power consumption
c_i	Normalized workload capacity of data center i
σ	Penalty for activating the elastic load at a unit capacity data center as a fraction of its energy consumption at full load in one interval
δ	Penalty for deactivating the elastic load at a unit capacity data center as a fraction of its energy consumption at full load in one interval
e_i^j	Unit cost of electricity at data center i during interval j
w^j	Operator's workload during interval j
x_i^j	Workload mapped to data center i during interval j
p_i^j	1 if data center i is active (either computing workload or idling) during interval j, 0 otherwise
b_i^j	1 if data center i's elastic load is activated at interval j, 0 otherwise
s_i^j	1 if data center i's elastic load is deactivated at interval j, 0 otherwise

Table 4.1: Data center network model parameters

$$p_i^j \geq x_i^j \ \ \forall i, \forall j \tag{4.9}$$

$$b_i^j \geq p_i^j - p_i^{j-1} \ \ \forall i, 2 \leq j \leq n \tag{4.10}$$

$$s_i^j \geq p_i^{j-1} - p_i^j \ \ \forall i, 2 \leq j \leq n \tag{4.11}$$

$$b_i^0 = p_i^0, s_i^0 = 0 \ \ \forall i \tag{4.12}$$

The workload capacity constraint is given in (4.6). Eq. (4.7) ensures that all incident workload is handled, while (4.8) represents the binary-value constraint. Inequality (4.9) ensures that the elastic load in a data center is active whenever there is any workload mapped to it. The constraint in Eq. (4.10) ensures that b_i^j is 1 if the elastic load is inactive in interval $j-1$ and active in the next interval. The involvement of b_i^j in the minimization objective function ensures that it is 0 otherwise. Similarly, the constraint in Eq. (4.11) ensures that s_i^j takes on the correct value depending on the deactivation of elastic load in the data centers. We assume that the elastic load in all data centers is initially inactive,

therefore, an activation may be necessary at the first interval whereas deactivation in the first interval is not necessary. These conditions are ensured by the constraints in Eq. (4.12). It is easy to customize this constraint such that all data centers are assumed to be initially active.

4.4 Sources of transition costs in the data center scenario

This thesis uses the overhead of data center activation/deactivation as the transition costs. However, in practice, there can be other forms of transition costs as well, which would vary from one deployment to the other. Examples of other sources of transition costs include, but may not be limited to, the following:

- An operator might change the way user traffic is routed to data centers by modifying entries in the Domain Name System (DNS). In such cases, it has been reported that many client-side DNS caches violate the DNS entry Time-To-Live (TTL) by continuing to cache expired DNS entries [71]. This means that user traffic may continue to arrive at a data center that the operator does not intend to handle workload at. If this overwhelms the active data center capacity, it may result in lost revenue due to excessive response times.

- The operator might change the way user traffic is routed to data centers by making changes to the Border Gateway Protocol (BGP) routing table. Due to the complex nature of inter-service provider connectivity and BGP dynamics, BGP routing table changes have been shown to take an unpredictable amount of time to reflect globally [72]. This means that we run into a similar situation as for the DNS-based method described above.

- In order for any request to be handled anywhere, the data store for the applications must be replicated and consistency must be maintained. The cost of inter-data center traffic is quite high, hence this form of transition costs would be quite significant. The magnitude is not easy to predict because replication schemes are operator and application dependent. To the best of our knowledge, the current body of knowledge lacks a generic model for such traffic. Therefore, similar to [24], in this thesis, we assume that content is perfectly replicated.

It is clear from the above discussion that the factors contributing towards transition costs depend not only on how the data center network is deployed and operated but also on the applications being hosted. Since this information is confidential, the utility of modeling a specific deployment is limited. The question that is significant, however, is the impact on the possible electricity cost savings (resulting from geo-temporal diversity in electricity prices) of variation in the magnitude of transition costs relative to the electricity cost in a given interval. Therefore, we have used a normalized and parametrized model for transition costs in our problem formulation. Our aim in this thesis is to present electricity cost optimization solutions which operators can use, along with the parameter data (idling costs, transition costs, number of data centers and their locations) from their own data center network.

4.4.1 Problem complexity and a heuristic

The optimal workload relocation problem for geo-diverse data centers is merely the Unit-Commit problem [73] in distributed electricity generation and transmission scenario, which is known to be NP-Complete. As we will see later in this chapter, we were able to solve reasonably large sized instances of this problem using an NP-Hard MIP formulation using the CPLEX solver within reasonable time. Nonetheless, since the branch and bound heuristic typically used in solving MIP problems depends on data values in addition to the size of

data, it is possible that it may take longer to obtain a solution for a different dataset. Thus, we now present a heuristic algorithm for it.

The pseudo-code of our heuristic algorithm for RED-BL is given in Algorithm 1. Assume that the workload vector for the planning window starts at a trough, then rises in a non-decreasing manner to the peak before falling off in a non-increasing manner to another trough. Since the activation/deactivation costs are expected to be significant, our heuristic is designed to select a small number of data centers to operate in long continuous stretches during a given day. For the assumed characterization of the workload, elastic load at a few data centers would be sufficient to handle the workload early (and late) in the planning window. As the workload rises gradually, elastic load at some more data centers would need to be brought online. As the workload starts to fall, elastic load at some data centers may gradually be deactivated until the planning window ends. Our heuristic places two pointers at the beginning and end of the planning window, determines the number of data centers (d_1 and d_2) needed to handle the workload corresponding to the two pointers and picks the smaller of these two values. It then finds $\min(d_1, d_2)$ best data centers in terms of having the least average electricity price over the planning window. The elastic load at these data centers will be kept active between the intervals corresponding to the two pointers. Furthermore, our algorithm assigns as much workload as possible to the selected data centers in ascending order of average electricity price in the chosen intervals.

As long as the workload in the intervals corresponding to the two pointers may be handled by the same number of data centers, both of the pointers are moved closer to each other. Otherwise, the pointer that corresponds to the interval requiring the smaller number of data centers is moved towards the other pointer. This pointer movement is performed until either the pointers cross each other or the number of data centers required to handle the workload in the interval corresponding to the moving pointer increases. In the former case, we are done and in the latter, the algorithm repeats the data center selection and workload mapping step.

The algorithm then activates elastic load at data center(s) to meet the workload requirement of the pointer corresponding to the interval with the higher workload. The data center(s) where the elastic load is activated at this time are the ones that are not used before and have the least average electricity price in the interval between the two pointers.

Require: $w[1..n]$: Cumulative data center workload for the planning window,
$e[1..m][1..n]$: Electricity prices for all data centers over the planning window
Ensure: $z[1..m][1..n]$: workload assigned to the data centers for all intervals
$y[1..m][1..n]$: Data center status (1=on/0=off) over the planning window

1: $g_1 = 0; \quad g_2 = n - 1; \quad l = w; \quad a = 1..m; \quad n_c = 0;$
2: $\quad y[i][j] = 0; \quad z[i][j] = 0; \quad (\forall i, \quad \forall j)$
3: **repeat**
4: $\quad d_1 = \lceil w[g_1]/c_1 \rceil; \quad d_2 = \lceil w[g_2]/c_1 \rceil; \quad n_d = \min(d_1, d_2)$
5: $\quad$ **if** $n_d > n_c$ **then**
6: $\quad\quad$ Sort a in ascending order of average electricity price in $[g_1, g_2]$
7: $\quad\quad$ **for all** $i \in a$ **do**
8: $\quad\quad\quad$ **for all** $j \in [g_1, g_2]$ **do**
9: $\quad\quad\quad\quad y[i][j] = 1; \quad n_c{+}{+}$
10: $\quad\quad\quad\quad z[i][j] = \min(l[j], c_i)$
11: $\quad\quad\quad\quad l[j] = l[j] - z[i][j]$
12: $\quad\quad\quad\quad$ Remove i from a
13: $\quad\quad\quad$ **end for**
14: $\quad\quad$ **end for**
15: $\quad$ **end if**
16: $\quad$ **repeat**
17: $\quad\quad g_1{+}{+}$
18: $\quad$ **until** $(\lceil w[g_1]/c_1 \rceil > n_c)$**or**$(g_1 > g_2)$**or**$(\lceil w[g_1]/c_1 \rceil > \lceil w[g_2]/c_1 \rceil)$
19: $\quad$ **repeat**
20: $\quad\quad g_2{-}{-}$
21: $\quad$ **until** $(\lceil w[g_2]/c_1 \rceil > n_c)$**or**$(g_1 > g_2)$**or**$(\lceil w[g_1]/c_1 \rceil < \lceil w[g_2]/c_1 \rceil)$
22: **until** $g_1 > g_2$

Algorithm 1: Heuristic for the RED-BL problem

On line 1, the algorithm starts by placing two pointers, g_1 and g_2, at the extremes of the workload vector. Our algorithm would work best if the beginning and end of the workload vector coincides with the two troughs of the workload in the planning window. Also, on line 1, our algorithm makes a local copy l of the workload vector w, so that l may be used to keep track of the algorithm's progress without modifying the input vector. Line 1 also

includes the initialization of a list of available data center indices, a, and the initialization of the number of data centers currently in use, n_c, to zero. These steps would run in O(m+n).

The algorithm will store its solution in 2-D arrays y and z. Here, $y[i][j]$ is 1 if data center i is to be on during interval j and 0 otherwise. Furthermore, $z[i][j]$ holds the amount of workload to be mapped to data center i during interval j. This initialization takes O(mn).

The algorithm starts by computing the minimum number of data centers required to handle the workload during intervals g_1 and g_2 as the values d_1 and d_2, respectively. Since we have considered homogenous data centers, all c_i are equal, therefore, this computation uses c_1 as the capacity of a data center. The smaller of d_1 and d_2 is picked as n_d. If n_d, the minimum data center demand during $[g_1, g_2]$ is greater than the current number of active data centers n_c, the algorithm enters the if-block on line 6. On line 7, the algorithm first computes the average electricity price for the available data centers between intervals g_1 and g_2 in O(mn) and then sorts the available data center indices in ascending order of average electricity price from g_1 to g_2 in O(m lg m). Between lines 8 and 15, the algorithm marks $n_d - n_c$ data centers to be on from g_1 to g_2 (line 10), updates the current number of data centers that have been used by the algorithm, n_c (line 10), assigns workload to each of these data centers (line 11), updates the local copy of the workload vector, l, by subtracting the amount of workload that the algorithm has handled so far for the relevant intervals and removes the data centers that have been assigned workload from the list of available data center a (line 13). Since line 13 involves removal of some contiguous entries from the beginning of an array, it runs in O(m).

Lines 17-22 update the two pointers until they either demark a section of the workload vector that requires a greater number of data centers than n_c or g_1 exceeds g_2 (which means that we are done). If the workload during intervals g_1 and g_2 is such that they both require the same number of data centers, both of the repeat until loops run and pointers g_1 and g_2 are moved until they reach a point where the workload for each pointer requires a greater

number of data centers. Otherwise, only one of the repeat-until loops runs and the pointer corresponding to the interval requiring the smaller number of data centers is moved towards the other pointer until it reaches an interval where the workload requires a greater number of data centers.

The if-block (line 6-16) will dominate the overall execution time. Within this if-statement, on line 7, the average electricity price is computed and then the data center indices are sorted in ascending order. In the worst case, the if-block will be entered in every iteration of the outer repeat-until loop and exactly one data center index will be removed from a (line 13) in each iteration of the outer repeat-until loop. In this case, the computation of average electricity prices will require mn + (m-1)(n-1) + (m-2)(n-2) + (m-3)(n-3) + ... + (m-n+1) primitive operations. Sorting the electricity prices will require O(m lg m) + O((m-1) lg (m-1)) + ... + O((m-n+1) lg (m-n+1)) running time, overall. Within the nested for-loops, line 13 is most complex which will require, in the worst case, (m-1) + (m-2) + (m-3) + ... + (m-n-1) primitive operations. This is smaller compared to the running time of the average electricity price computation, so in Big-Oh notation, we can ignore it. The overall worst case running time for the electricity price averaging can be computed as follows. We first consider that the electricity price averaging is done exactly i times and will later replace i by its actual value of n.

$$
\begin{aligned}
& mn + (m-1)(n-1) + (m-2)(n-2) + ... + (m-i+1)(n-i+1) \\
& imn - n - 2n - ... - (i-1)n - m - 2m - ... - (i-1)m + 1 + 4 + ... + (i-1)^2 \\
& imn - (n+m)(1 + 2 + ... + (i-1)) + 1 + 4 + ... + (i-1)^2 \\
& imn - (n+m)\frac{i(i+1)}{2} + \frac{n(n-1)(2n-1)}{6}
\end{aligned}
$$

Substituting i by n, we get the overall worst case running time for the average electricity

price calculation step as O(mn^2 + n^3). The overall worst case running time of the entire algorithm (including the average electricity price sorting step) is O(mn^2 + n^3 + m lg m).

In the best case, only one data center is sufficient to handle all workload throughout the planning window and the outer repeat-until loop is only entered once. The average electricity cost is computed in O(mn), and the data center indices are sorted in O(m lg m). Removal of the sole selected data center's index from the array of available data center indices is done in O(m), whereas the increment of g_1 and decrement of g_2 is O(n). Hence, the best case running time complexity of the algorithm is O(mn + m lg m).

4.5 Experimental setup

In this section, we describe the experimental setup to perform a comparative study of different workload placement algorithms under various scenarios.

4.5.1 Application workload

We used an year-long trace of hourly workload for 3 social networking applications, with a subscription base of over 8 million users [74]. In order to make the dataset representative of a large data center network operator, we aggregated these traces into a week long trace as follows. We sliced the trace into week-long segments and considered each slice as workload for a different application, for the same week. We, then, normalized the sum of these trace vectors so that the peak cumulative workload corresponds to a value of 0.9. The normalized workload intensity is plotted in Figure 4.2. The statistical characteristics of our workload, as plotted in Figure 4.3 are quite similar to those reported by Google for "thousands of servers during a six-month interval at a Google data center" [11].

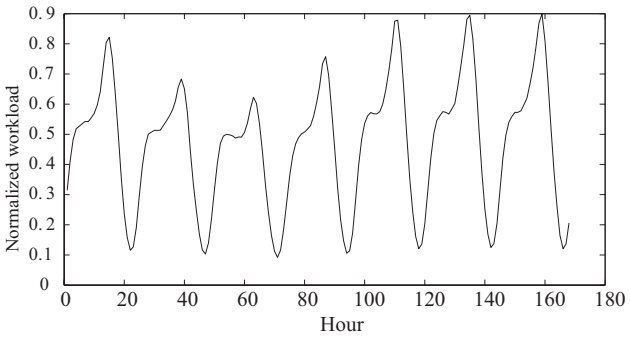


Figure 4.2: Normalized workload

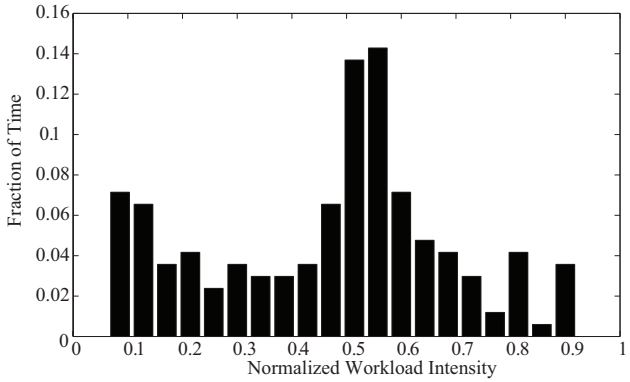


Figure 4.3: Workload intensity histogram

4.5.2 Electricity prices

We selected 33 different regions in the USA for which hourly electricity prices are available online. These regions belong to the following Independent System Operators (ISOs): NYISO, CAISO, MISO, ISO-NE and PJM. We used the day-ahead prices for these locations, i.e., the electricity price negotiated for the same hour on the following day. In all the experiments for this thesis, we considered an operator with data centers at all 33 locations in our dataset.

4.5.3 Algorithms for workload distribution/relocation

The workload relocation problem has the following dimensions based on which different algorithms may be formulated.

- For a given interval, the strategy for distribution of workload amongst data centers.
- For a given interval, the strategy for the state (on/off) of elastic load at a data center which has not been assigned any workload. In such cases, there is a trade-off between keeping the elastic load on (and incurring idling costs) and deactivating it (while incurring deactivation overhead and possibly activation overhead if it needs to be brought back online later in the planning window).

- Over the planning window, does the algorithm report transition costs in the total electricity cost?

In this thesis, we report comparative results for six workload placement algorithms.The following list describes and differentiates these algorithms. The same comparison is also presented in tabular form in Table 4.2.

- **RED-BL:** This is our proposed algorithm that determines the global optimal cost of electricity over a planning window while considering and reporting the transition costs. The choice of workload distribution as well as the state of elastic resources with no workload is governed by the optimal solution as determined by the CPLEX solver.
- **Heuristic:** This is the heuristic algorithm that we proposed in Section 4.4.1.
- **UNIFORM:** This algorithm represents the choice of those operators that find an even loading of their data centers desirable. This algorithm does not deactivate elastic loads and hence does not incur transition costs.
- **Greedy algorithms:** The originally proposed algorithm in [24] distributes workload to data centers such that, for each interval in the planning window, it makes a greedy assignment (in terms of current electricity price) of workload to data centers. Furthermore, this original algorithm keeps the elastic load at all data centers active in all intervals, incurring significant idling costs and hence is naturally disadvantaged against RED-BL. To have a fair comparison with the greedy workload distribution strategy, we use several variants of the original algorithm as well.
  - **Local optimal with Idling (LI):** This is the originally proposed algorithm from [24]. It does not deactivate elastic load.
  - **Local optimal withOut transition costs (LO):** This variant of LI was proposed in [24]. It deactivates un-needed elastic load while ignoring the transition

costs. This algorithm does not report transition costs in the total electricity cost of its proposed workload mapping for the planning window. This algorithm is very useful because it defines the lower bound on electricity cost that any algorithm can ever achieve.

- **Local optimal with Deactivation (LD):** This algorithm is similar to LO in all respects except that it also reports the activation/deactivation costs as part of the total cost of its proposed solution. Unlike LO, its results are practically relevant. Its total cost is less than (for all practical cases) LI, which makes it somewhat competitive to RED-BL.
- **Local optimal with Selection (LS):** In cases where transition costs are high compared to idling costs it would be better to keep the elastic resources at a data center active and incur idling costs if it will be needed again after the lapse of a small number of intervals. LS is a variant of LD that is empowered with the ability to *select* whether to deactivate unneeded elastic load at a data center or keep it idling. The cost of LS is never greater than that of LD.

4.6 Results

To evaluate the utility of workload relocation for electricity cost minimization, we formulated seven different scenarios. For each scenario, we ran seven experiments (one for each day's workload in our dataset) and report the average of the total electricity cost for each algorithm. Each experiment determines an operational plan for a planning window consisting of 24 consecutive intervals, each with a duration of one-hour.

Workload mapping strategy	
LI, LD, LS, LO	Greedy
RED-BL	Based on global optimal solution
UNIFORM	Workload equally divided amongst all data centers
State of a data center in an interval when it has no workload	
LI	Active and idling
LD, LO	Inactive
LS	Either inactive or idling, whichever is cheaper
RED-BL	Based on global optimal solution
UNIFORM	Active
Is transition cost reported in the total electricity cost reported?	
LI	N/A
LD, LS	Yes
LO	No
RED-BL	Yes
UNIFORM	N/A

Table 4.2: A comparison of the data center workload distribution algorithms studied in this thesis

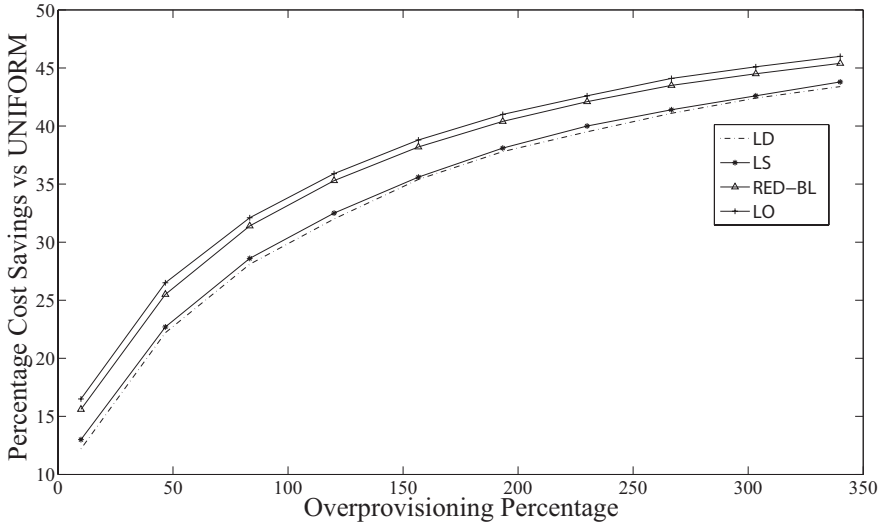


Figure 4.4: Percentage savings with over-provisioning

4.6.1 Sensitivity of electricity cost savings to extent of over provisioning

In this scenario, we investigate the relationship of the amount of data center capacity over-provisioning with the electricity cost savings. As we increase the amount of over-provisioning, each individual data center's capacity would increase enabling more and more workload to be mapped to data centers at locations with cheaper electricity price.

With data centers at all 33 locations in our dataset, we varied c_i between 0.03 and 0.12 (in increments of 0.01). This covers a variety of operators whose workload capacity ranges from just over expected peak workload to almost 300% over-provisioning.

We computed the total electricity cost for all algorithms while setting $f = \sigma = \delta = 0.65$. The percentage savings in total electricity cost by various algorithms compared to UNIFORM are plotted against the data center capacity over-provisioning in Figure 4.4. We found that for the wide range of capacity over-provisioning that we considered, LI is able to do only slightly better than the naive UNIFORM algorithm (about 2%). This is due to the significant idling costs incurred by LI under the experiment's conditions. For this reason, we have omitted LI from this plot.

The most competitive practical variants of LI, i.e., LS was 10.35% off from the ideal lower bound (LO). Meanwhile, RED-BL solution is quite close to the ideal lower bound (LO). The reason for greater savings with RED-BL compared to the greedy solutions (LS and LD) is that, the transition costs being significant, the former does fewer state transitions. In several intervals, RED-BL chooses data centers with relatively higher electricity price than the greedy solutions, but makes up for the higher computational cost by a reduction in the transition costs incurred.

4.6.2 Sensitivity of electricity cost savings to magnitude of transition costs

As the magnitude of transition costs relative to the state cost for an interval grows beyond a certain point, the benefits of deactivating elastic load at data centers would diminish. Accordingly, the electricity cost savings achievable by the workload relocation schemes would drop with increase in transition costs. In this scenario, we determine the percentage savings in total electricity cost for each algorithm, except STATIC_MIN[2], compared to UNIFORM, while varying the activation/deactivation overhead between 0 and 1, in increments of 0.1. The lower bound on σ (and δ) implies the ideal condition of no transition overheads. We set the upper bound to 1 so that the transition costs equal the cost of operating a data center at full load for an interval. A transition cost higher than this does not make sense as a workload relocation scheme would be better off keeping the elastic load at unloaded data centers idle. In this scenario, we kept $f = 0.65$.

LI does not (de)activate unneeded elastic load and thus its electricity cost is independent of the magnitude of transition costs. We observed that it offered a saving of merely 1.74% compared to UNIFORM. Figure 4.5 shows the electricity cost savings for the other algorithms compared to UNIFORM. The LS and LD adaptations of the LI algorithm offer savings that scale almost linearly to the magnitude of transition costs. Both LS and LD also bring an average reduction in the elastic load's electricity cost by a factor of 4, compared to that of LI. RED-BL not only scales better than LS and LD but also achieves electricity cost saving that is fairly close (only 3% higher, on average) to the ideal lower bound as reported by LO.

[2]STATIC_MIN uses a static workload mapping so it is insensitive to variations in transition costs

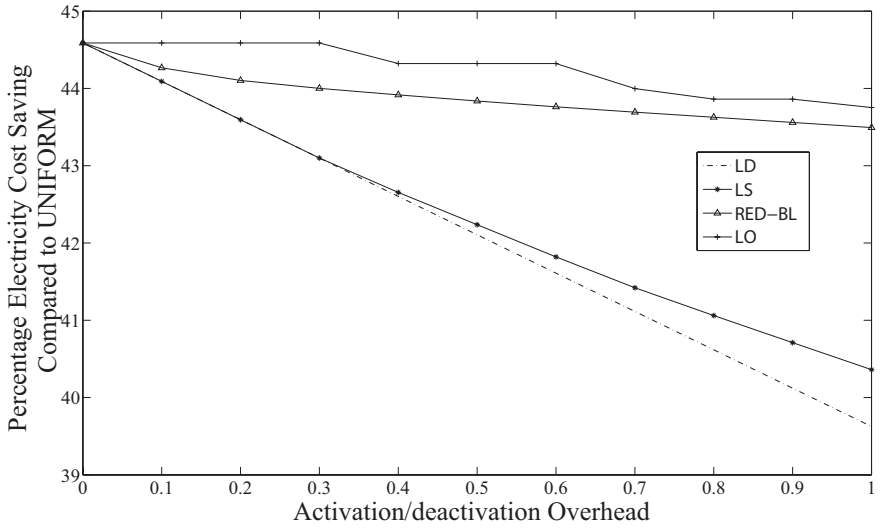


Figure 4.5: Total cost vs transition overhead

4.6.3 Sensitivity of electricity cost savings to resource pruning granularity

In this scenario, we investigate the potential benefits of deactivating the elastic load in a data centers in equal sized chunks instead of an all or nothing approach. The size of the portion of the elastic load in a data center that may be independently (de)activated may be deployment-dependent or operator-dependent. Possible choices of granularity may be a rack, a pod or one half of the elastic load etc. This granular deactivation can also be seen as throttling the equipment in a data center using facilities like processor frequency scaling.

Granular (de)activation is expected to bring additional power savings. For instance, if the elastic load in a data center is operating at 10% of its capacity, then the remaining 90% of the load is still consuming significant idling power. If we had the ability to power down half of the elastic load at a data center, we could cut idling energy cost significantly.

The optimization problem formulation with l granular (de)activation levels is given by:

$$\text{minimize} \sum_{j=1}^{n}\sum_{i=1}^{m} c_i e_i^j \left(p_i^j \lambda \left(\frac{f}{l} + (1-f)\frac{x_i^j}{c_i} \right) + \frac{b_i^j \sigma}{l} + \frac{s_i^j \delta}{l} \right)$$

subject to:

$$x_i^j \leq c_i \ \ \forall i, \forall j \tag{4.13}$$

$$\sum_{i=1}^{m} x_i^j = w^j \ \ \forall j \tag{4.14}$$

$$p_i^j, b_i^j, s_i^j \in \{0, 1, ..., l\} \ \ \forall i, \forall j \tag{4.15}$$

$$p_i^j \geq x_i^j * l/c_i \ \ \forall i, \forall j \tag{4.16}$$

$$b_i^j \geq p_i^j - p_i^{j-1} \ \ \forall i, 2 \leq j \leq n \tag{4.17}$$

$$s_i^j \geq p_i^{j-1} - p_i^j \ \ \forall i, 2 \leq j \leq n \tag{4.18}$$

$$b_i^0 = p_i^0, s_i^0 = 0 \ \ \forall i \tag{4.19}$$

There are three primary differences from the vanilla RED-BL formulation. The first difference is in the objective function, where the idling, bootup and shutdown costs depend on the number of granular units involved in the idling, bootup or shutdown process respectively. Since the computational cost component of power consumption only depends on the workload and is independent of the data center capacity, it is independent of the number of granular units being used at a data center during a given interval. The second difference is in the domain of p_i^j, b_i^j and s_i^j (see constraint 4.15). The third difference is in the constraint 4.16, which ensures that p_i^j takes on an appropriate value from $0, 1, ..., l$.

In the current evaluation scenario, we explore how the RED-BL electricity cost savings scale with change in size of the unit of independent (de)activation. In Figure 4.6, we have

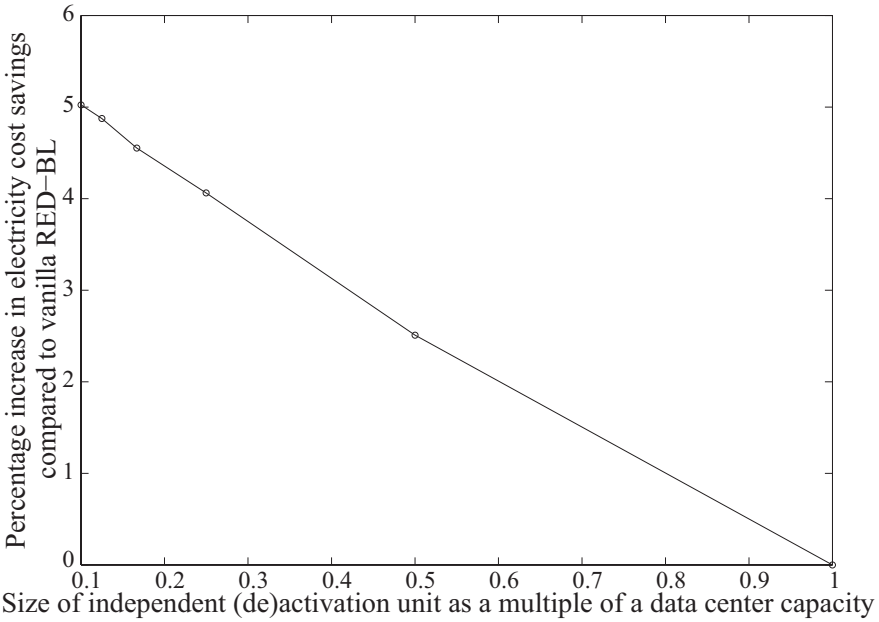


Figure 4.6: Cost saving vs (de)activation granularity

plotted the percentage savings in electricity cost vs the granularity of data center's elastic load (de)activation. The savings are computed against the scenario where only the entire elastic load in the data center may be (de)activated as a whole. This baseline corresponds to a value of l equal to one. Accordingly, in Figure 4.6 we see no savings for that value of l. We also see that the ability to independently (de)activate half of a data center's elastic load provides around 2.5% additional such savings on top of what the vanilla RED-BL can achieve. The electricity cost savings grow almost linearly when going to more granular size of independent (de)activation.

4.6.4 Sliding window re-optimization

So far, all of our simulation scenarios have been driven by error-free workload traces. The underpinning assumption to the corresponding results, therefore, is the availability of accurate workload estimates. We opine that this is not such a bad assumption given that the cumulative workload on the granularity of an hour changes slowly from one hour to the next

and from one hour on a day to the same hour the next day. However, workload forecasting will have some error, however small it may be.

In order to accommodate workload mis-estimation, we propose a sliding window-based algorithm that is somewhat similar to receding horizon control [75]. The algorithm is so called because it performs workload forecasts for a planning window and later slides the window by a constant offset before making another workload forecast, this time for intervals currently in the planning window.

Our approach compensates for workload estimation errors on two (often) different timescales. At an (often) longer timescale, the workload for the next n intervals is forecast and a RED-BL plan is generated. This forecasting is repeated every γ intervals, the *window slide interval* parameter. The motivation is that at interval number 1, the workload forecast for interval number $gamma + k$ is expected to contain a greater amount of error than if the forecast for the same interval is done at interval γ, due to availability of more historical workload data at the later interval. This step of repeated forecasting and subsequent generation of a RED-BL plan is called *global trajectory correction*. The basic idea behind this approach is that lowering of workload estimation error should bring the planned state trajectory closer to the optimal state trajectory (the one that results from perfect workload estimates).

On a relatively shorter timescale, in each interval, our algorithm locally corrects for forecasting errors. The planned state for an interval may be *infeasible* in the sense that it may be based on an under-estimation of workload and we might not have sufficient active data center capacity for the actual workload. Also, in case of over-estimation of workload, the originally planned state may be *locally sub-optimal* as some data center resources would unnecessarily consume idling costs. This step that corrects for locally infeasible or sub-optimal states, considers only a single interval and, hence, is called *local trajectory correction*. Upon entering an infeasible or locally sub-optimal planned state, we perform a local correction by finding a new state for the current interval. As shown in Figure 4.9, we start at the initial

state S_0 and are scheduled to transition to state $\hat{S}_1$ at interval 1. However, at interval 1, we know the actual workload received and might discover that the planned state is locally infeasible or sub-optimal. To accommodate this, we transition to a locally better state S_1. At the end of interval 1, we are scheduled to transition to state $\hat{S}_2$, and the process repeats. Since we only have accurate information about the workload for the present interval, the revision of the next state is always deferred to the next local trajectory correction step.

The local trajectory correction for interval j is an optimization problem that attempts to minimize the electricity cost of the corrected state S_j and the cost of transition between the planned state $\hat{S}_j$ and the corrected state. The mixed integer linear programming formulation for the local trajectory correction step for interval j is as follows:

$$\text{minimize } \sum_{i=1}^{m} c_i e_i^j \left(p_i^j \lambda \left(f + (1-f)\frac{x_i^j}{c_i} \right) + \left(b_i^j + \hat{b}_i^j \right) \sigma + \left(s_i^j + \hat{s}_i^j \right) \delta \right) \tag{4.20}$$

subject to:

$$x_i^j \leq c_i \ \forall i \tag{4.21}$$

$$\sum_{i=1}^{m} x_i^j = w^j \tag{4.22}$$

$$p_i^j, b_i^j, s_i^j \in \{0,1\} \ \forall i \tag{4.23}$$

$$p_i^j \geq x_i^j \ \forall i \tag{4.24}$$

$$b_i^j \geq p_i^j - \hat{p}_i^j \ \forall i \tag{4.25}$$

$$s_i^j \geq \hat{p}_i^j - p_i^j \ \forall i \tag{4.26}$$

$$\hat{b}_i^j \geq \hat{p}_i^j - p_i^{j-1} \ \forall i \tag{4.27}$$

$$\hat{s}_i^j \geq p_i^{j-1} - \hat{p}_i^j \ \forall i \tag{4.28}$$

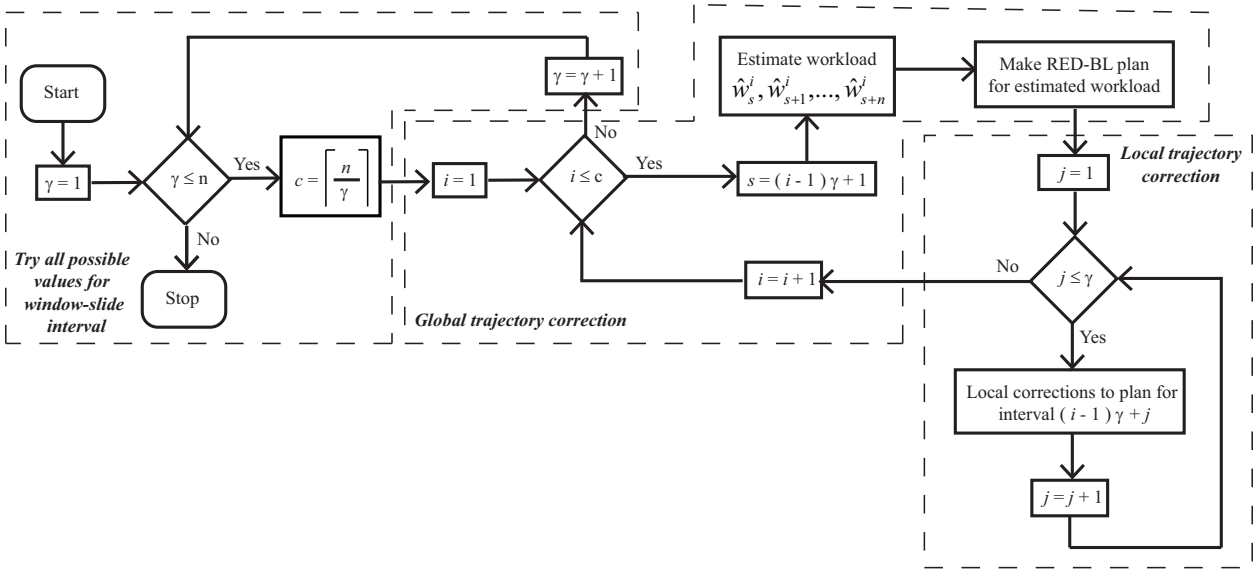


Figure 4.7: Flow of sliding window experiments

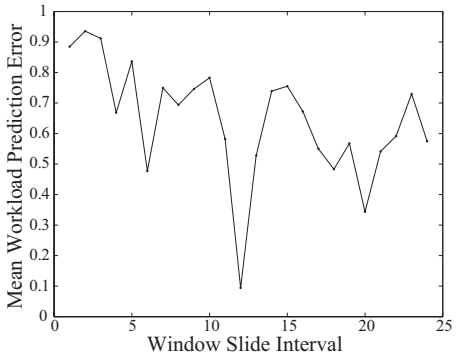


Figure 4.8: Mean absolute workload prediction error vs sliding window size

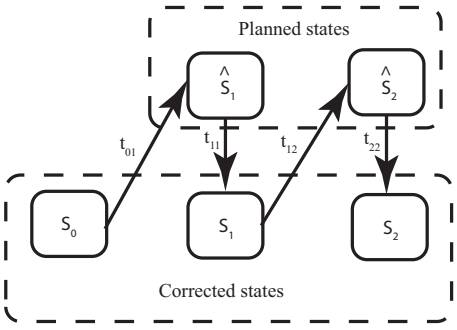


Figure 4.9: Local trajectory correction technique for three consecutive intervals

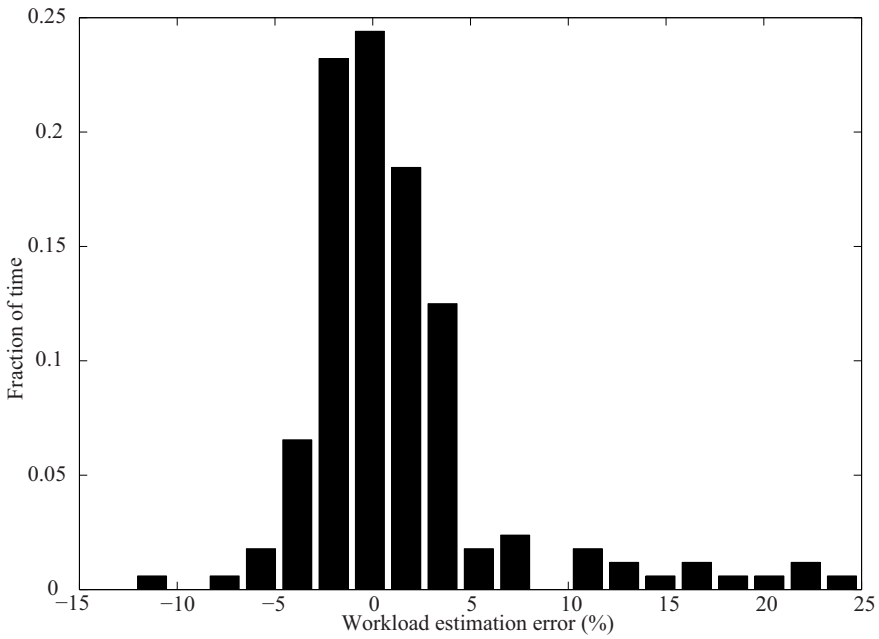


Figure 4.10: Distribution of workload prediction error for sliding window size of 12 hours

Given that the planning window size is n intervals, the possible values for γ are $1, 2, .., \gamma$. We experimented with all possible values for γ. Figure 4.7 shows the flow of our experiments. We first pick a value for the window sliding interval and estimate the workload for the next n-intervals and then invoke RED-BL. As an example, consider $\gamma = 2$. We start by forecasting the workload for the first n-intervals, denoted by $\hat{W}_1^2 = [\hat{w}_{1,1}^2, \hat{w}_{2,1}^2, ..., \hat{w}_{n,1}^2]$. Here, $\hat{w}_{j,1}^2$, for instance, represents the workload forecast for interval j during the first forecasting operation while the value of γ is 2. For workload forecasting, we trained an ARMA(4, 4) [76] model on a day's workload. Using $\hat{W}_1^2$ as the expected workload vector, we propose a RED-BL deployment plan for the first n-intervals. After the lapse of γ intervals, i.e., at the start of the third interval (for $\gamma = 2$), we forecast the workload for the next n intervals, leveraging the additional information about the actual workload for the first two intervals which was not available in the first forecast step at $t = 0$. This forecast is denoted by $\hat{W}_2^2 = [\hat{w}_{3,2}^2, \hat{w}_{4,2}^2, ..., \hat{w}_{n+2,2}^2]$. Then, we compute the RED-BL deployment plan for intervals $3, 4, ..., n + 2$ as the global trajectory correction step. Since the window sliding interval size

is γ and the number of intervals in our experiments is n, the number of times the window must slide, for a given value of γ is $\lceil n/\gamma \rceil$. For $\gamma = 1$, our scheme reduces to something resembling receding horizon control.

Having trained the model on the first day's data, we ran experiments for the last six days' workload in our dataset. We computed the average error of the daily electricity cost reported by these experiments compared to the total daily electricity cost for the same period with perfect workload estimates. The size of the planning window was set to 24 hours.

The first set of results in this scenario is the percentage workload estimation error for various sliding window sizes. We see in Figure 4.8 that the mean absolute percentage prediction error is less than 1%. The minimum mean error is for a sliding window size of 12 hours. For this sliding window size, the distribution of percentage workload estimation error is plotted in Figure 4.10. Most of the workload estimates are quite close to error-free, while a few estimates are as much as 24% off. This low average error for $\gamma = 12$ is expected due the nature of daily variations in the cumulative workload.

The difference of the electricity cost resulting from the use of the sliding window trajectory correction approach compared to the optimal solution with perfect workload knowledge is plotted in Figure 4.11. We see that the electricity cost achievable with RED-BL in a sliding window fashion is within 5-7% of the optimal cost achievable with perfect workload estimates.

4.6.5 Sensitivity of electricity cost savings to the server idle-peak power ratio

Chip manufacturers and computer vendors are striving to make servers more energy proportional/efficient. It would be interesting to see how the electricity cost savings for RED-BL would improve as the server's idle to peak power ratio (f) drops. To this end, we conducted a set of experiments in which we kept all other parameters fixed while varying f from 0 to

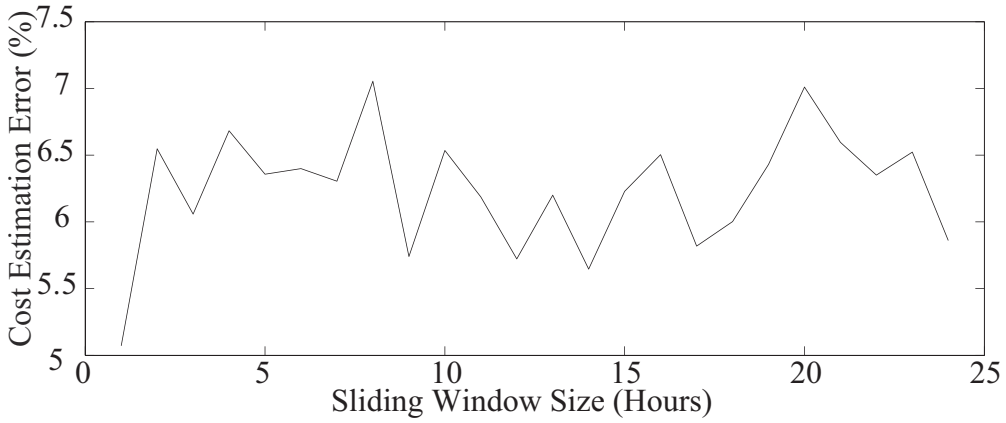


Figure 4.11: Percentage error of sliding window forecasts compared to global optimal with error-free workload

1. Note that f=0 means that the servers are completely energy proportional, whereas f=1 means that the server power consumption is totally inelastic.

Figure 4.12 shows the variation of electricity cost savings as the value of f is varied while the b/s parameters are kept at a low value of 0.01. In this setting, we wish to investigate if server energy proportionality improvement really matters if the bootup/shutdown costs were really low. Figure 4.12 shows that for an increase from f=0 to f=1, the global optimal solution over the planning window only spans an increase in average total electricity cost of 8.34%. Also, a drop from the typical f=0.6 all the way to f=0 results in a 4.88% drop in total electricity cost. It is noticeable that the bootup/shutdown overheads are really small and the idling fraction of electricity costs increases almost linearly with the value of f.

Figure 4.13 shows the same results for b/s=0.65. In this case, an increase from f=0 to f=1 spans an 11.44% increase in average daily total electricity costs. Also, as f drops from the typical f=0.65 all the way to f=0, the total electricity cost of the global optimal solution over the planning window drops by 7.42%. Since the bootup/shutdown overhead is significant in this case, for all values of f, the total electricity cost of the global optimal solution over the planning window is characterized by an almost fixed contribution from bootup/shutdown, while the idling fraction increases almost linearly.

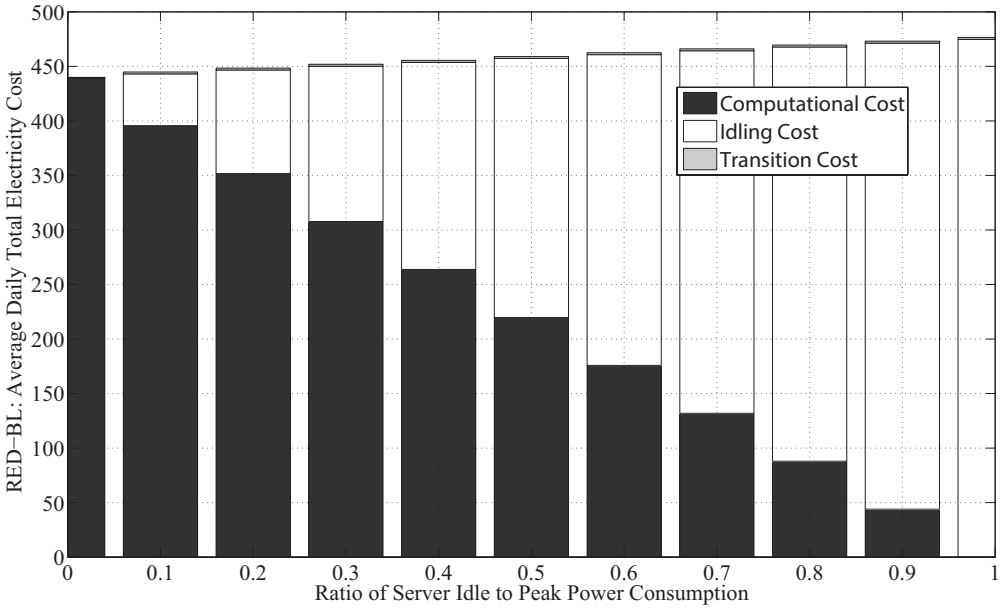


Figure 4.12: Average daily total electricity cost and its components vs f, For b/s = 0.01

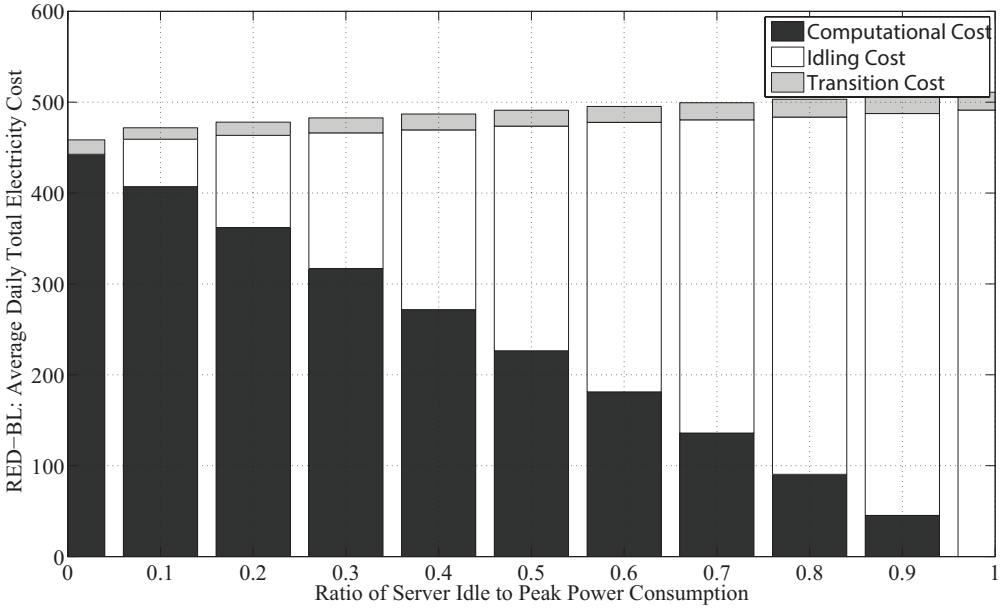


Figure 4.13: Average daily total electricity cost and its components vs f, For b/s = 0.65

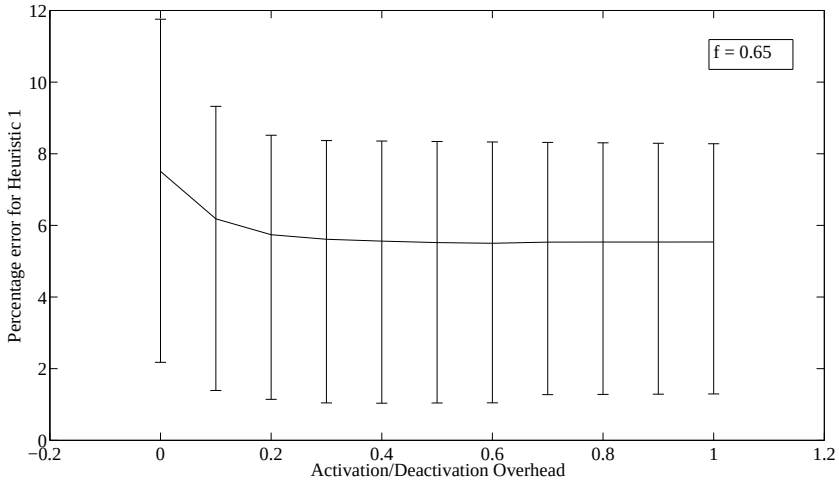


Figure 4.14: The minimum, maximum and average percentage difference between the cost of our heuristic and RED-BL

4.6.6 Performance of the heuristic algorithm

Figure 4.14 shows the performance of our heuristic algorithm compared to the optimal solution of the problem for various values of the (de)activation overhead parameters. For each value of the b/s parameters, we have plotted the average error over the seven days in our workload dataset (the curve) as well as the minimum and maximum error for any given day (the vertical bars). The performance of the heuristic is the worst for $b/s = 0$, because the heuristic avoids bootup/shutdown which has zero cost for $b/s = 0$. For other small values of b/s also, the bootup/shutdown overhead is not significant and by avoiding it, our heuristic fares relatively poorly compared to the optimal solution. As the value of b/s increases, our heuristic's error compared to the optimal solution drops until it starts a slight rise. The rising trend in the heuristic's performance for relatively high values of b/s is because in this regime, it may often be better to allow idling of some data centers instead of a bootup at the intervals defined by p_1 and shutdown at those defined by p_2. We observed similar trends for other values of f as well, when b/s is varied from 0 to 1.

4.7 Summary

In this chapter, we have instantiated the generalized problem formulation of joint workload relocation and resource pruning to optimize electricity costs for geo-diverse data centers. The generalized formulation performs a state trajectory optimization and in case of geo-diverse data centers, the network state consists of a combination of discrete as well as continuous variables.

Our formulation determines a deployment configuration plan that is optimal over a planning window consisting of several intervals. This deployment configuration consists of bootup/shutdown times for each data center as well as the workload to be mapped to each data center for all intervals in the planning window. Discrete optimization problems such as ours are difficult to solve exactly because of their computational complexity. However, using the CPLEX solver we were able to compute the optimal plans for planning windows consisting of 24 one-hour intervals within a few seconds.

Our problem formulation involves several parameters whose values are expected to vary over time (for instance, the value of idle to peak power consumption ratio is expected to improve as chip and server energy proportionality improves), from one operator to another (for instance, the over provisioning ratio) and from application to application (for instance, the value of the transition cost). With so many parameters, each with many possible values, it is not possible to exhaustively test RED-BL. However, in such a case, it suffices to know the sensitivity of electricity cost savings to the variation in each parameter, while all others are held at some reasonable default values. We performed a series of such experiments for each parameter and reported the results.

Since RED-BL devises a deployment plan for a planning window based on workload forecasts, which may be somewhat erroneous, in a practical setting, RED-BL plans must be revised periodically to bring the system closer to an optimal state. For this purpose, we

proposed a sliding-window re-optimization procedure and evaluated its performance.

We used workload datasets from three live Internet applications. While the number of applications is quite small for a large geo-diverse data center operator, the statistical characteristics of the dataset are reasonably representative of such a scenario. We also used real electricity prices from 33 locations in the US that were publicly available.

While we find that RED-BL finds a useful application in electricity cost optimization of geo-diverse data centers, our thesis is not the final word on this problem. Several avenues of further exploration are yet to be explored. We have listed some future work in Chapter 6.

Chapter 5

Case Study II: Cellular Networks

5.1 Prelude

Cellular networks consume several tens of TWhs (terawatt-hours) of electrical energy worldwide every year [36], exacerbating rising ecological concerns. Beyond these concerns, the corresponding cost of electricity makes up a significant proportion of the overall operational cost of a cellular service provider. In European markets, for example, the electricity cost is estimated to be around 18% of the operational costs [5]. This fraction is even higher in developing regions due to the shortage of grid electricity and the use of small-scale generation powered by diesel fuel. Thus, cellular operators are keen on reducing the power consumption of their networks.

In this chapter, we discuss RED-BL's application to save energy in cellular networks. To this end, we utilize *Base Transceiver Station (BTS) Power Savings* and *Call Hand-off* to reduce the BTS power consumption. Using deployment and traffic data from a cellular provider in a large metropolitan area in a developing region, we show that RED-BL can save about 10% in the amount (and cost) of electrical energy. This translates into millions of dollars in annual savings, just for this one service provider.

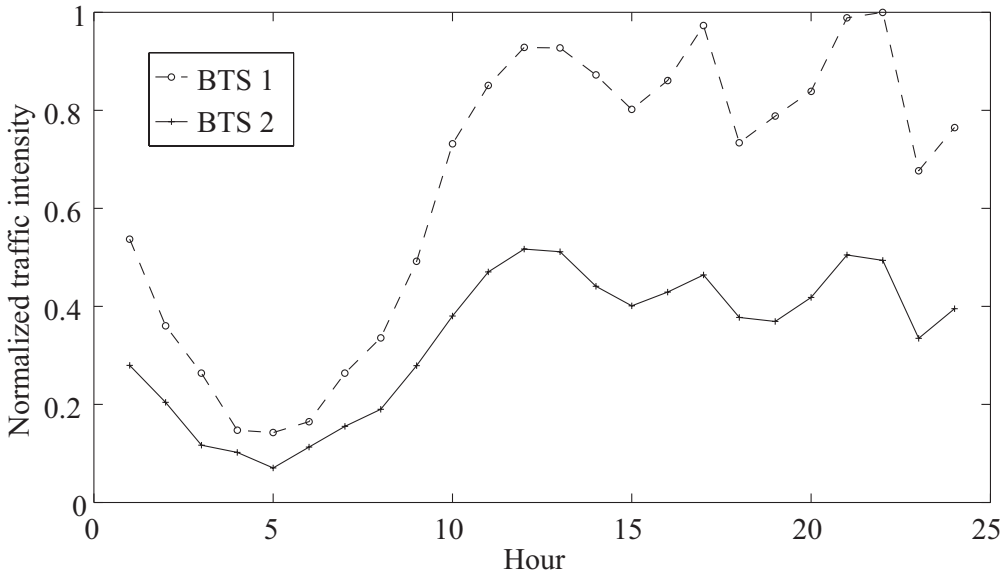


Figure 5.1: Traffic load variations at two neighboring BTSs during a single day from our dataset. For most of the day, the instantaneous load is a fraction of the peak traffic load.

RED-BL achieves this saving in energy consumption by reducing the power consumption at BTSs, which account for 60% to 80% of a cellular network's total power consumption [36, 35, 8], by making them more energy proportional. Figure 5.1 shows the normalized load of consumer traffic at two neighboring BTSs in our dataset. Observe that while the traffic load peaks for a short period of time, it mostly stays at a fraction of its peak[1].

However, the radio circuitry at BTSs is provisioned in accordance with the peak load. This radio circuitry lacks energy proportionality i.e., its power consumption does not scale in proportion to the traffic load. As a result, a BTS also lacks energy proportionality and consumes power at about the same level as it would at the peak load [8]. This leads to wasted energy, which RED-BL aims to reduce—by varying the power consumption at a BTS in accordance with its traffic load.

If the instantaneous power consumed at a BTS can be made proportional to the instantaneous workload, savings in power consumption will ensue. For the provider data available

[1]Operational cellular networks have been widely observed to exhibit traffic load variations both over time and space [8].

to us, a fully energy proportional BTS subsystem of a cellular network would save between 44%−52% of electrical energy depending on the BTS model. Thus, there exists potential to save electricity cost in a cellular network by reducing the energy consumption at low workloads. RED-BL exploits this potential in two ways: (i) **Resource pruning**: shutting down part of the radio circuitry, and (ii) **workload relocation**: rerouting calls from one BTS to a nearby BTS.

Coarse-grained energy proportionality in BTSs may be achieved in one of the following two ways:

- BTSs may be turned off when traffic is low and turned on later when traffic load increases [77, 78, 79, 80, 36, 81]. However, operators are often reluctant to switch on/off entire BTSs due to coverage and equipment lifetime concerns.

- *Frequency dimming* [82] proposes to turn off a fraction of the radio circuitry when traffic is low, such that the traffic may be handled by the circuitry that stays on. Most vendors' BTSs support such a feature, which we term as *BTS power savings* in this paper. Our conversations with cellular operators reveal that they regularly use this feature. RED-BL also makes use of this feature.

The novelty in our approach is that we couple BTS power savings with call hand-offs to increase the energy savings possible through BTS power savings alone. While call hand-off is commonly used to reduce mobile station (MS) transmitted power, to the best of our knowledge, our work is the first to exploit call hand-offs for reducing BTS power consumption.

Traffic traces collected from a large network operator indicate that if some calls are handed off between neighboring BTSs, the number of BTSs that can be put in BTS power savings mode can be increased. Thus, RED-BL proposes to hand-off calls between neighboring BTSs, without making a negative impact on the network quality of service, such that the *BTS power savings* can be applied to a maximal number of base stations throughout

the cellular network. In comparison to uncoordinated *BTS power savings*, as used in current cellular network deployments, RED-BL offers additional power savings as it may allow a larger number of radio circuits to be deactivated.

The underlying assumption in RED-BL is that calls can be handed off to neighboring BTSs without being dropped and without exceeding their traffic capacity. This is possible because (a) traffic load in cellular networks exhibits significant variation over time and space and (b) most callers often receive sufficiently strong signal from *several* nearby BTSs [8, 83]. This coverage diversity is evident in Figure 5.2, which shows the CDF of the number of BTSs available to an end-user in our dataset of live traffic. Observe that the results show that about half of the callers have 3 or more candidate BTSs available at all times Thus, some calls may be handed off from one BTS to a nearby BTS in order to increase energy savings over those possible through BTS power savings alone.

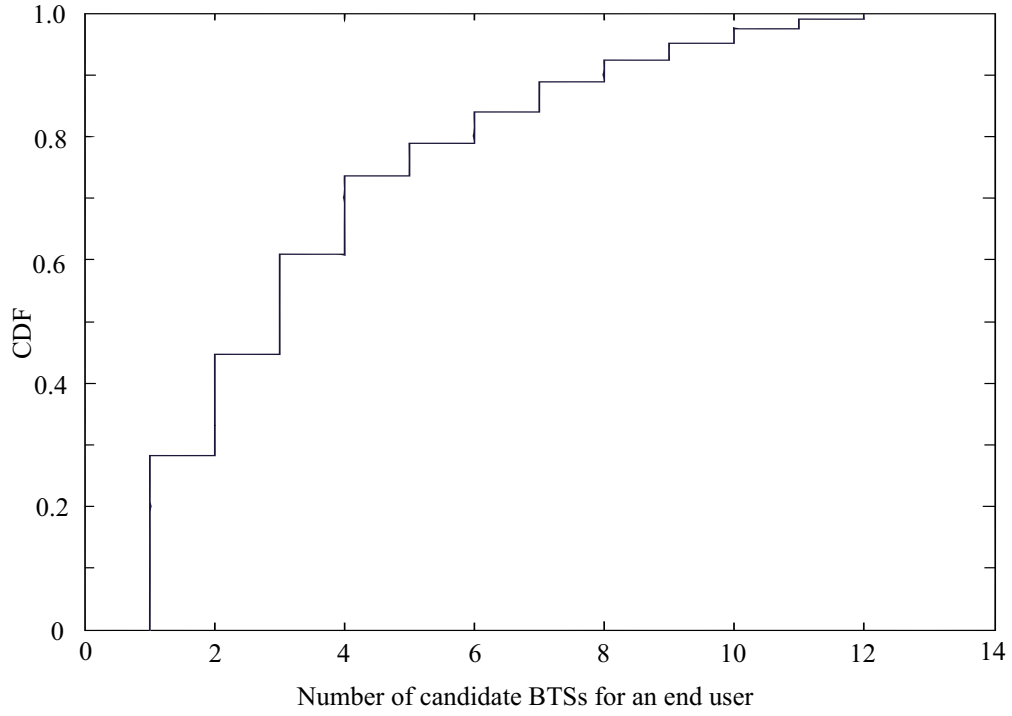


Figure 5.2: Cumulative distribution function (CDF) of the number of potential serving BTSs for a call in our dataset (large metropolitan area).

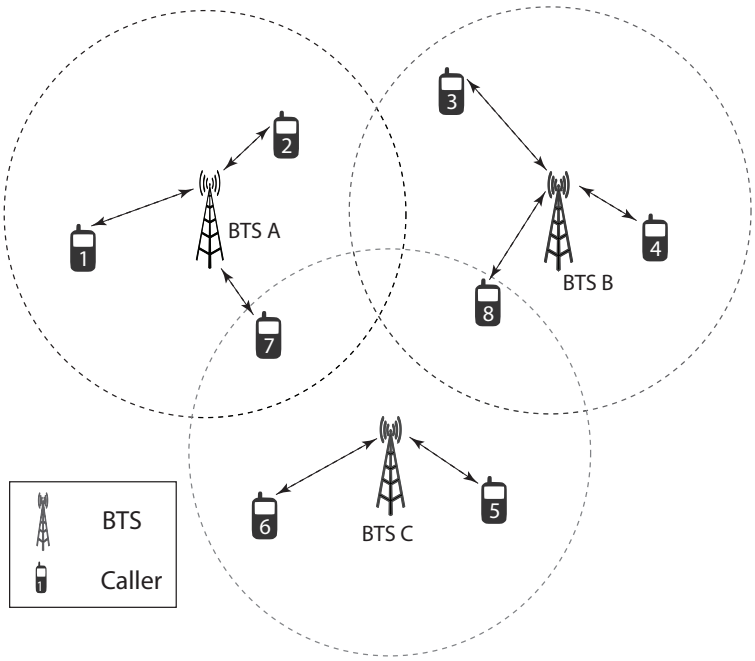


Figure 5.3: The scenario for the motivating example. Three BTSs (A, B and C) are shown along with eight active calls. Each call is handled by the BTS from which it receives the strongest signal (the default in GSM networks). The serving BTS for each call is shown using arrows. If the power savings mode can be enabled at a BTS that has up to two active calls, then only BTS C can be put in the power savings mode. However, if calls 7 and 8 were handed off to BTS C, both BTS A and B can be put into the power savings mode, thereby resulting in greater energy savings.

During low traffic periods, network operators often use a feature available in most vendor's equipment that deactivates TRX circuits at locations that serve very few customers. Huawei calls this feature *TRX shutdown* while Ericsson calls it *BTS power savings*. We use the latter term generically in this paper. Turning off one TRX cuts down BTS power consumption anywhere from $20W$ to $100W$, depending upon the frequency band (900 or 1800) and deployed equipment [37, 84]. Thus, scaling a "6+6+6" to a "2+2+2" configuration, by deactivating 12 TRXs will result in a saving of 240W to 1200W on a single site.

Scheme	Calls being handled by BTSs			BTSs in power saving mode
	A	B	C	
Default	1, 2, 7	3, 4, 8	5, 6	None
Power saving	1, 2, 7	3, 4, 8	5, 6	C
RED-BL	1, 2	3, 4	5 – 8	A, B

Table 5.1: A comparison of schemes for BTS power savings

5.1.1 Motivating example

To illustrate the working principle of RED-BL, we consider an example deployment consisting of three BTSs serving eight calls in neighboring cells as shown in Figure 5.3. By default, each call is served by the BTS from which the mobile station receives the strongest signal. Assume that each BTS may handle up to six simultaneous calls and that the power-saving threshold is two calls, i.e., a BTS serving up to two calls may be put into power-saving mode.

The first row in Table 5.1 indicates that under default call association, no BTS in the example deployment of Figure 5.3 is placed in power saving mode. With this default call association, the operator may still save some electrical energy by activating the power saving mode on BTS C (second row in Table 5.1). Additional energy savings are possible by deploying RED-BL, which hands off calls 7 and 8 to BTS C and enables power saving mode on two BTSs (A and B), as shown in the third row in Table 5.1.

This example shows that the current practice of enabling power savings mode based on traffic conditions that are local to a BTS can result in sub-optimal energy savings. RED-BL achieves greater energy savings by jointly using call hand-off and BTS power savings.

5.2 Related work

The power consumption of a BTS depends on a number of factors. A BTS's power consumption increases with the number of TRXs. The frequency band, modulation scheme and operating conditions also influence a BTS's power consumption [85].

For a given traffic load, the power consumption of a BTS may be reduced by using more energy efficient designs for the components such as TRXs. One such technique is to use switch mode power amplifiers instead of linear analogue power amplifiers [86, 87, 88, 89] or other architectural improvements [90, 91]. In [91] the authors showed that energy efficiency of macro-cell based deployments deteriorates with increasing demand for higher data rate services and proposed that a hybrid deployment of residential pico cells and macro-cells be used instead. They showed that this can result in a 60% reduction in energy consumption. In contrast to [91], our focus is on making incremental changes to a deployed network that is based pre-dominantly on macro-cells to reduce the energy consumption.

In [8], the authors proposed that when the traffic in an area being served is low, the serving BTS may be completely turned off to save power. Later, when the traffic volume rises, such BTS may be turned on again. To avoid lack of coverage when a BTS is turned off, its neighboring BTSs must, however, increase their transmit power. Similar proposals are also reported in [77, 78, 79, 80, 36, 81]. In [92], Marsan et al. considered an wireless Internet access in an area covered by multiple operators. They proposed that under high traffic conditions, all networks should operate but as the traffic drops, networks could be progressively turned off to save energy. Our conversations with cellular operators in Pakistan indicate that they are reluctant to turn off entire BTSs due to reduced expected lifetime of the installed electronic equipment.

Several centralized as well as distributed algorithms for placing maximal number of BTSs in sleep mode or turning them off during low traffic load have been proposed recently. For instance, it was proposed in [93, 94] to group nearby BTSs in an LTE network into energy partitions and then hand-off traffic out of lightly loaded BTSs to put them into sleep or turn them off. Also in the context of LTE networks, Viering et al. proposed an algorithm that adjusts transmission power levels for BTSs in response to traffic load variations with small changes to the coverage area.

Frequency dimming [82] proposes to turn off some TRXs on a BTS when the traffic is low. RED-BL goes beyond frequency dimming — it reroutes the calls and can potentially enable power savings mode on a larger number of BTSs. Coarse estimates of the energy saving potential of TRX deactivation were presented in [5]. In contrast, we use site locations and real traffic traces from a large cellular network with more than 13 million subscribers to run a simulation study assessing the benefits of dynamic equipment scaling coupled with call hand-offs. Note that BTS power saving represents RP in the RED-BL framework whereas call handoff represents WR.

There exists a large body of work on solutions to constrained resource optimization problem, like the one we formulate for RED-BL. For example, such problems have been addressed in the context of data centers [95, 21, 46, 47, 14], scheduling in compute clusters [96], System on Chip (SOC) [58], electric power systems and smart grids [59, 60, 61, 62], WiFi access points [63], wide area networks [64] and high performance computing [65, 66, 67, 68, 69, 70].

5.3 Instantiating the generalized optimization formulation

The RED-BL optimization problem minimizes the sum of state and transition costs for a network over a sequence of consecutive intervals in a planning window. The state cost for a particular interval is the sum of electricity cost incurred at all BTSs in the network. In order to compute the state cost, we will first derive a mathematical model for electricity consumption at a single BTS. Then, we will generalize it to a multi-BTS cellular network setting to represent the state cost for a single interval. The transition cost would be the cost of electricity incurred in activating or deactivating TRXs. Our conversations with network operators reveal that transitions into and out of BTS power saving mode do not consume a noticeable amount of power. These state transitions are reasonably fast in contrast to the

delayed convergence due to transitions in the geo-diverse data center scenario. The RED-BL transition costs in cellular networks may, therefore, be ignored.

If δ is the traffic capacity of a single TRX, then the traffic handling capacity of BTS with r TRXs is $r\delta$. If x_i is the number of calls currently in progress at BTS i, and the power consumed by the BTS under no load and full load is P_{min} and P_{max}, respectively, then the instantaneous BTS power consumption at the BTS[2], P_i, approximated as an affine function of its traffic load [8], is given by

$$P_i = P_{min} + x_i(P_{max} - P_{min})/r\delta \tag{5.1}$$

If a TRX's no-load power consumption is γ (its value depends on the equipment type [37, 84]), then $P_{min} = a\gamma$, where a is the number of active TRXs. Thus, by scaling the number of active TRXs in response to traffic volume changes, BTS power consumption may be reduced. For example, if the traffic volume at a BTS falls below a power-saving threshold $r\delta/2$, then half of the TRXs may be turned off. As a result, the BTS will transition to the low-power mode whereby the power consumption profile drops by an amount $r\gamma/2$ as shown in Figure 5.4. The slope of the power consumption profile remains the same whether or not the BTS is operating in low-power mode.

If the BTS is switched into low-power mode as soon as traffic falls below $r\delta/2$, then short time scale variations in traffic might cause a TRX to turn on/off rapidly. Since this may be detrimental to a TRX's lifetime, low-power mode is activated only when traffic reaches threshold $r\delta/2-\epsilon$. Here, ϵ is a parameter which may be set equal to 0 to achieve an aggressive power saving strategy of turning off a TRX as soon as opportunity presents itself. On the other extreme, ϵ may be set equal to $r\delta/2$ in which case a TRX is never turned off. These

[2]In the following discussion, when we refer to BTS power consumption, the implication is the component of BTS power consumption due to TRXs only. Since TRXs account for a large fraction of BTS power consumption [37], minimizing TRX power consumption will reduce BTS power consumption.

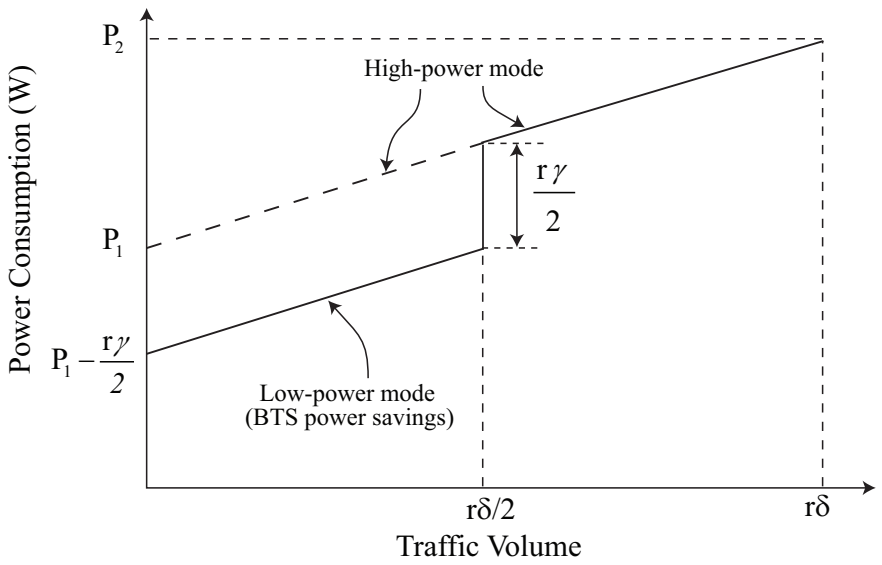


Figure 5.4: Two-state power consumption model for a BTS with r TRXs. Low-power (BTS power savings) mode is optional and kicks in at low loads.

two extremes relate to a trade off between equipment lifetime and energy savings.

Instead of the all-or-half approach of Figure 5.4, power-saving may also be applied in a more granular way, such as that shown in Figure 5.5, whereby one-third of the TRX may be switched on/off independently in response to current traffic volume. In general, the number of granular steps in the power-saving strategy is limited by the number of TRXs installed on a BTS.

5.3.1 Multi-BTS cellular setting

Since BTS power consumption is an additive function of the number of active TRXs, to minimize the power consumption over the network, we must minimize the total number of active TRXs. Consider a cellular network consisting of m BTSs and n callers. Let $c_{i,j}$ be a binary variable which is equal to one if caller i *can be* served through BTS j and zero otherwise. Also, let $w_{i,j}$ be a binary variable which is equal to one if caller i *is* served through BTS j and zero otherwise. Suppose that on a particular BTS, we may independently turn

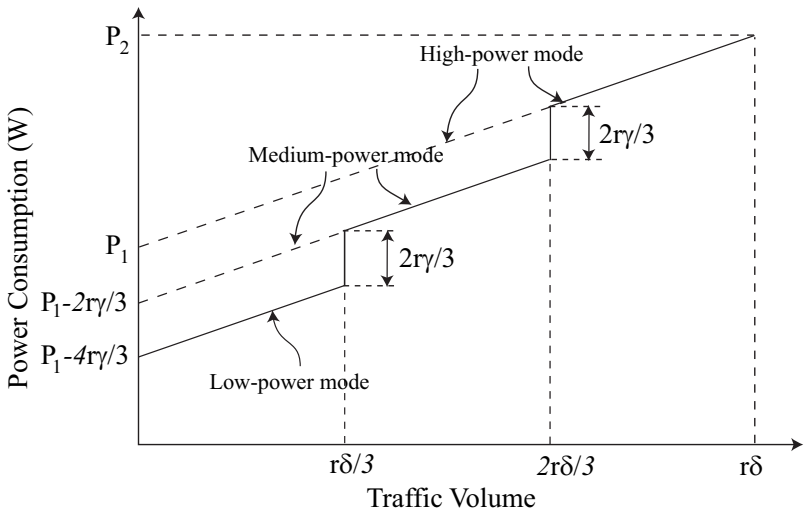


Figure 5.5: Three-state power consumption model for a BTS with r TRXs. BTS power savings is applied in a more granular way than the model of Figure 5.4.

on/off a block of β TRXs at a time.

Our conversations with network operators reveal that the transition costs resulting from activating or deactivating TRXs using BTS power saving feature of the BTSs is negligible. Thus, the $T(.)$ part of equation 3.1 may be approximated as zero. According to equation 5.1, the state cost part, $C(.)$, from equation 3.1 is a linearly increasing function of the total number of TRXs in the network. If y_j is the number of active TRX blocks at BTS j, then the RED-BL optimization problem may be given as:

$$minimize \quad \sum_{j=1}^{m} y_j \tag{5.2}$$

subject to the following constraints:

$$\sum_{j=1}^{m} w_{i,j} = 1 \qquad \forall i \tag{5.3}$$

$$w_{i,j} \leq c_{i,j} \qquad \forall i, j \tag{5.4}$$

$$\delta\beta y_j - \alpha - \sum_{i=1}^{n} w_{i,j} \geq \epsilon \tag{5.5}$$

$$1 \leq y_j \leq r/\beta \qquad (y_j \in \mathbb{N}) \qquad \forall j \tag{5.6}$$

The first constraint (Eq. 5.3) ensures that every call is served by exactly one BTS. The second constraint (Eq. 5.4) ensures that a call is served by a BTS that *can* serve it, thereby securing the uplink and downlink transmit power budget. The third constraint (Eq. 5.5) ensures that the number of active TRXs is large enough such that there is a residual call capacity for at least ϵ more calls[3]. The fourth constraint (Eq. 5.6) specifies the range of values that y_j may take.

The granularity of applying power-saving is determined in the above optimization formulation through the value of β. For instance, if β equals 1, then each TRX may be independently turned on/off. Similarly, if β equals $r/2$, the problem reduces to the two-step model of Figure. 5.4. Setting β equal to $r/3$ results in the three-step model of Figure. 5.5.

5.3.2 Problem complexity

In this section, we develop a proof of NP-Hardness for RED-BL as applied to cellular networks[4]. For this proof, we define the RED-BL problem as:

Problem Name: RED-BL

Input:

[3]A number of logical channels are reserved for control purposes in each sector. Here, α is the number of control channels per BTS

[4]This proof was done through personal communication with Dr. Mudassir Shabir

- A set of callers C
- A set of BTSs B
- Scalar values P_1, P_2, δ, γ, $t_{\max}$

Output: An assignment of each caller to exactly one BTS that minimizes the cost as given in equation X.

We also define the modified dominating set problem (MDS) as follows:

Problem Name: Modified Dominating Set

Input: A bipartite graph $G = \{E, V\}$ with sets of vertices J, K such that every edge in E connects a vertex in J to a vertex in K.

Output: A minimum cardinality set of vertices belonging to J that cover every vertex in K.

We claim that we can use RED-BL to solve MDS. If we set δ to 0 and P_1 to some non-zero value for RED-BL, then there is a set-up cost on each BTS, i.e., a fixed cost of P_1 is incurred for the first call on each BTS. Hence, in this case, the optimal solution to RED-BL will use the fewest number of BTSs to handle all calls. Therefore, if we invoke RED-BL as follows:

- Set C equal to the vertices in K
- Set B equal to the vertices in J
- Assign P_1 some positive value greater than zero
- Assign P_2 some value greater than P_1
- Assign γ some positive value greater than zero
- Set δ equal to zero

then RED-BL will solve MDS, i.e., use the fewest number of vertices in J to cover all vertices in K.

Now, consider the minimum dominating set problem (DS).

Problem Name: Minimum Dominating Set

Input: A graph $G = \{E, V\}$

Output: A minimum cardinality subset D of V such that every vertex not in D is adjacent to at least one vertex in D

We can solve DS using MDS as follows:

input : A graph $G = \{E, V\}$
output: A minimum cardinality subset D of V such that every vertex not in D is adjacent to at least one vertex in D
1 $V_1 = V$;
2 $V_2 = V$;
3 $V' = V_1 \cup V_2$;
4 $E_1 = \{(u, v) : u \in V_1, v \in V_2$ if $(u, v) \in E$ or $u = v\}$;
5 MDS(V', E_1);

Algorithm 2: Solving DS using MDS

We first create two copies of the set of vertices in G, named V_1 and V_2 and add them to set V'. We also create a set of edges E_1 as follows. If there is an edge between vertices u and v in graph G, then we add an edge from vertex u in V_1 to vertex v in V_2. Furthermore, we also add an edge between each vertex u in V_1 to vertex u in V_2. This results in a bipartite graph $G' = V', E_1$ where every edge in E_1 connects a vertex in subset V_1 of V' to a vertex in subset V_2 of V'.

Invoking MDS on G' finds the minimum number of vertices in V_1 that cover every vertex in V_2. Since V_1 and V_2 are essentially the same vertices, MDS finds the minimum cardinality set of vertices in V to which all other vertices are connected. Thus, MDS solves DS. It is well know that DS is NP-Hard [97]. Since RED-BL can solve DS, by reduction it is NP-Hard as well.

5.3.3 Heuristic solution to RED-BL for cellular networks

Even though the RED-BL problem for cellular networks is NP-Hard, for a 26 site dataset that we collected from a live cellular network, we were able to find the RED-BL optimal solution within a few seconds. However, for an entire cellular network, RED-BL becomes computationally infeasible. We propose two ways to handle this intractability. First, the entire network can be divided into several smaller regions and RED-BL can be applied to each region independently. Second, a heuristic solution to RED-BL, such as the one given in Algorithm 3, may be applied to the entire network.

The pseudo code for our proposed heuristic is given in Algorithm 3. In the first iteration of the outer loop (lines 1 through 29), our heuristic divides the set of BTSs into two sets. Set B_1 consists of those BTSs that have traffic volume greater than $(r - \beta)\delta$. The set B_2 consists of all other BTSs. The heuristic algorithm picks a BTS from the set B_1 uniformly at random. It then attempts to move this BTS to set B_2 (lines 23–26) by handing off some calls to other BTSs in B_2 without exceeding the power-saving threshold[5] (lines 12–19). In the end, blocks of β TRXs may be turned off on all members of B_2 (line 28). Further iterations of the outer loop attempt to turn off more blocks of β TRXs in a similar manner.

Our heuristic is a $O(rm^2n/\beta)$ randomized algorithm, which is not guaranteed to find the optimal solution. However, a high quality solution is likely to be obtained if the best solution is picked from amongst several invocations of the algorithm.

5.4 Experimental setup

Our dataset is obtained from a cluster of 26 BTSs operated by a large network operator with more than 7000 sites. These 26 sites are spread over a $31.25\,km^2$ urban terrain. We

[5]Note that if the traffic exceeds the power-saving threshold for one or more BTSs in the set B_2 after the calls are handed off, it would cause BTSs to move to set B_1, which is undesirable.

input : B (the set of BTSs) $= \{b_1, b_2, ..., b_m\}$,
n (the number of calls),
W (call association) $= \{w_i^j | 1 \leq i \leq n, 1 \leq j \leq m\}$,
C (Adjacency matrix) $= \{c_{i,j} = 1$ if c_i can be served through b_j, 0 otherwise$\}$

output: A new and potentially more energy efficient mapping of calls to BTSs

```
1  for k := 1 to r/β do
2      B_2 = {b_j | Σ_{i=1}^{n} w_i^j < kδ};
3      B_1 = random_shuffle(B − B_2);
4      forall the b_j ∈ B_1 do
5          a = Σ_{i=1}^{n} w_i^j − δ;
6          d = 1;
7          s = 0;
8          while d < n AND s ≤ a do
9              if w_d^j = 1 then
10                 e = 1;
11                 m = 0;
12                 while e ≤ m AND m = 0 do
13                     if e ∈ B_2 AND c_{d,e} = 1 then
14                         w_d^e = 1;
15                         w_d^j = 0;
16                         s = s + 1;
17                     end
18                     e = e + 1;
19                 end
20                 d = d + 1;
21             end
22         end
23         if Σ_{i=1}^{n} w_i^j < (r − kβ)δ then
24             B_1 = B_1 − {b_j};
25             B_2 = B_2 + {b_j};
26         end
27     end
28     Deactivate β TRXs on all BTSs ∈ B_2;
29 end
```

Algorithm 3: Energy-saving heuristic

obtained each site's coverage prediction using a tool called Forsk Atoll [98] which is quite popular in the cellular network operators community. Using this BTS coverage prediction and a caller's location, we can determine the candidate set of BTSs for the corresponding call (the c_i^j parameters).

Also available to us are the hourly cumulative traffic, in Erlang, for each of the sites, spanning two consecutive weekdays. The traffic remained remarkably similar across both days for each site. We have, therefore, only used one day's traffic data in our experiments.

Using the above datasets, we conducted a set of simulation experiments mimicking a 24-hour operation of a cellular network comprising 26 cells. Each experiment is a discrete event simulation of the arrival and placement of calls. Under the assumption of Poisson call arrivals and given the hourly traffic intensity (in Erlang) for each BTS, we use Little's law to determine the corresponding Poisson call arrival rate. Our simulator schedules call arrivals for *each* BTS according to this Poisson call arrival process. For each call, its departure event is also scheduled based on an exponential distribution of call durations with a mean of 180 seconds [99].

The location where each call originates in a cell is uniformly distributed over the corresponding BTS's coverage area. Based on the randomly picked location at which a particular call originates, the entries $c_{i,j}$ are determined such that $c_{i,j} = 1$ if call i is within service range of BTS j.

To mimic present day practices in operational networks, our simulator associates each call to the BTS from which the received signal is strongest. Using this call-BTS association, a time series of traffic load for each BTS is calculated, which determines the power consumption profile for each BTS. Integrating the power consumption time series for each BTS gives its energy consumption over a 24 hour period. Summing the energy consumption for all BTSs gives the network's total energy consumption. This number represents a baseline for assessing the performance of the energy efficiency improvement techniques that we investigate.

Iterating over the traffic load time series for each BTS, our simulator places those BTSs that have sufficiently low traffic into power-saving mode. Using the power consumption functions given in equation 5.1, our simulator calculates the energy consumption for the network over a 24 hour period when BTS power-saving is applied.

During the simulation, at a configurable frequency, we also invoke the RED-BL optimization problem on the current call traffic. We thus determine the energy consumption for the network over a 24 hour period when call hand off and BTS power saving are applied in a coordinated fashion.

If RED-BL is invoked very frequently, the network will remain in an optimal state most of the time. Therefore, an aggressive re-optimization scheme will enable greater energy savings. In order to study how the energy savings scale with re-optimization frequency, we experimented with a range of intervals between successive optimizations, ranging from a minute to an hour.

5.4.1 Site characteristics

All sites in our dataset had three sectors, each equipped with 6 TRXs. The maximum number of simultaneous calls for each site is 132 [6].

The BTS power consumption model parameters may vary from one equipment to another. In this thesis, we use three different sets of model parameters as listed in table 5.2. We now describe the sources and methods from which we obtained these models.

[6]Each TRX's frequency is shared in time-domain by 8 calls for a total of $3 \times 6 \times 8 = 144$ channels. Four channels in each sector were reserved for control and broadcast purposes, resulting in 132 channels available for voice calls. The half-rate codec feature of GSM standard can be used to handle greater traffic volume, but we do not consider it in the present work in favor of model simplicity.

Model 1

For the first model, we have used 1.5 kW as the maximum power consumption [6], a 20 W per TRX saving when scaling the BTS down [84] and a 5 % variation in power consumption between no-load and full-load [8].

Model 2

Lorincz et al. reported the single sector DC power consumption for a GSM 900 BTS [37]. The sector under consideration had 7 TRXs, as opposed to 6 TRXs in our case. To approximate the DC power consumption for a site with 3 sectors, each with 6 TRXs, we scaled the power consumption by a factor of $3 \times 6/7$. The DC power consumption does not include the AC power consumed in the power supply units and in air-conditioning. We must, therefore, also compensate for those, to obtain the overall site power consumption. Power supply unit load is negligible compared to air-conditioning (typical A/C power consumption of 1 kW [6]). We applied this scaling and addition to the minimum reported DC power consumption for the GSM 900 site to obtain an approximate value of P_{min} for a site comparable to ours. Similarly, we used the maximum reported DC power consumption and applied the scaling and AC load correction to approximate the value of P_{max}. Furthermore, the authors measured a drop of 50 W in power consumption when a TRX is disabled, which gives us the value of γ as listed in Table 5.2.

Model 3

We used the measurements for the GSM 1800 BTS reported in [37] to determine the values for for P_{min} and P_{max} in the same manner as described in section 5.4.1. The value of γ was reported to be 100 W [37]. The parameter values for this model are given in Table 5.2

Parameter	Value		
	Model 1	Model 2	Model 3
P_{min}	1425	2401.8	2341.5
P_{max}	1500	3887.5	2973.9
γ	20	50	100

Table 5.2: BTS model parameter values

5.5 Results

We now present the results of the simulation experiments. These experiments were conducted on all three BTS models with varying inter-optimization interval.

5.5.1 BTS with two possible power states

First, we consider the benefit of BTS power-saving alone, compared to running the network in the default configuration. The percentage reduction in energy consumption is listed in Table 5.3. The results indicate that a saving of between 4% and 12% can be achieved in a network just by activating BTS power savings. We note here that some of these results are in agreement with Ericsson's claim of saving 10-20% energy by using BTS power-saving on Germany's Vodafone network [100].

In absolute terms, this represents a cumulative saving of between 43 kWh and 217 kWh per day on 26 BTSs. Now, consider that there are five cellular operators in Pakistan: Mobilink with more than 8500 sites [101], Ufone with more than 8000 sites [102], Zong with more than 5500 sites [102], Telenor with more than 7000 sites [103] and Warid with more than 4500 sites [102]. Overall, there were more than 31000 sites in Pakistan at the end of 2011. We extrapolated the daily energy savings number over 26 BTSs to calculate the daily energy savings possible for a country like Pakistan with over 31000 BTSs (see the last row of Table 5.3). The results indicate that mere activation of BTS power saving option itself can save quite a bit of electrical energy, a critical resource, especially in a developing country

Energy saving	Model 1	Model 2	Model 3
Percentage	4.73%	5.43%	12.89%
Daily absolute saving over 26 BTSs (in kWh)	43.28	109.68	217.12
Country-wide daily saving over 31000 sites (in MWh)	51.6	130.77	258.87

Table 5.3: Energy savings by using BTS power savings only

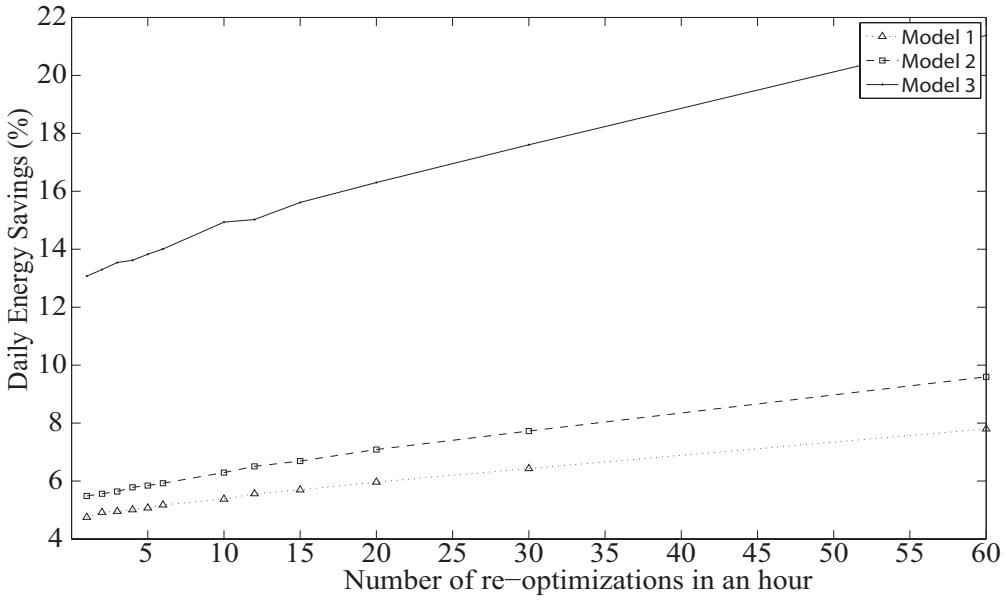
(see last row in Table 5.3. As we shall see next, greater energy savings are possible if we couple periodical call shuffling with) BTS power savings in the network.

If periodic optimization of call placement is coupled with BTS power-saving, the energy saving improves, as shown in Figure 5.6(a). For all three BTS models, we see an almost linear increase in power saving as the duration of the re-optimization interval is decreased. Recall that the three models are significantly different in terms of power consumption (see Table 5.2). Therefore, we can not directly say that since Model 3 BTS offers the highest percentage reduction in energy consumption, it also saves the most energy (in kWh).

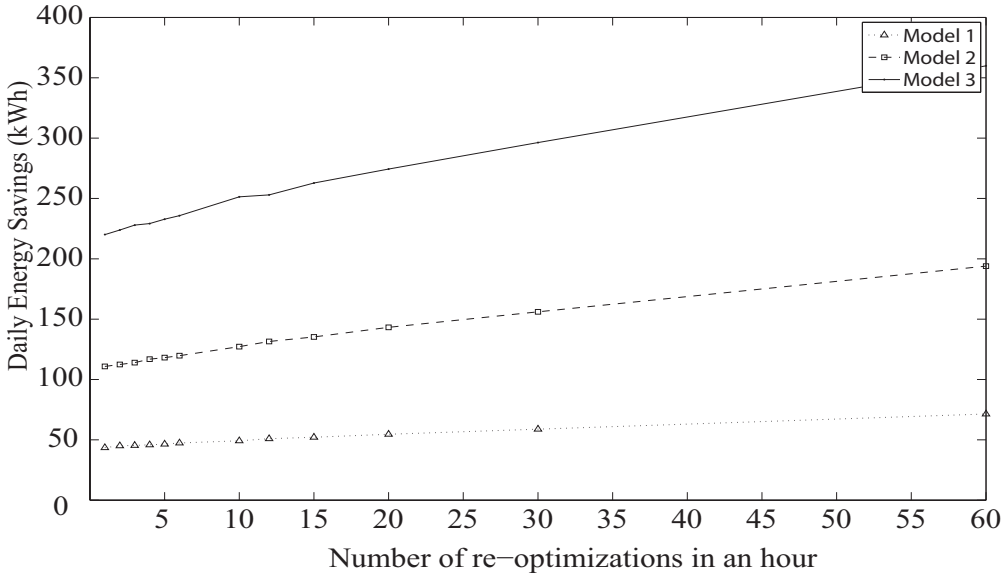
To *compare* the three BTS models in terms of energy saving potential, we also present the absolute reduction in energy consumption for the three BTS models in Figure 5.6(b). We see the same linear trend along with the same relative order of the three models in terms of amount of saved energy, as in Figure 5.6(a).

Re-optimizing at an interval less than the mean call duration should offer greater savings than a less frequent re-optimization, because the former regime is able to optimally assign BTSs to most of the calls at least once. This is confirmed in our results. For instance, the gain in energy savings for Model 1 BTS when going from a 60 minutes inter-optimization interval to 30 minutes gains an energy saving of only 0.0506 kWh per minute, while decreasing the inter-optimization interval from 2 minutes to 1 minute gains 12.5421 kWh.

Let us now interpret what these results mean physically in terms of ecological impact. If we extrapolate our results, the total energy saving for Pakistan are projected to be



(a)



(b)

Figure 5.6: (a) Percent reduction in energy consumption vs re-optimization interval, (b) Reduction in energy consumption vs re-optimization interval

60.72 MWh, 156.84 MWh and 301.61 MWh daily, respectively, according to the three BTS models. These savings in energy are significant, especially for small and developing countries. Since network deployments and traffic patterns are similar in different countries, we also expect that similar savings should be achievable in many other countries as well.

In the above extrapolation, we have assumed that the same amount of energy saving would be applicable in rural as well as urban settings. While this may not necessarily be true because the deployments are sparse in rural settings, resulting in reduced potential to save energy by means of call hand-off to neighboring sites, the potential to save energy merely by BTS power-saving should be higher in a rural setting because traffic loads are typically lower.

5.5.2 Multi-state BTS

In our experimental results discussed so far, we have observed that going from a 6+6+6 configuration to a 2+2+2 configuration can save a significant amount of energy. Intuition suggests that going to a finer granularity of resource pruning should enable greater energy savings. We now present two cases that are different from the configuration considered so far. In the first case, we consider the ability to (de)activate TRXs in pairs, i.e., a site may be in one of three configurations at a given time: 6+6+6, 4+4+4 or 2+2+2. In the second case, we consider the ability to (de)activate each TRX on a site independently.

In this scenario, we conducted simulation experiments where a re-optimization was performed every six minutes using all three BTS models. The results of these experiments are given in Table 5.4. For all three BTS models, we see that going from a 2-state model to a 3-state model gives a relatively small increase in energy savings. However, when all TRXs may be independently controlled, we get a significant improvement in energy savings.

Granularity	BTS Model 1	BTS Model 2	BTS Model 3
2-State	5.38%	6.29%	14.94%
3-State	6.81%	7.73%	18.62%
r-State	8.70%	9.65%	23.37%

Table 5.4: Percentage electricity savings for different granularity of resource pruning

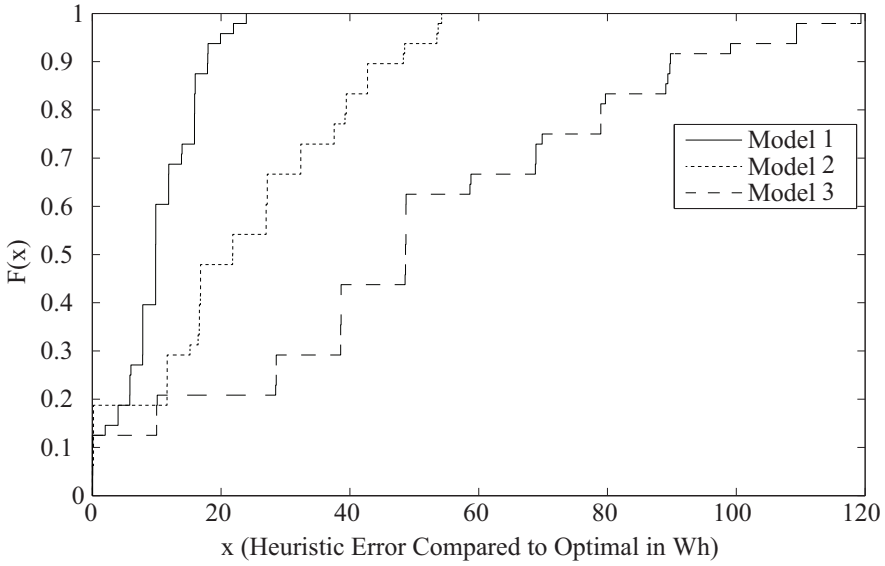


Figure 5.7: Empirical CDF of the difference between the cost offered by our heuristic compared to the optimal

5.5.3 Performance of heuristic algorithm

We also ran experiments for each BTS model in which the electricity cost for the optimal as well as the heuristic algorithm (Algorithm 1) was computed. We assessed the performance of our heuristic by computing the difference (error) in the electricity cost of the two solutions. For statistical significance, we computed the error in our heuristic relative to the optimal solution over 48 different experiment runs for each BTS model. The resulting CDF of the heuristic error (in Wh) is plotted in Figure 5.7. We can see in Figure 5.7 that our heuristic algorithm 1 is quite close to the optimal solution most of the time, especially for the Model 1 and Model 2 BTS. For Model 3 BTS, although the error is comparatively larger, but since

the amount of savings with the optimal solution is quite high (Figure 5.6(a)), the heuristic will still result in significant energy savings.

5.5.4 Sensitivity to the value of ϵ

If the value of ϵ in our optimization is set too aggressively, a BTS may oscillate at times between low power and high power states rapidly due to short time scale traffic variations. Such state oscillations may be undesirable and to avoid these, the value of ϵ must be set at a safe value. Furthermore, if ϵ is set too aggressively, a BTS placed in low-power mode would be operating very close to its *new* and lower traffic capacity. If several calls arrive in a short time window, the BSC may not have sufficient time to bring the BTS back into high-power mode and, thus, some calls may be blocked. However, if ϵ is set too conservatively, the energy savings would be smaller.

We carried out experiments to assess the impact of the value of ϵ on the energy savings achievable through RED-BL. For this purpose, we fixed the inter-optimization interval at 6 minutes and carried out RED-BL optimizations for all three BTS models. Furthermore, we considered a two-state BTS model, i.e., a BTS may be placed in either a 6+6+6 or a 2+2+2 configuration. The range of possible values for epsilon were 5, 10, 15 and 25. Since the traffic capacity of a 2+2+2 BTS is 44[7], any larger value for ϵ did not make sense. Figure 5.8 shows the results. As expected, the percentage savings deplete almost linearly with increasing values of ϵ.

5.6 Summary

In this chapter, we have seen an application of RED-BL to cellular networks. The nature of cellular networks is different compared to geo-diverse data centers (considered in the previous

[7]The capacity of the 2+2+2 BTS is $3\times2\times8 = 48$, but 4 channels were reserved by the operator for control and broadcast channels.

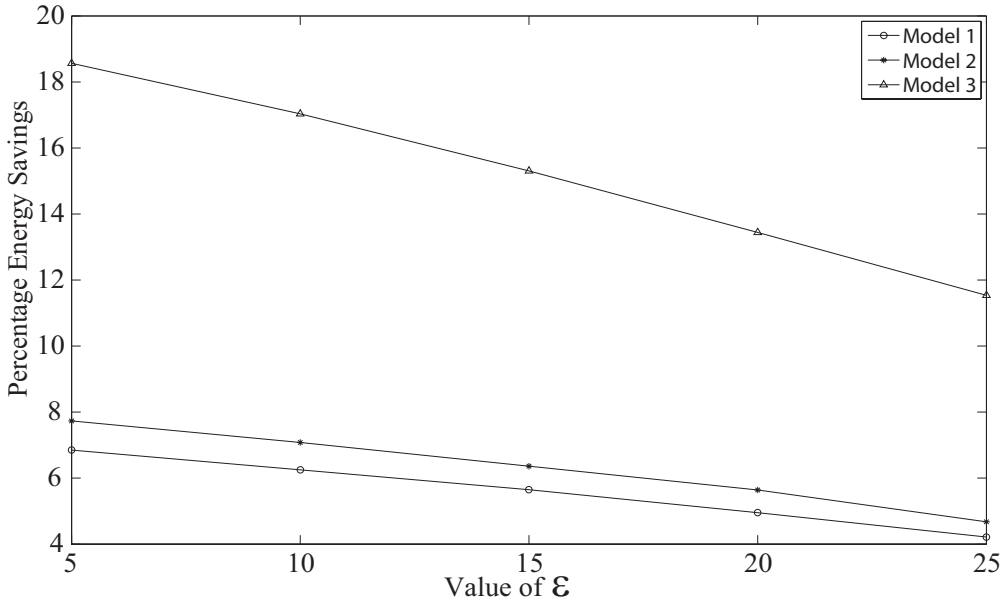


Figure 5.8: The percentage energy savings for all three BTS models considered vs the value of ϵ, with a six minute inter-optimization interval

chapter) in the following ways:

- The workload (calls) has limited geographic flexibility and may only be handled at a few candidate resources (BTSs)
- There is no geo-diversity in electricity prices

For resource pruning and workload relocation, we utilized two features built in to currently deployed equipment in the form of BTS power savings and network-controlled call hand-off, respectively. Our simulation results show that jointly using workload relocation and resource pruning can bring in considerably higher gains in electricity savings compared to using resource pruning alone. Our results show the promising potential that RED-BL holds in greening current generation cellular networks.

Chapter 6

Conclusions and future work

Geo-diverse data centers and cellular networks are quite similar in some respects and hence may be abstractly represented as a set of resources. Each of these resources is constrained by the maximum amount of workload it can handle. Furthermore, for both cellular networks and geo-diverse data centers, the instantaneous power consumption for the resources is similar, i.e., an affine function of the amount of workload they handle. The no-load power consumption for the resources in these networks is a large fraction of the full-load power consumption.

The workload is quite variable, and the network must be dimensioned according to peak workload demand. However, the workload peaks for only a short duration and drops to a much lower trough. Thus, these networks are energy inefficient and their electricity cost is quite high. One way to improve the energy efficiency and save electricity costs is to scale the networks resources in response to workload variations.

We represent the status (on or off) of network resources and the workload mapped to them as a network state. Using this representation, we model the energy efficiency improvement problem as an optimal state trajectory problem, which we call Relocate Energy Demand to Better Locations (RED-BL). We used workload traces collected from real networks and real

electricity costs to assess the utility of RED-BL. In doing so, we have made the following contributions:

- We identified abstractions for a generic power consumption model applicable to two different networks: geo-diverse data centers and cellular networks. Our model represents these networks as a set of resources that have an associated workload handling capacity. When a resource handles workload, it consumes some power according to a power consumption function.

- The aggregate mapping of workload to resources may be viewed as a network state. We accordingly model the electricity cost minimization problem in networks as a multi-interval optimal state trajectory problem.

- We provide a mathematical optimization formulation for the optimal state trajectory problem. The formulation is parameterized and abstract for broad applicability. We modeled state transition costs as being a fraction of the cost of a state in a single interval.

- We apply the mathematical optimization problem to geo-diverse data centers as well as cellular networks with the following contrasts:

  - The optimal mapping of workload to resources is a discrete optimization problem. The workload capacity of a single resource is quite large in case of geo-diverse data centers, hence any fraction of workload mapped to a resource is likely to be quite close to a whole number of client requests. If this is not the case, the number of client requests mapped to a resource may be rounded to the nearest integer. The difference in power consumption resulting from this rounding is expected to be quite small. Hence, the optimal mapping of workload to resources, which is a discrete optimization problem may be relaxed to a fractional problem

without much error in case of geo-diverse data centers. In cellular networks, on the other hand, the number of workload units being handled simultaneously by a single resource is much smaller. Thus, a fractional mapping of workload to resources in a cellular network may represent a call being handled by multiple BTSs simultaneously, which does not make sense. Hence, a fractional relaxation to the optimization problem is not possible in case of cellular networks.

  - A call may only be handled by a restricted set of nearby BTSs. In contrast, in a geo-diverse data center setting, it is common for applications to be replicated across data centers and in such cases, a client request may be handled at any data center.
  - Geo-diverse data centers are so far apart that geographic diversity in electricity prices is quite apparent. Meanwhile, BTSs in cellular networks are not too distant and geographic diversity in electricity prices is not present.
  - The transition costs in geo-diverse data centers are expected to be significant. However, in cellular networks, the electricity cost impact of resource activation and deactivation is negligible.

- We show that the electricity cost minimization problem is NP-Complete in both geo-diverse data center and cellular network scenarios and provide heuristic algorithms for solution of the problem in each of these networks. We were also able to solve reasonably sized problems for both network types.

- We studied the sensitivity of the state trajectory problem to variations in the parameter values such as the magnitude of transition costs relative to the cost of a state in a single interval.

6.1 Scope of our work

Like all research work, our work's scope is not unlimited. To the best of our knowledge, the following list covers the scope of our work:

- We only considered resource activation and deactivation costs as the source of transition costs. Data replication costs in geo-diverse data centers have not been evaluated because of (to the best of our knowledge) lack of models in the literature. Coming up with a general model for this cost is beyond the scope of this thesis.
- For geo-diverse data centers, we have experimented with day ahead electricity prices for the US market.
- RED-BL computes the hourly traffic volume to be mapped to each data center. It does not provide a mapping of this workload to individual servers within the data center. This is a complementary research problem, which is presently receiving significant research attention.
- In case of cellular networks, the ability to handle greater traffic volume through the half-rate codecs has not been considered.

6.2 Future work

Some of the avenues for future inquiry related to our work are:

- Study the inter-data center traffic to examine if there is a relationship between the volume of such traffic with the number/nature of client requests, duration of the interval for which data volume is measured, or some other variables. This may help build a model for the expensive inter-data center traffic. Such a model may be integrated with RED-BL to have a more elaborate optimization framework that is sensitive to

the potential increase in inter-data center traffic due to data center elastic resource (de)activation.

- Replication of data stores across data centers requires some overhead traffic. An empirical study could be performed to build a model for such traffic in a few representative scenarios such as news websites, social networking sites, micro-blogging etc. Our work saves electricity cost by turning off elastic resources. This may mean taking the application data stores offline. When the elastic resources come back online, they would need to bring their data stores in sync with the rest of the data centers. The results of the aforementioned empirical study could be used to predict the volume of traffic that would be generated during this re-synchronization event.

- The deployment of a small-scale GSM testbed using open source GSM software could be done to validate the results of our simulation study.

- RED-BL may be applied to other networks such as generation resource scheduling in smart grids or packet switching networks.

Bibliography

[1] J. Verge, "Google Pumps $400 Million More into Iowa, Investment Now Tops $1.5 Billion," April 2013. [Online; accessed 23-May-2013].

[2] "Mobile Broadband: The Benefits of Additional Spectrum," tech. rep., Federal Communications Commission, October 2010. White Paper.

[3] R. Miller, "Facebook: $50 Million A Year on Data Centers," September 2010. [Online; accessed 24-May-2013].

[4] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," in *Computer Communications Review*, vol. 39, January 2009.

[5] O. Blume, "Energy savings in mobile networks based on adaptation to traffic statistics," *Bell Labs Technical Journal*, vol. 15, pp. 77–94, September 2010.

[6] S. Mbakwe, M. T. Iqbal, and A. Hsaio, "Design of a 1.5kW Hybrid Wind/Photovoltiac Power System for a Telecoms Base Station in Remote Location of Benin City Nigeria," in *IEEE NECEC*, November 2011.

[7] T. Italia, "Telecom Italia Annual Report 2012," January 2013. [Online; accessed 24-May-2013].

[8] C. Peng, S.-B. Lee, S. Lu, H. Luo, and H. Li, "Traffic-driven power saving in operational 3G cellular networks," in *ACM MobiCom '11*, pp. 121–132, 2011.

[9] X. Fan, W.-D. Weber, and L. A. Barroso, "Power provisioning for a warehouse-sized computer," in *ISCA '07: Proceedings of the 34th annual international symposium on Computer architecture*, (New York, NY, USA), pp. 13–23, ACM Press, 2007.

[10] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and S. Diot, "Packet-level traffic measurements from the sprint ip backbone," *Network, IEEE*, vol. 17, no. 6, pp. 6–16, 2003.

[11] L. A. Barroso and U. Holzle, "The case for energy-proportional computing," *Computer*, vol. 40, pp. 33–37, 2007.

[12] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, pp. 50–58, Apr. 2010.

[13] V. Inc, "Reduce Energy Costs and Go Green With VMWare Green IT Solutions," March 2009. [Online; accessed 26-May-2013].

[14] J. S. Chase, D. C. Anderson, P. N. Thakar, A. M. Vahdat, and R. P. Doyle, "Managing energy and server resources in hosting centers," *SIGOPS Oper. Syst. Rev.*, vol. 35, pp. 103–116, Oct. 2001.

[15] G. Chen, W. He, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao, "Energy-aware server provisioning and load dispatching for connection-intensive internet services," in *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'08, (Berkeley, CA, USA), pp. 337–350, USENIX Association, 2008.

[16] D. Meisner, B. T. Gold, and T. F. Wenisch, "Powernap: eliminating server idle power," in *Proceedings of the 14th international conference on Architectural support for programming languages and operating systems*, ASPLOS XIV, (New York, NY, USA), pp. 205–216, ACM, 2009.

[17] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," in *IEEE INFOCOM*, 2011.

[18] A. Qureshi, "Plugging Into Energy Market Diversity," in *7th ACM Workshop on Hot Topics in Networks (HotNets)*, (Calgary, Canada), October 2008.

[19] J. Li, Z. Li, K. Ren, and X. Liu, "Towards optimal electric demand management for internet data centers," *IEEE Trans. Smart Grid*, vol. 3, no. 1, pp. 183–192, 2012.

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

[21] Y. Chen, A. Das, W. Qin, A. Sivasubramaniam, Q. Wang, and N. Gautam, "Managing server energy and operational costs in hosting centers," in *ACM SIGMETRICS*, pp. 303–314, 2005.

[22] M. Mazzucco and D. Dyachuk, "Balancing electricity bill and performance in server farms with setup costs," *Future Generation Computer Systems*, vol. 28, no. 2, pp. 415 – 426, 2012.

[23] L. Rao, X. Liu, L. Xie, and W. Liu, "Minimizing Electricity Cost: Optimization of Distributed Internet Data Centers in a Multi-Electricity-Market Environment," in *IEEE Conference on Computer Communications 2010 (INFOCOM'2010)*, (San Diego, USA), March 2010.

[24] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the Electric Bill for Internet-Scale Systems," in *ACM SIGCOMM*, (Barcelona, Spain), August 2009.

[25] "Kubernetes: Manage a cluster of Linux containers as a single system to accelerate Dev and simplify Ops." http://kubernetes.io.

[26] "Docker: An open platform for distributed applications for developers and sysadmins." http://www.docker.com.

[27] E. Schurman and J. Brutlag, "The User and Business Impact of Server Delays, Additional Bytes, and HTTP Chunking in Web Search," June 2009.

[28] P. Dixon, "Shopzilla's site redo - you get what you measure," June 2009.

[29] D. Artz, "The secret weapons of the AOL optimization team," June 2009.

[30] A. Vahdat, M. Al-Fares, N. Farrington, R. Mysore, G. Porter, and S. Radhakrishnan, "Scale-out networking in the data center," *Micro, IEEE*, vol. 30, no. 4, pp. 29–41, 2010.

[31] D. Abts and B. Felderman, "A guided tour of data-center networking," *Commun. ACM*, vol. 55, pp. 44–51, June 2012.

[32] P. Mockapetris, "Domain names - concepts and facilities." RFC 1034 (INTERNET STANDARD), Nov. 1987. Updated by RFCs 1101, 1183, 1348, 1876, 1982, 2065, 2181, 2308, 2535, 4033, 4034, 4035, 4343, 4035, 4592, 5936.

[33] P. Mockapetris, "Domain names - implementation and specification." RFC 1035 (INTERNET STANDARD), Nov. 1987. Updated by RFCs 1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2181, 2137, 2308, 2535, 2673, 2845, 3425, 3658, 4033, 4034, 4035, 4343, 5936, 5966, 6604.

[34] T. Berners-Lee, R. Fielding, and H. Frystyk, "Hypertext Transfer Protocol – HTTP/1.0." RFC 1945 (Informational), May 1996.

[35] J. T. Louhi, "Energy efficiency of modern cellular base stations," in *IEEE INTELEC '07*, (New York, NY, USA), pp. 475–476, IEEE, 2007.

[36] E. Oh, B. Krishnamachari, X. Liu, and Z. Niu, "Toward dynamic energy-efficient operation of cellular network infrastructure," in *IEEE Communications Magazine*, June 2011.

[37] J. Lorincz, T. Garma, and G. Petrovic, "Measurements and modelling of base station power consumption under real traffic loads," *Sensors*, vol. 12, no. 4, pp. 4281–4310, 2012.

[38] B.-Y. Choi, S. Moon, Z.-L. Zhang, K. Papagiannaki, and C. Diot, "Analysis of point-to-point packet delay in an operational network," in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 3, pp. 1797–1807 vol.3, 2004.

[39] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960.

[40] K. G. Brill, "The invisible crisis in the data center: The economic meltdown of moore's law," tech. rep., The Uptime Institute, 2007.

[41] C. L. Belady, "In the data center, power and cooling costs more than the it equipment it supports," *Electronics Cooling*, vol. 23, Jan. 2007.

[42] E. Kayaaslan, B. B. Cambazoglu, R. Blanco, F. P. Junqueira, and C. Aykanat, "Energy-price-driven query processing in multi-center web search engines," in *Proceed-*

ings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, SIGIR '11, (New York, NY, USA), pp. 983–992, ACM, 2011.

[43] N. Buchbinder, N. Jain, and I. Menache, "Online job-migration for reducing the electricity bill in the cloud," in *Proceedings of the 10th international IFIP TC 6 conference on Networking - Volume Part I*, NETWORKING'11, (Berlin, Heidelberg), pp. 172–185, Springer-Verlag, 2011.

[44] U. Bhaskar and L. Fleischer, "Online mixed packing and covering," *CoRR*, vol. abs/1203.6695, 2012.

[45] 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*, SIGMETRICS '11, (New York, NY, USA), pp. 221–232, ACM, 2011.

[46] M. Mazzucco, D. Dyachuk, and R. Deters, "Maximizing cloud providers revenues via energy aware allocation policies," *CoRR*, vol. abs/1102.3058, 2011.

[47] F. Y.-K. Oh, H. S. Kim, H. Eom, and H. Y. Yeom, "Enabling consolidation and scaling down to provide power management for cloud computing," in *USENIX HotCloud*, (Berkeley, CA, USA), pp. 14–14, USENIX Association, 2011.

[48] L. Rao, X. Liu, L. Xie, and Z. Pang, "Hedging against uncertainty: A tale of internet data center operations under smart grid environment," *Smart Grid, IEEE Transactions on*, vol. 2, pp. 555 –563, sept. 2011.

[49] K. Le, R. Bianchini, M. Martonosi, and T. D. Nguyen, "Cost- and energy-aware load distribution across data centers," in *Workshop on Power Aware Computing and Systems (HotPower '09)*, pp. 1–5, 2009.

[50] X. Zheng and Y. Cai, "Energy-aware load dispatching in geographically located internet data centers," *Sustainable Computing: Informatics and Systems*, vol. 1, no. 4, pp. 275 – 285, 2011.

[51] G. Koutitas and P. Demestichas, "Challenges for energy efficiency in local and regional data centers," *Journal of Green Engineering*, vol. 1, pp. 1 – 32, October 2010.

[52] Z. Abbasi, T. Mukherjee, G. Varsamopoulos, and S. Gupta, "Dahm: A green and dynamic web application hosting manager across geographically distributed data centers12," *Atlanta*, vol. 60, p. 80, 2011.

[53] L. Chiaraviglio and I. Matta, "Greencoop: cooperative green routing with energy-efficient servers," in *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, pp. 191–194, ACM, 2010.

[54] D. H. Phan, J. Suzuki, R. Carroll, S. Balasubramaniam, W. Donnelly, and D. Botvich, "Evolutionary multiobjective optimization for green clouds," in *ACM GECCO 2012*, ACM, 2012.

[55] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. Andrew, "Greening geographical load balancing," in *Proceedings of the 2011 ACM SIGMETRICS*, SIGMETRICS '11, (San Jose, California, USA), ACM, 2011.

[56] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. Andrew, "Geographical load balancing with renewables," in *Proceedings of the 2011 ACM GREENMETRICS*, GREENMETRICS '11, ACM, 2011.

[57] A. Sucevic, L. L. Andrew, and T. T. Nguyen, "Powering down for energy efficient peer-to-peer file distribution," *SIGMETRICS Perform. Eval. Rev.*, vol. 39, pp. 72–76, Dec. 2011.

[58] Z. Fang, L. Zhao, R. R. Iyer, C. F. Fajardo, G. F. Garcia, S. E. Lee, B. Li, S. R. King, X. Jiang, and S. Makineni, "Cost-effectively offering private buffers in SoCs and CMPs," in *Proceedings of the ACM ICS '11*, pp. 275–284, 2011.

[59] F. Javed and N. Arshad, "On the use of linear programming in optimizing energy costs," in *IWSOS '08*, pp. 305–310, 2008.

[60] T. Logenthiran, D. Srinivasan, and A. M. Khambadkone, "Multi-agent system for energy resource scheduling of integrated microgrids in a distributed system," *Electric Power Systems Research*, vol. 81, no. 1, pp. 138 – 148, 2011.

[61] G. Celli and F. Pilo, "Optimal distributed generation allocation in MV distribution networks," in *Innovative Computing for Power - Electric Energy Meets the Market. 22nd IEEE PICA*, pp. 81 –86, 2001.

[62] F. Javed and N. Arshad, "AdOpt: An Adaptive Optimization Framework for Large-scale Power Distribution Systems," *IEEE SASO*, pp. 254–264, 2009.

[63] M. A. Marsan, L. Chiaraviglio, D. Ciullo, and M. Meo, "A simple analytical model for the energy-efficient activation of access points in dense WLANs," in *Proceedings of the 1st International Conference on Energy-Efficient Computing and Networking*, e-Energy '10, pp. 159–168, 2010.

[64] C. Cavdar, A. Yayimli, and L. Wosinska, "How to cut the electric bill in optical WDM networks with time-zones and time-of-use prices," in *37th ECOC*, pp. 1 –3, Sept. 2011.

[65] S. Lee and S. Sahu, "Efficient server consolidation considering intra-cluster traffic," in *IEEE GLOBECOM*, pp. 1–6, 2011.

[66] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath, "Load balancing and unbalancing for power and performance in cluster-based systems," in *2nd Workshop on Compilers and Operating Systems for Low Power*, 2001.

[67] Y. Yao, L. Huang, A. Sharma, L. Golubchik, and M. Neely, "Data centers power reduction: A two time scale approach for delay tolerant workloads," in *IEEE INFOCOM 2012*, pp. 1431 –1439, march 2012.

[68] H. Herodotou, H. Lim, G. Luo, N. Borisov, L. Dong, F. B. Cetin, and S. Babu, "Starfish: A self-tuning system for big data analytics," in *5th CIDR*, pp. 261 –272, January 2011.

[69] H. Herodotou, F. Dong, and S. Babu, "No one (cluster) size fits all: automatic cluster sizing for data-intensive analytics," in *2nd ACM SOCC '11*, pp. 18:1–18:14.

[70] D. Aikema and R. Simmonds, "Electrical cost savings and clean energy usage potential for HPC workloads," in *IEEE ISSST*, pp. 1 –6, May 2011.

[71] J. Pang, A. Akella, A. Shaikh, B. Krishnamurthy, and S. Seshan, "On the responsiveness of DNS-based netowrk control," in *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pp. 21–26, ACM, 2004.

[72] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet Routing Convergence," in *Proc. of the ACM SIGCOMM*, August 2000.

[73] N. Padhy, "Unit commitment-a bibliographical survey," *Power Systems, IEEE Transactions on*, vol. 19, no. 2, pp. 1196–1205, 2004.

[74] A. Nazir, S. Raza, and C.-N. Chuah, "Unveiling facebook: a measurement study of social network based applications," in *Proceedings of the 8th ACM SIGCOMM con-*

ference on Internet measurement, IMC '08, (New York, NY, USA), pp. 43–56, ACM, 2008.

[75] W. H. Kwon and S. H. Han, *Receding Horizon Control*. Elsevier, 2005.

[76] G. Box, G. M. J. , and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*. Prentice-Hall, 1994.

[77] E. Oh, K. Son, and B. Krishnamachari, "Dynamic base station switching-on/off strategies for green cellular networks," *Wireless Communications, IEEE Transactions on*, vol. 12, pp. 2126–2136, May 2013.

[78] S. Kokkinogenis and G. Koutitas, "Dynamic and static base station management schemes for cellular networks," in *IEEE GLOBECOM*, pp. 3443–3448, Dec 2012.

[79] E. Oh and B. Krishnamachari, "Energy savings through dynamic base station switching in cellular wireless access networks," in *Proceedings of the IEEE GLOBECOM 2010*, pp. 1–5.

[80] M. Marsan, L. Chiaraviglio, D. Ciullo, and M. Meo, "Optimal energy savings in cellular access networks," in *Communications Workshops, 2009. ICC Workshops 2009. IEEE International Conference on*, pp. 1–5, June 2009.

[81] M. A. Marson, L. Chiaraviglio, D. Ciullo, and M. Meo, "Energy-aware UMTS access networks," *Proc. International Workshop on Green Wireless (WGreen 2008)*, September 2008.

[82] D. Tipper, A. Rezgui, P. Krishnamurthy, and P. Pacharintanaku, "Dimming cellular networks," in *IEEE GLOBECOM'10*, December 2010.

[83] M. Ilyas, G. Baig, M. Qureshi, Q. Nadeem, A. Raza, M. Qazi, and B. Rassool, "Low-carb: Reducing energy consumption in operational cellular networks," in *IEEE GLOBECOM 2013*, pp. 2568–2573, Dec 2013.

[84] Nokia Siemens Networks, "Flexi BSC (BSC14)." `http://bit.ly/QMD8O5`, June 2011.

[85] Z. Hasan, H. Boostanimehr, and V. K. Bhargava, "Green cellular networks: A survey, some research issues and challenges," *Communications Surveys & Tutorials, IEEE*, vol. 13, no. 4, pp. 524–540, 2011.

[86] H. Ertl, J. W. Kolar, and F. C. Zach, "Analysis of a multilevel multicell switch-mode power amplifier employing the" flying-battery" concept," *Industrial Electronics, IEEE Transactions on*, vol. 49, no. 4, pp. 816–823, 2002.

[87] E. McCune, "High-efficiency, multi-mode, multi-band terminal power amplifiers," *Microwave Magazine, IEEE*, vol. 6, no. 1, pp. 44–55, 2005.

[88] C. Yoo and Q. Huang, "A common-gate switched 0.9-w class-e power amplifier with 41% pae in 0.25-μm cmos," *Solid-State Circuits, IEEE Journal of*, vol. 36, no. 5, pp. 823–830, 2001.

[89] A. Grebennikov, N. O. Sokal, and M. J. Franco, *Switchmode RF and microwave power amplifiers*. Academic Press, 2012.

[90] A. Amanna, "Green communications: Annotated review and research vision," *Virginia Tech*, p. 87, 2010.

[91] H. Claussen, L. T. Ho, and F. Pivit, "Effects of joint macrocell and residential picocell deployment on the network energy efficiency," in *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pp. 1–6, IEEE, 2008.

[92] M. A. Marsan and M. Meo, "Energy efficient wireless internet access with cooperative cellular networks," *Computer Networks*, vol. 55, no. 2, pp. 386–398, 2011.

[93] K. Samdanis, D. Kutscher, and M. Brunner, "Self-organized energy efficient cellular networks," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2010 IEEE 21st International Symposium on*, pp. 1665–1670, IEEE, 2010.

[94] K. Samdanis, D. Kutscher, and M. Brunner, "Dynamic energy-aware network reconfiguration for cellular urban infrastructures," in *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*, pp. 1448–1452, IEEE, 2010.

[95] R. Jeyarani, N. Nagaveni, and R. Vasanth Ram, "Design and implementation of adaptive power-aware virtual machine provisioner (APA-VMP) using swarm intelligence," *Future Gener. Comput. Syst.*, vol. 28, pp. 811–821, May 2012.

[96] H. Al-Daoud, I. Al-Azzoni, and D. G. Down, "Power-aware linear programming based scheduling for heterogeneous computer clusters," *Future Generation Computer Systems*, vol. 28, no. 5, pp. 745 – 754, 2012.

[97] M. Liedloff, "Finding a dominating set on bipartite graphs," *Inf. Process. Lett.*, vol. 107, pp. 154–157, Aug. 2008.

[98] "Forsk Atoll: Wireless Network Engineering Software." http://www.forsk.com/atoll/.

[99] M. Gerla and J. T.-C. Tsai, "Multicluster, mobile, multimedia radio network," *Wireless Networks*, vol. 1, pp. 255–265, Aug. 1995.

[100] Ericsson, "Vodafone Germany first to launch Ericsson's power-saving feature to reduce energy consumption and cut CO_2 emissions." `http://bit.ly/Sls1v1`, December 2007.

[101] Mobilink, "Mobilink Network." `http://bit.ly/UWXYzB`, September 2012.

[102] Pakistan Telecommunication Authority, "Pakistan Telecommunication Authority, 2011 Annual Report." http://bit.ly/S8YtWS, December 2011.

[103] Wikipedia, "Telenor Pakistan - Wikipedia." http://bit.ly/R1Uyqm, September 2012.