

# **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

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Computer networks pervade . . . . .	1
1.2	Electricity costs for operational networks . . . . .	3
1.3	The impact of energy inefficiency on operational networks' electricity costs .	4
1.4	Prevalent electricity cost reduction techniques . . . . .	6
1.4.1	Reducing the amount of energy consumed . . . . .	7
1.4.2	Using cheaper electricity - Workload Relocation (WR) . . . . .	8
1.5	Our thesis . . . . .	9
1.6	Contributions . . . . .	9
1.7	Organization . . . . .	10
<b>2</b>	<b>Background - Power consumption models in geo-diverse data centers and cellular networks</b>	<b>11</b>
2.1	Geo-Diverse Data Centers - enablers of cloud computing . . . . .	11
2.1.1	Structure of a data center . . . . .	13
2.1.2	Inter-data center traffic . . . . .	15
2.1.3	Handling of client requests . . . . .	17
2.1.4	Data center power consumption model . . . . .	19
2.2	Cellular Networks . . . . .	20

2.2.1	Structure of a cellular network . . . . .	21
2.2.2	Call placement . . . . .	23
2.2.3	BTS power consumption model . . . . .	24
2.3	A comparison of data center and cellular networks . . . . .	25
2.4	Discussion . . . . .	26
<b>3</b>	<b>A generalized framework for electricity cost optimization</b>	<b>27</b>
3.1	Problem Model . . . . .	27
3.2	Optimization problem formulation . . . . .	31
3.2.1	The objective function . . . . .	32
3.2.2	The constraints . . . . .	32
3.2.3	Comments on the problem formulation . . . . .	33
<b>4</b>	<b>Case Study I: Geo-diverse Data Centers</b>	<b>35</b>
4.1	Prelude . . . . .	35
4.2	Related work . . . . .	37
4.3	Sources of transition costs in the data center scenario . . . . .	40
4.4	Instantiating the generalized optimization formulation . . . . .	41
4.5	Experimental setup . . . . .	50
4.5.1	Application workload . . . . .	50
4.5.2	Electricity prices . . . . .	51
4.5.3	Algorithms for Workload Distribution/Relocation . . . . .	51
4.6	Results . . . . .	55
4.6.1	Sensitivity of electricity cost savings to extent of overprovisioning . .	56
4.6.2	Sensitivity of electricity cost savings to magnitude of transition costs	57
4.6.3	Sensitivity of electricity cost savings to resource pruning granularity .	58
4.6.4	Sliding window re-optimization . . . . .	60

4.7	Sensitivity of electricity cost savings to the server idle-peak power ratio . . .	65
4.8	Performance of the heuristic algorithm . . . . .	68
4.9	Discussion . . . . .	69
<b>5</b>	<b>Case Study II: Cellular Networks</b>	<b>72</b>
5.1	Prelude . . . . .	72
5.2	Related work . . . . .	75
5.3	Instantiating the generalized optimization formulation . . . . .	77
5.3.1	Single Base Transceiver Station (BTS) . . . . .	77
5.3.2	Multi-BTS Cellular Setting . . . . .	78
5.4	Experimental setup . . . . .	80
5.4.1	Site Characteristics . . . . .	83
5.5	Results . . . . .	85
5.5.1	Sensitivity of electricity cost savings to the duration of an optimization interval . . . . .	86
5.5.2	Comparison of our heuristic with the optimal solution . . . . .	88
5.5.3	Sensitivity of electricity cost savings to the resource pruning granularity	89
5.5.4	Sensitivity of electricity cost savings to the margin of state-change damping . . . . .	92
5.6	Discussion . . . . .	92
<b>6</b>	<b>Conclusions and Future Work</b>	<b>94</b>
6.1	Contributions . . . . .	94
6.2	Limitations . . . . .	96
6.3	Future work . . . . .	96

# List of Figures

1.1	Networks lack energy proportionality . . . . .	5
1.2	Call traffic for an operational cellular site over two days . . . . .	6
2.1	Google Data Center Locations - Source: <a href="http://www.google.com/about/datacenters/inside/locations/">www.google.com/about/datacenters/inside/locations/</a>	
2.2	The modern data center's architecture . . . . .	15
2.3	Resolving the IP address for a server hosted in a data center . . . . .	18
3.1	An example of mapping variable workload to capacity-limited network resources 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 resources are situated. (c) A uniform mapping of workload to network resources does not exploit electricity price diversity. (d) Mapping workload to network resources 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. . . . .	29
4.1	Normalized workload . . . . .	50
4.2	Workload intensity histogram . . . . .	50



4.3	Percentage savings with over-provisioning . . . . .	55
4.4	Total cost vs transition overhead . . . . .	58
4.5	What . . . . .	62
4.6	Mean absolute workload prediction error vs sliding window size . . . . .	63
4.7	Local trajectory correction technique for three consecutive intervals . . . . .	63
4.8	Distribution of workload prediction error for sliding window size of 12 hours	63
4.9	Percentage error of sliding window forecasts compared to global optimal with error-free workload . . . . .	65
4.10	Cost saving vs (de)activation granularity . . . . .	66
4.11	Average daily total electricity cost and it's components vs $f$ , For $bs = 0.01$ .	67
4.12	Average daily total electricity cost and it's components vs $f$ , For $bs = 0.65$ .	68
4.13	The minimum, maximum and average percentage difference between the cost of our heuristic and RED-BL . . . . .	69
5.1	Location of the BTS sites in our dataset (spread over a $31.25 \text{ km}^2$ urban terrain)	73
5.2	Empirical CDF of the number of potential serving BTSs for a call in our dataset (large metropolitan area) . . . . .	74
5.3	Normalized traffic for a 24-hour period at two neighboring sites in our dataset shows the geo-diversity in traffic . . . . .	74
5.4	BTS power consumption model. Low-power (BTS power savings) mode is optional and kicks in at low loads. . . . .	78
5.5	Percentage reduction in energy consumption vs inter-optimization interval .	87
5.6	Percentage reduction in energy consumption vs inter-optimization interval .	87
5.7	Empirical CDF of the difference between the cost offered by our heuristic compared to the optimal . . . . .	88

5.8	The percentage energy savings for all three BTS models considered in this paper vs the value of $\epsilon$ , with a six minute inter-optimization interval . . . . .	93
-----	--	----

# List of Tables

2.1	A comparison of data center and cellular networks . . . . .	25
4.1	Data Center Network Model Parameters . . . . .	44
4.2	Sources of electricity prices used in our work . . . . .	51
4.3	Algorithms compared in our work . . . . .	52
4.4	A comparison of the algorithms studied in this paper . . . . .	52
5.1	BTS model parameter values . . . . .	85
5.2	Energy savings by using BTS power savings only . . . . .	86
5.3	Percentage electricity savings for different granularity of resource pruning . .	91

## **Abstract**

Abstraction

# Chapter 1

## Introduction

### 1.1 Computer networks pervade

Services like 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 Internet Protocol (IP) 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 know as service providers or network operators. Examples of such network operators are:

- **Cellular Network Operator:** Cellular network operators deploy and run a cellular network infrastructure. Cellular networks can also be characterized as computer networks as both the end users' devices (cell phones) and most of the devices in the infrastructure are computers. The most common services offered by these operators are voice calls and short text messages.

A few years ago, the predominant means of connecting to the Internet was dialup over

Public Switched Telephone Network (PSTN). These days, however, many end users use cellular networks to access Internet resources. In this context, the cellular network merely acts as an access mechanism to connect to an Internet Service Provider's network.

- **Internet Service Providers (ISPs):** The Internet itself is an interconnection of ISPs networks. End users connect to the Internet by connecting to an ISP's network. This is typically done by a paid subscription to use the ISP's network.

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 by the networks run by other service providers such as the ones given below.

- **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 like Facebook, Amazon, Microsoft and Google operate geo-diverse data centers. Servers in these data center networks run applications like 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 keep the latency from a user to an Internet resource small. 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<sup>1</sup>.

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 electricity bill. Optimizing operational costs is critical for network operators in order to offer cost-effective services to consumers and maximize 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]. Similarly, for an operator with 7000 cellular sites in a country as small as Pakistan, the annual electricity cost can be roughly estimated at \$9.19 Million<sup>2</sup>. Telecom Italia reported a consumption of 1.793 GWh in 2012 [6], which is significantly higher compared to our estimates for the Pakistani network operator and hence the electricity costs are also expected to be much higher.

---

<sup>1</sup>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

<sup>2</sup>Using a 1.5 kW draw for a single cellular site [5], 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 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.

### 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 the dotted line in Figure 1.1. The network should consume no power when there is no workload and in the presence of workload, it's 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 [7, 8], as shown by the solid line in Figure 1.1. We say 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 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 prospective callers, 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, it's power consumption under no load is quite high.
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 and the server idle power consumption is a large fraction of their peak power consumption.



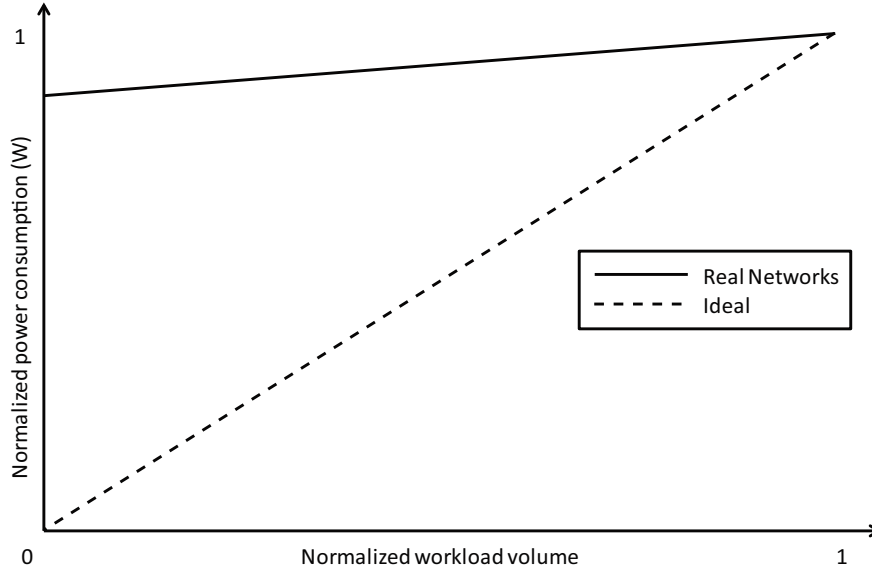


Figure 1.1: Networks lack energy proportionality

Lack of energy proportionality is of little concern if the network consistently operates in the high workload region because in this region, the 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 [9] and data center [10] traffic also show similar trends.

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 networks have a much higher energy consumption than what they would ideally incur. In other words, today's networks are energy inefficient. Another way to look at energy 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

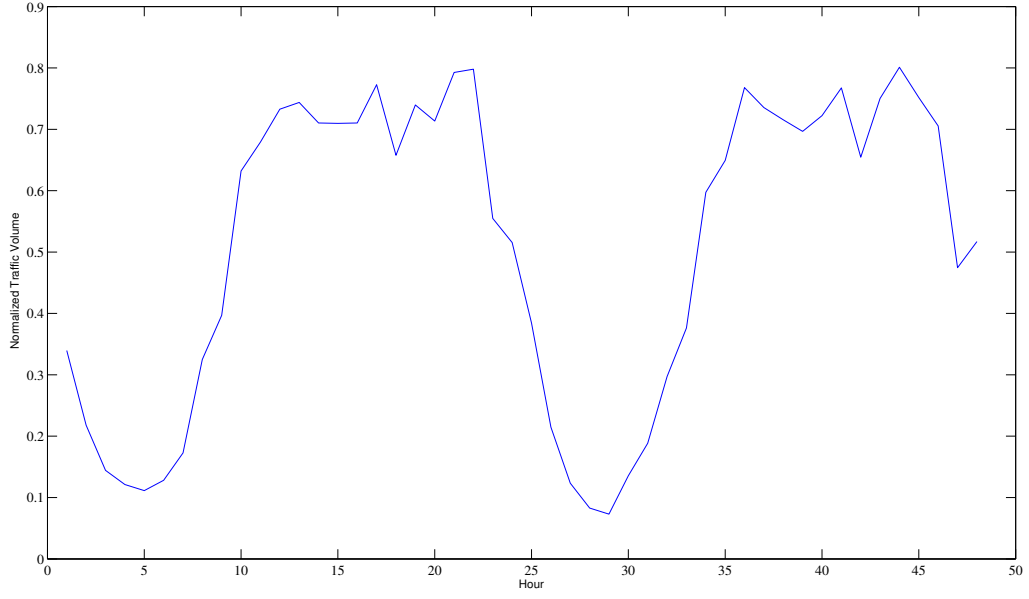


Figure 1.2: Call traffic for an operational cellular site over two days

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:

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

Current 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 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, thereby reducing electricity cost. Therefore, hardware upgrades are 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 the end of life of deployed equipment. 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. 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% [11].
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

resource may be deactivated (shutdown or put in power-saving mode depending on what is supported by the equipment) when the workload is low [12, 13, 14, 15, 7]. 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.

### 1.4.2 Using cheaper electricity - Workload Relocation (WR)

Electricity prices exhibit geographic diversity [16], i.e., the price of electricity varies from one location to another. 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<sup>3</sup>. Most networks span large enough distances for geographic diversity in electricity prices to be apparent.

If the network workload is quite flexible in terms of where it is handled, then the workload originating at a location with high electricity price may be relocated to a different location that has lower electricity price, thereby cutting the cost of electricity. We call this technique Workload Relocation (WR).

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.

Electricity prices also exhibit temporal diversity [16], i.e., the relative order of electricity prices at different locations keeps changing. If a city in Kansas presently has cheaper electricity than one in Oklahoma, an hour later, the reverse may be true. This means that mapping

---

<sup>3</sup>With the exception of factors such as different tariffs for domestic, commercial and industrial consumers

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 Our thesis

Based on the similarity in workload characteristics and the dependence of power consumption on workload, we opine that a generalized power optimization framework 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 framework would use workload relocation and resource pruning in tandem to reduce electricity costs<sup>4</sup>.

## 1.6 Contributions

This thesis makes the following contributions:

- We present a generalized model for electricity cost optimization applicable to different types of networks that jointly uses workload relocation and resource pruning. We show that this problem is NP-Hard.
- We present a framework called Relocate Energy Demand to Better Locations (RED-BL), pronounced Red Bull, that solves the electricity cost minimization problem. We apply RED-BL to geo-diverse data centers as well as cellular networks using real data traces.

---

<sup>4</sup>Hardware virtualization is complimentary to our framework

- We obtain optimal solutions to RED-BL problems for both data center and cellular networks for realistic problem sizes. To handle the intractability of the problem, we also propose some heuristics that would be useful for even larger instances of the electricity cost minimization problem.
- 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 framework.
- 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.

## 1.7 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 geo-diverse data centers and cellular 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 Geo-Diverse Data Centers - enablers of cloud computing

Traditionally, an individual or organizations that requires computing resources would procure it. Authors needing to do electronically typeset manuscripts from home 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 factors such as renewal of software licenses, salaries for staff to smoothly run the software and training to customize and operate the newly acquired software. On the other hand, we use utilities such as electricity, water, telephone and gas without having to deploy and run facilities power generation plants or water storage systems. We just pay a per-use subscription charge to some organization that does it for us, while we just consume the utility as needed. Due to economies of scale, utilities are economical for 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.

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 Operating System 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 on the end user. 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 Operating System (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.

- **Software as a Service (SaaS):** A service user often wishes to be least concerned with software installation, configuration and maintenance. 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 given 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. This is done for two reasons. First, having data centers at different locations provides fault tolerance. If one site goes down for some reason, the other site may take over as a backup. Also, multiple remote sites are less likely to be affected simultaneously by a natural disaster. A second reason to have multiple data centers is to have low latency to clients at different locations. For instance, Amazon has multiple data centers in different continents, thereby ascertaining that no matter where a client may be, there is an Amazon data center relatively close by compared to the case if Amazon only had one data center in the US. Figure 2.1 shows the locations of Google's data centers across the globe (according to [www.google.com/about/datacenters/inside/locations/](http://www.google.com/about/datacenters/inside/locations/) as of August 10, 2014).

### 2.1.1 Structure of a data center

Today's data centers are organized hierarchically [17], as shown in Figure 2.2. A typical data center hosts tens of thousands of servers [18]. 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, 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



Figure 2.1: Google Data Center Locations - Source: [www.google.com/about/datacenters/inside/locations/](http://www.google.com/about/datacenters/inside/locations/)

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 each other. 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 avenues for traffic coming in and going out of the data center.

All of the equipment is quite tightly packed within a pretty small space 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.

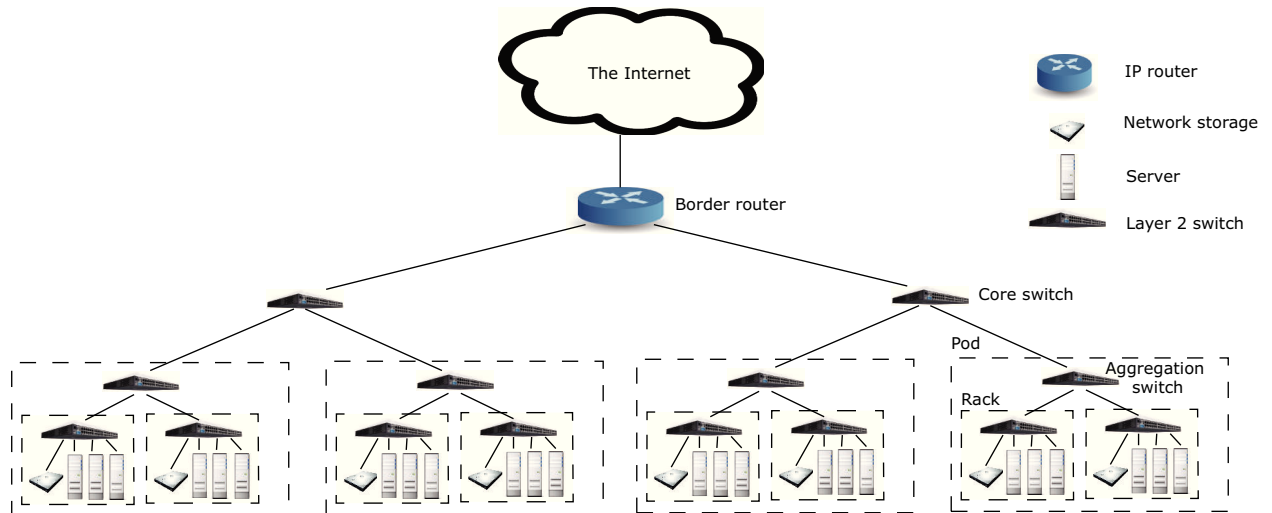


Figure 2.2: The modern data center's architecture

### 2.1.2 Inter-data center traffic

An operator typically deploys multiple data centers that are geographically distributed. Each of these data centers is inter-connected by means of high-speed inter data center network 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 replica does not actively handle client requests until the primary server fails. When such an event occurs, the 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.

Unless the content being hosted by a server is static, which is rarely the case these days, replication incurs an overhead. 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 not static (growing when new emails are received and shrinking when emails are deleted) and these inboxes may grow and shrink at different rates, depending on factors such as number of mailing lists that a user subscribes, 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 communicating requests, intermediate results and final responses. For instance, a web search provider may distributed 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 must be aggregated after being received on the same links.

### 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<sup>1</sup>. Since a single server would hardly be sufficient to handle all traffic for a popular web site, several servers must be 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. For details on DNS resolution process, see [19, 20].

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 the URL. The local DNS resolver communicates with the DNS server for the client's ISP<sup>2</sup>. 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 list of IP addresses, it picks one of the IP addresses to fetch the web page. For simplicity, many web

---

<sup>1</sup>It is also possible to specify the IP address of the web server directly in the URL.

<sup>2</sup>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.

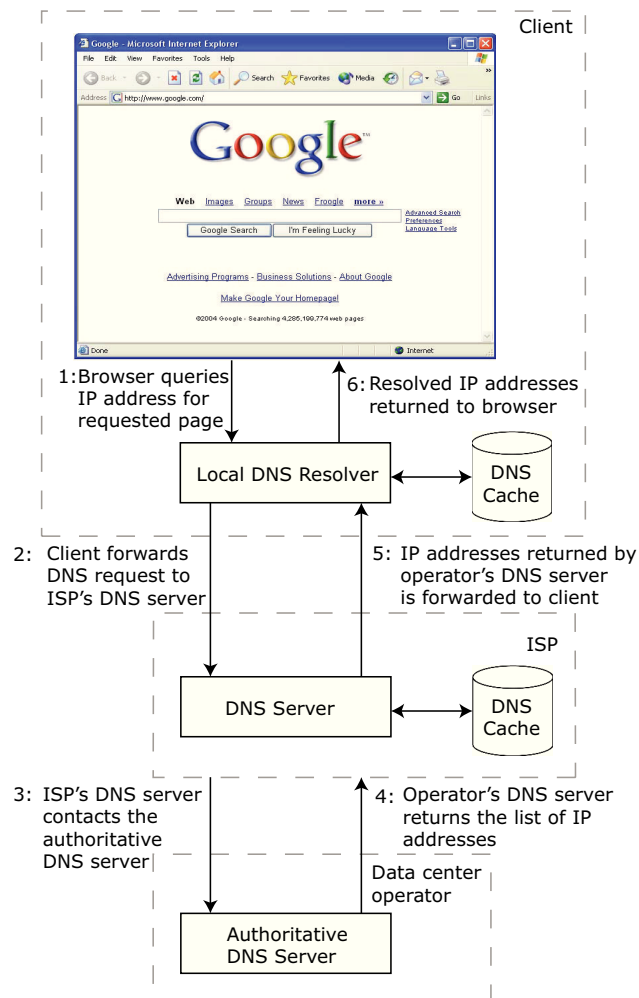


Figure 2.3: Resolving the IP address for a server hosted in a data center

browsers simply pick the first IP address in the list. If, for some reason, 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 simply picks the first IP address, 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) [21] 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 ISP until they reach the data center where the required web server is hosted. The border router in the data center 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 response packets are forwarded from the web server back to the border router which routes it back to the client machine.

#### **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. used the results of a measurement study to show that the total power consumption in a data center can be well-modeled as an affine function of the average CPU utilization [8]. As more and more client traffic arrives at servers in a data center, the average CPU utilization increases. Therefore, data center power consumption can also be represented as an affine function of workload.

## 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. Literature often categorizes mobile network technologies in terms of *generations*. 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 (a macrocell) 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)<sup>3</sup> 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). Multiple frequency channels are available in each sector - one for each TRX - and each frequency is divided into eight time slots. A particular frequency and a particular time slot identify a physical channel. A few physical channels in each sector are reserved for control signaling purposes and the rest can be used for user traffic such as voice

---

<sup>3</sup>A single cell site sometimes hosts multiple BTSs, for instance, when multiple network operators share a single site

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.

The duration of each time slot for a given frequency is 0.577 ms. A group of eight time slots for a given frequency is collectively referred to as a GSM frame. The transmission in each time slot for a given frequency is termed as a burst. The recurrence of a particular burst across all GSM frames is a physical channel. A group of 26 GSM frames is referred to as a GSM multi-frame.

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. This band 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, 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<sup>4</sup>.

Frequency assignment and frequency hopping are examples of activities that require co-

---

<sup>4</sup>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.

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 links or microwave. Since a BSC only controls a subset of BTSs, it can only perform local coordination. In order to ensure global coordination in radio resource allocation and control, the BSCs are also interconnected using backhaul 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 handoff is coordinated by a Base Station Controller (BSC).

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% [22] - 80% [23] 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 send 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 numbers. 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 [22] claimed that BTSs contribute 50% of overall network power consumption, whereas Oh et. al [23] 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-

	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	Limited
Geo-flexibility in resource selected for workload handling	High	Low

Table 2.1: A comparison of data center and cellular networks

traffic conditions. They concluded that the BTS power consumption may be approximated as an affine function of call traffic [24]. 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 data centers, whereas it is voice calls in case of cellular networks. The workload in both types of networks exhibits diurnal patterns [10, 7]. 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.

A subtle difference between geo-diverse data centers and cellular networks is that the former is characterized by geo-temporal diversity in electricity prices. Furthermore, any client request may be handled at any data center, irrespective of their location. Thus, for a given client request at a given time, the candidate data centers are situated at locations with different electricity prices. In case of cellular networks, however, a given call may be handled at only a few BTSs in the vicinity, which are so close that an electricity price differential is unlikely.

## 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 these networks in terms of their power consumption. In the next chapter, we will use these abstractions to derive such a generic power consumption model and thus, formulate an electricity cost minimization framework.

# Chapter 3

## A generalized framework 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 amount of workload handled. 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 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 resource and marbles to represent a unit workload. The network resource 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 workload (and electricity price) 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 resources 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 resources. 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 resources.

One possible operational strategy is to map each workload unit to the nearest available resource 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<sup>1</sup>. In geo-diverse data center settings, this mapping strategy is also often the default strategy because it minimizes the access latency for all clients<sup>2</sup>.

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

---

<sup>1</sup>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

<sup>2</sup>Network latency has been shown to have a strong correlation with the physical shortest path distance between two locations on the globe [25]. So, the commonly understood physical measure of "shortest" applies in this case.



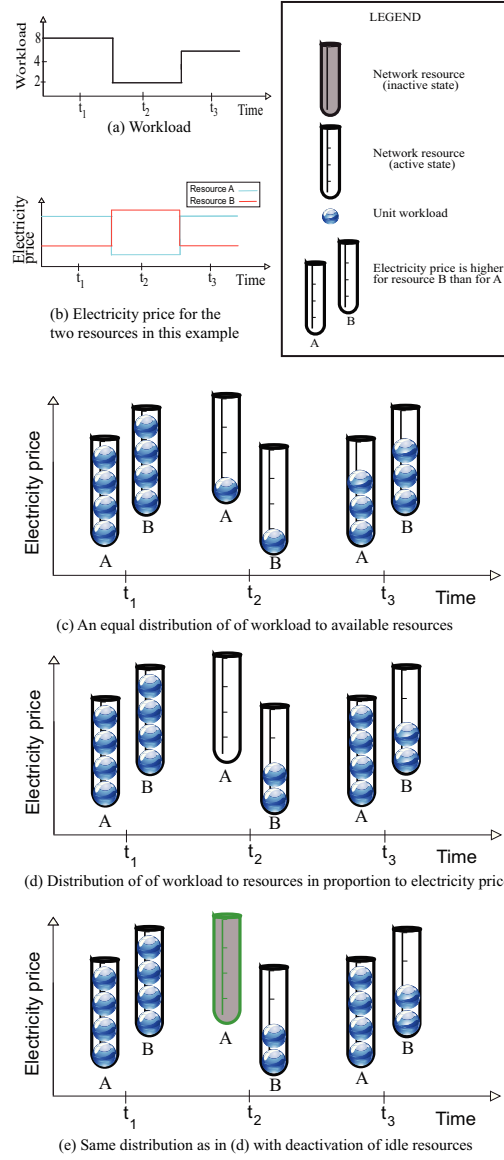


Figure 3.1: An example of mapping variable workload to capacity-limited network resources with geo-temporal diversity in electricity prices. Three consecutive intervals  $t_1$ ,  $t_2$  and  $t_3$  are considered. Workload and electricity prices may only change between two consecutive intervals. (a)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

electricity prices. We can exploit geo-diversity in electricity prices to reduce the electricity cost over the planning window by mapping more workload to resources at cheaper locations. 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-resource 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 cheaper locations. In interval  $t_1$ , since the cumulative workload equals the total network capacity, both network resources 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 resource 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 resource A is loaded to full capacity and the remaining workload is mapped to the other resource.

Due to lack of energy proportionality in networks, the power consumption of idle resources 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-resource 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 resource continue to consume 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, if a data center is running at 50% of its capacity in a given interval, half of it might be deactivated. In our example of Figure 3.1 (e), in intervals 2 and 3, half of resource B might be deactivated.

## 3.2 Optimization problem formulation

On a high-level, routine network operation (in the context of our thesis) involves distributing workload to network resources and periodically updating the fraction of workload mapped to each network resource. For simplicity of modeling and analysis, we assume that the mapping of workload to network resources, henceforth referred to as *workload mapping* or simply *mapping*, is updated at the beginning of intervals of fixed duration. We define resource  $i$ 's state during interval  $j$ , denoted  $s_i^j$ , as the corresponding resource's status (on or off) and the amount of workload mapped to it. The aggregated state of all network resources 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 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:

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

Here  $C(S^j)$  represents a function that evaluates the cost of being in state  $S^j$  and  $T(S^j, S^{j-1})$  represents the cost of transitioning from state  $S^{j-1}$  to state  $S^j$ . In the context of our thesis, the function  $C(S^j)$  should evaluate the electricity cost of network operation given that the network is in state  $S^j$ . Similarly,  $T(S^j, S^{j-1})$  should compute the cost of changing network state between two consecutive intervals that may arise due to factors such as turning network resources on or off.

### 3.2.2 The constraints

The state trajectory problem must be subject to a number of problem-specific constraints. Some constraints are common to all types of networks.

- **Resource capacity must be respected:** During all intervals, we must ensure that the workload mapped to each network resource does not exceed it's capacity.
- **All workload must be handled:** During all intervals, the sum of workload mapped to all network resources must equal the offered workload for that interval.
- **Network resource status is binary:** The status of a network must be represented as a binary variable which takes on the value 1 if the said network resource is on

during a given interval, and 0 otherwise. We can not simply represent the resource state using a real-valued variable, representing the fraction of the current cumulative workload mapped to the corresponding resource, because it is not possible to determine instances of activation/deactivation of resources using such real-valued variables.

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.

### 3.2.3 Comments on the problem formulation

The decision variables in our problem formulation are the state of network resources for each interval in the planning window. A resource's state has two parts: the amount of workload it handles and its status (on or off). The resource status needs to be a discrete (binary) 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 resource represents the number of active calls being handled by a BTS. In some other cases, on the other hand, such as the geo-diverse data center scenario, the amount of workload mapped to a resource may be represented using a real-valued variable, representing the fraction of total workload being handled by a data center during a given interval. In the first scenario, such as cellular networks, the resource state is purely discrete, whereas in the second scenario, such as geo-diverse data centers, the resource 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, however, and must be solved using techniques such as branch and bound [26] or other heuristics. If functions  $C(\cdot)$  and  $T(\cdot, \cdot)$  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), having linear objective functions results in lower 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.

In this chapter, we have presented only an abstract formulation of the general optimization problem for minimizing electricity cost in service provider networks. We find it appropriate to comment here on the more concrete mathematical formulation of the problem in specific network types as well as the techniques used to solve those concrete instances of the problem. 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, which uses the branch and bound heuristic, available with the ILOG CPLEX Studio which is available free of cost for academic use. 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* rather than proposing heuristics for approximate solutions to the problem<sup>3</sup>. As we shall see in the next two chapters, we were able to solve problems of reasonable size using simple desktop PCs within a reasonable amount of time.

---

<sup>3</sup>We do, however, propose a heuristic for the optimization problem in a cellular network setting.

# Chapter 4

## Case Study I: Geo-diverse Data Centers

### 4.1 Prelude

To enable robust and low-latency cloud services, companies such as Amazon, Google and Microsoft deploy large-scale infrastructure in the form of distributed data centers. The cost of electricity needed to run these data centers accounts for a significant portion (15%) of the total capital and operational expenditure [4]. Furthermore, due to increasing service demand as well as electricity prices, the fraction representing the cost of electricity is on the rise [27, 28], making it important for the cloud providers and data center operators to cut down on their electric bills.

In the previous chapter we have presented an abstract and generalized formulation of an electricity cost optimization framework for various types of networks. In this chapter, we will derive a specialized instance of the general problem, discuss its solution (exact as well as heuristic) and perform a sensitivity analysis. We will also discuss our formulation's benefits and limitations.

Our problem formulation talks about cutting electricity cost by shutting down resources, data centers in this case, during low-workload regimes to improve energy efficiency of the overall network. However, each data center is a very large distributed system. Several possible ramifications of shutting down entire data centers raises some concerns. First, when a data center is booted up, some equipment requires ramp-up time before the data center can start regular operation. If the ramp-up time is of the order of the time at which electricity prices change, our solution will not be able to fully exploit the geo-temporal variation in electricity prices. An example of such equipment is air-conditioning. Another concern is that a restart of networking equipment such as border routers often results in convergence delays before the forwarding information across the network is consistent. During this time, the forwarding information may have undesirable states such as forwarding loops. For this reason, it may not be wise to shutdown networking equipment even when the data center is not expected to handle any workload. Third, it may be important to keep application data stores (such as search indices) consistent, hence the database servers on all data centers need to be up and running irrespective of presence or absence of workload. Fourth, some clients might keep cached location information about server's addresses and may continue to send traffic to a data center that an operator has shutdown. Dropping such *unexpected traffic* does not make good business sense and some servers may need to be kept active to handle it. It appears from this discussion that a data center may only be partially shutdown, if needed, so that performance and revenues are not impacted.

The electricity load at a data center may be partitioned into two sets: one that is inelastic and must be on all the time and the other elastic portion that may be shutdown if there is no workload for the data center to handle. Let the first set comprise a fraction  $r$  of the total data center power draw. The second part accordingly contributes a fraction  $(1 - r)$  of the total power draw. According to [29], the breakup of power draw in a typical data center is servers: 56%, cooling: 30%, power conditioning: 8%, network: 5% and lighting: 1%. Since



we put only the servers in the elastic pool of devices,  $r = 0.44$ .

The inelastic electricity load in a data center results in a fixed power consumption which can not be minimized by a workload relocation scheme. Hence, while formulating the electricity cost minimization problem, we must focus solely on the elastic electricity load in the data center, i.e., servers. To model server power consumption, we observe that it is known to be an affine function of CPU utilization [30]. Another way to arrive at this model is to note that data center power consumption is an affine function of average CPU utilization [8] and server power consumption is a known fraction of a data center’s total power consumption.

## 4.2 Related work

While much prior work has considered scaling the capacity within a data center by scaling the number of servers that are on [31, 32, 33, 34, 35], to the best of our knowledge the cost of shutting down entire data center(s) has largely been ignored [36]. In the present work, we take a macro-scale view of an entire data center, whereby we use decision variables representing a whole data center’s state. Micro-scale approaches that attempt to manage per-server state across all distributed data centers tend to suffer from scalability problems and thus rely mostly on approximation algorithms. Our formulation is more scalable because of the macro-scale view.<sup>1</sup>

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 [31]. 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

---

<sup>1</sup>The trade-off is that this macro-level focus might produce somewhat sub-optimal results. Investigation of the magnitude of the sub-optimality is left as future work.

individual data center or a fraction thereof. This limitation is evident from the small number of servers used in the simulation-based evaluation in [31]. 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. They used the Soccer World Cup 1998 webserver workload traces and electricity prices at four different locations to evaluate their proposal.

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 [37]. Buchbinder et. al. proposed online algorithms for relocating MapReduce jobs between geo-diverse data centers to reduce the electricity bill while considering the cost of inter-data center bandwidth [38]. 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 MapReduce 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 [39]. 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 [32]. 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 [40]. Their proposed scheme recharges the UPS units when the electricity prices are lower.

Investigation of strategies of infrastructure scaling to conserve power in a single data center is reported in [33, 41, 42, 12]. 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 [33]. 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. present similar strategies in [41]. Oh et. al. considered a virtualized environment and proposed solutions for optimally placing VMs on servers and map workload to the VMs such that electricity costs are minimized [42]. In [12], Chase et. al. present policies for resource allocation in a hosting center along with 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 [13], 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 [43]. The authors used workload data from Google search cluster and evaluated a scenario of an operator with data centers at two different locations.

When one talks about cutting data center electricity costs without cutting the total electricity consumption, questions of the sustainability also arise. Efforts towards *green* mapping of workload to data centers have been reported in [44, 45, 46, 47, 48, 49, 50, 51]. For minimizing total electricity consumption on participant hosts in a peer-to-peer file download system, Sucevic et. al. studied various approaches such that peers could be turned off when they are not needed [52].

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) [53], electric power systems and smart grid [54, 55, 56, 57], WiFi access points [58], wide area networks [59], cellular networks [7] and high performance computing [60, 61, 62, 63, 64, 65].

### 4.3 Sources of transition costs in the data center scenario

Our present work 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 [66]. 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 [67]. 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 [36], in the present work, 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 paper 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 Instantiating the generalized optimization formulation

Consider a geo-distributed data center infrastructure comprising  $m$  interconnected data centers. The workload consists of client requests for hosted applications. Every client request is routed to one of the data centers in the network. The fraction of requests mapped to each data center may be revised periodically. For ease of modeling, we assume that these changes may be done at the start of discrete intervals of duration  $\lambda$ . Let the workload handled at

data center  $i$  during interval  $j$  be  $x_i^j$ . We consider a normalized measure of workload, so that  $\sum_{i=1}^m x_i^j \leq 1$ .

Let  $c_i$  be the workload capacity for data center  $i$ . Also, let  $P^{max}$  and  $P^{min}$  be the sum of the peak and idle power consumption, respectively, of the elastic portion of electric load over all data centers. In the present work, we consider homogenous data centers, so an individual data center's peak and idle server power consumption are proportional to its workload capacity, i.e.,  $P_i^{max} = c_i P^{max}$  and  $P_i^{min} = c_i P^{min}$ . Since the average utilization of data center  $i$  during interval  $j$  is  $x_i^j/c_i^j$ , its power consumption during that interval may be given by:

$$P_i^j = c_i(P^{min} + \frac{x_i^j(P^{max} - P^{min})}{c_i}) \quad (4.1)$$

Let the ratio of  $P^{min}$  to  $P^{max}$  be  $f$ . The case when  $f = 0$  corresponds to complete energy-proportionality in the elastic load, which is purely ideal and does not occur in reality, whereas  $f = 1$  means that the elastic load's power consumption is completely independent of workload. Substituting  $P^{min} = fP^{max}$  in the above expression gives:

$$P_i^j = c_i P^{max} (f + \frac{x_i^j(1-f)}{c_i}) \quad (4.2)$$

Since  $P^{max}$  is a constant parameter, it plays no role in electricity cost minimization. Hence, we may normalize  $P_i^j$  over  $P^{max}$  to get:

$$\hat{P}_i^j = f c_i + x_i^j(1-f) \quad (4.3)$$

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 4.3 goes to zero and the power consumption reduces to the first term in equation 4.3 only, which we refer to as *idle power consumption*. The second term in

equation 4.3 indicates the workload-dependent *computational power consumption*, which is independent of the data center capacity.

Electricity cost incurred at data center  $i$  during interval  $j$  is a product of its power consumption ( $P_i^j$ ), duration of the interval ( $\lambda$ ) and the unit price of electricity ( $e_i^j$ ). Hence, the sum of costs for idling and computational energy consumption incurred at data center  $i$  during interval  $j$  may be given as:

$$e_i^j \lambda (c_i f + (1 - f) x_i^j) \quad (4.4)$$

Now, let us consider the activation/deactivation costs for the elastic portion of the data center workload. Let  $\sigma$  and  $\delta$  be the average power consumption, over a single interval, required to activate or deactivate, respectively, the elastic load at a unit capacity data center. Then, the bootup power consumption for the elastic load at data center  $i$  is  $\sigma c_i$  and the shutdown power consumption is  $\delta c_i$ .

The electricity cost for booting up data center  $i$ 's elastic load at the start of interval  $j$  is given by  $c_i e_i^j \sigma$ . Multiplication with the duration of an interval, i.e.,  $\lambda$  is not necessary, because  $\sigma$  is defined as the per interval cost. We are assuming that the elastic load at a data center may be booted up within a single interval. The value of  $\lambda$  that we used in our experiments is equal to one hour, which seems to be a sufficiently large interval for this boot up. Similarly, the electricity cost for shutting down data center  $i$ 's elastic load at the start of interval  $j$  is given by  $c_i e_i^j \delta$ .

The total cost of electricity for data center  $i$  during interval  $j$  is the sum of computational, idling, activation and deactivation electricity costs incurred during that interval. Therefore, the RED-BL optimization problem formulation, a summation of electricity costs over all data centers and over all intervals in a planning window may be given as:

Parameter	Description
$m$	Number of data centers
$r$	Fraction of data center load that is inelastic, i.e., may never be shutdown
$n$	Number of intervals in a planning window
$f$	The ratio between a data center's peak and idle power consumption
$c_i$	Normalized workload capacity of data center $i$
$\sigma$	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
$\delta$	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 $j$
$\lambda$	Duration of an interval in hours
$w^j$	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

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

subject to:

$$x_i^j \leq c_i \quad \forall i, \forall j \quad (4.6)$$

$$\sum_{i=1}^m x_i^j = w^j \quad \forall j \quad (4.7)$$

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

$$p_i^j \geq x_i^j \quad \forall i, \forall j \quad (4.9)$$

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

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



$$b_i^0 = p_i^0, s_i^0 = 0 \quad \forall i \quad (4.12)$$

In addition to the parameters that we've already described, the RED-BL optimization problem includes three auxiliary indicator decision variables:  $p_i^j$ ,  $b_i^j$  and  $s_i^j$  (a summarized description of all parameters and variables is given in Table 4.1).  $p_i^j$  is 1 if the elastic load in data center  $i$  is active during interval  $j$ , or 0 otherwise. Similarly,  $b_i^j$  (and  $s_i^j$ ) is 1 if the elastic load in data center  $i$  is booted up (shut down) at the start of interval  $j$ . In equation 4.5, multiplication of the first two terms 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. Similarly, multiplication of the last two terms in equation 4.5 by  $b_i^j$  and  $s_i^j$ , respectively, ensures that bootup and shutdown costs contribute to the summation only when the elastic load in a data center is booted up or shutdown.

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. If the elastic load in data center  $i$  is inactive in interval  $j - 1$  and active in interval  $j$ , then there is an activation at the beginning of interval  $j$ . The constraint in Eq. (4.10) ensures that  $b_i^j$  is 1 in such cases and 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 shutdown, 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.

As discussed in Section ??, the optimal workload mapping and relocation problem is NP-Complete. While we are able to solve reasonably sized instances of the problem to global

optimality over 24-hour planning windows, it is appropriate to also propose a heuristic algorithm for the problem.

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 mannner to another trough. Under this assumption, the two ends of the workload vector define the intervals for which the minimum sufficient number  $f$  data centers is the smallest over the entire planning window. Without loss of generality, let's assume that one data center is sufficient for the workload at the two ends of the planning window. The single data center to be picked to run continuously from the first interval to the last, then, would be the one that has the lowest average electricity price. Based on this idea, we have defined a heuristic that places two pointers at the intervals that define the first and last intervals for which a certain number of data centers is sufficient, then picks the ones that have the least average electricity price in those intervals before sliding these pointers to their new appropriate position. The pseudo-code of our heuristic algorithm for RED-BL is given in Algorithm 2. It is designed based on the observation that the cumulative hourly workload increases or decreases slowly in somewhat fixed patterns. Since the bootup/shutdown 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.

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 conincides 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 it's 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  $mn + (m-1)(n-1) + (m-2)(n-2) + (m-3)(n-3) + \dots + (m-n+1)$  primitive operations. Sorting the electricity prices will require  $O(m \lg m) + O((m-1) \lg (m-1)) + \dots + O((m-n+1) \lg (m-n+1))$  running time, overall. Within the nested for-loops, line 13 is most complex which will require, in the worst case,  $(m-1) + (m-2) + (m-3) + \dots + (m-n+1)$  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  $n$ .

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

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  $O(mn + m \lg m)$ .

---

**Algorithm 1** Heuristic for the RED-BL problem

---

**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 each data center over all intervals in the planning window  
 $y[1..m][1..n]$ : Array of data center status (1=on/0=off) over the planning window

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

---

## 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 [68]. In order to make the dataset representative of a large data center network operator, we normalized 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 4.1. The statistical characteristics of our workload, as plotted in Figure 4.2 are quite similar to those reported by Google for “thousands of servers during a six-month interval at a Google data center” [10].

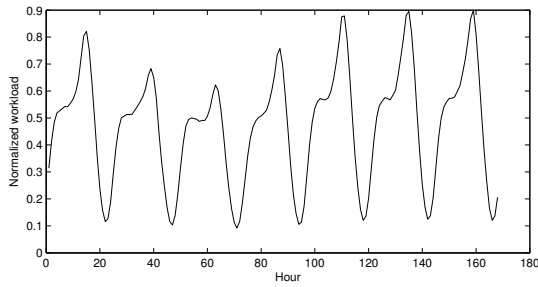


Figure 4.1: Normalized workload

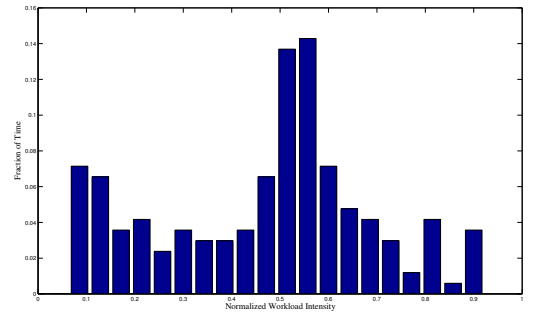


Figure 4.2: Workload intensity histogram

S. No.	Source	S. No.	Source	S. No.	Source	S. No.	Source	S. No.	Source	S. No.	Source
1	CAPITL	7	LONGIL	13	OH	19	MISO	25	WCMass	31	SEMass
2	CENTRL	8	MHKVL	14	PJM1	20	Illinois	26	Boston	32	Vermont
3	DUNWOOD	9	MILLWD	15	WEST	21	Cinergy	27	CT	33	PJM2
4	GENESE	10	NYC	16	PGAE	22	Michigan	28	Maine		
5	HQ	11	NORTH	17	SCE	23	Minnesota	29	NH		
6	HUDVL	12	NPX	18	SDGE	24	First Energy	30	Rhode		

Table 4.2: Sources of electricity prices used in our work

## 4.5.2 Electricity prices

We selected 33 different regions in the USA for which hourly electricity prices are available online. To ease reproducibility of our results, the names for these locations as used in the online datasets are listed in Table 4.2. We used the day-ahead prices for these locations, i.e., the electricity price negotiated for the same hour on the following day. We formulated the following deployment scenarios using this dataset:

1. **Single site operator:** An operator that only has one data center at a fixed location. We considered 33 different cases in this type of deployment, corresponding to each location for which we had electricity price data.
2. **Multi-site operator:** For a multi-site operator, we considered six different deployment scenarios. Five of these scenarios represent operators with data centers at the first 10, 15, 20, 25 and 30 locations that we selected, while the sixth scenario represents an operator with a data center at all 33 locations.

## 4.5.3 Algorithms for Workload Distribution/Relocation

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

- For a given interval, the strategy for distribution of workload amongst data centers.

Algorithm	Remarks
LI	Local optimal with idling
LD	Local optimal with deactivation
LS	Local optimal with selection
LO	Local optimal without transition costs
RED-BL	The global optimal solution for the planning window
UNIFORM	Distribute workload equally over all data centers in all intervals
STATIC_MIN	A single site data center operator

Table 4.3: Algorithms compared in our work

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

Table 4.4: A comparison of the algorithms studied in this paper



- For a given interval, the strategy for the elastic load at a data center which has not been assigned any workload. For such a data center, an algorithm may either keep the elastic load active, thereby incur idling electricity costs, or it may deactivate the elastic load.
- Over the planning window, does the algorithm report transition costs in the total electricity cost?

In this paper, we report comparative results for seven workload placement algorithms, listed in Table 4.3. The following list describes and differentiates these algorithms. the same comparison is also presented in tabular form in Table 4.4.

- **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 is governed by the optimal solution as determined by the CPLEX solver. Similarly, RED-BL decides to idle or deactivate unneeded elastic load based on the optimal solution returned by the CPLEX solver.
- **Heuristic:** This is the heuristic algorithm that we proposed in Section 4.4.
- **UNIFORM:** This algorithm equally distributes workload amongst all data centers. It has the sometimes desirable property of even utilization across all data centers. This algorithm does not deactivate elastic loads and hence does not incur transition costs.
- **STATIC\_MIN:** This algorithm considers only one data center with unit workload capacity that is situated at the location that has the lowest average electricity price over the planning window. While this runs counter to the latency-minimizing philosophy of distributed data centers, it offers a good baseline for evaluation of benefits of geographic diversity. Since there is only one data center, the workload mapping strategy is trivial, there is no workload relocation and hence no transition costs.

- **Greedy algorithms:** The originally proposed algorithm in [36] distributes workload to data centers such that it picks the fewest data centers needed from the list of data centers sorted in order of increasing electricity prices. It then assigns workload to data centers starting with the cheapest one, until all workload is distributed. Furthermore, this original algorithm keeps 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 [36]. It does not deactivate elastic load.
- **Local optimal withOut transition costs (LO):** This is the variation of LI that was also proposed in [36]. However, this algorithm ignores transition costs and does not report them. The utility of this algorithm in our work is that it defines the lower bound on electricity cost that any algorithm can ever achieve.
- **Local optimal with Deactivation (LD):** This algorithm always deactivates unneeded elastic load at data centers and reports the resulting transition costs in the total cost as well. This would be an improvement over LI making it somewhat competitive with 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 the elastic load is not needed at a particular data center for a small number of consecutive 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.

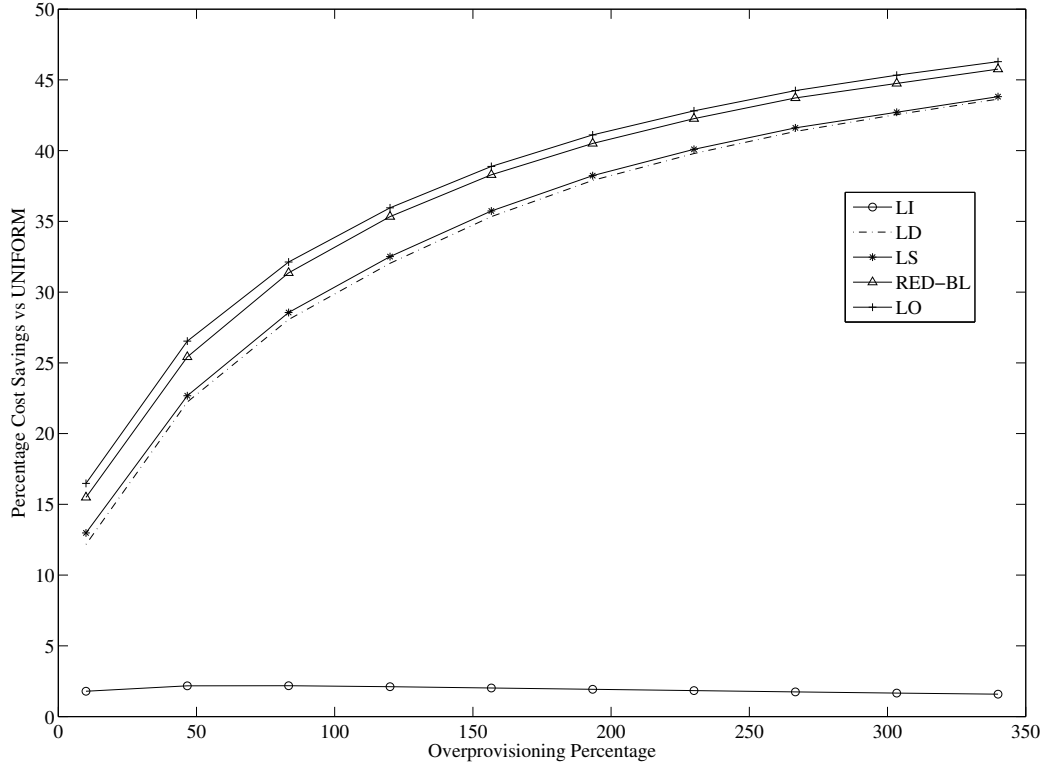


Figure 4.3: Percentage savings with over-provisioning

## 4.6 Results

To evaluate the utility of workload relocation for electricity cost minimization, we formulated five different scenarios. For each scenario, we ran 7 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.

### 4.6.1 Sensitivity of electricity cost savings to extent of overprovisioning

In this scenario, we investigate the relationship of the amount of data center network 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, except `STATIC_MIN`<sup>2</sup>. In this scenario, we set  $f = \sigma = \delta = 0.65$ . The percentage savings in total electricity cost by various algorithms compared to `UNIFORM` are plotted against the data center capacity over-provisioning in Figure 4.3. An interesting observation is that for the wide range of capacity over-provisioning that we considered, `LI` is able to do only slightly better than `UNIFORM`. This is because `LI` incurs significant idling costs because it does not deactivate the elastic load at data centers that are not computing any workload.

The cost of the greedy strategy can be significantly improved, compared to that of `LI`, by deactivating the elastic load at data centers when they are not needed. Prior work has reported this (shown as `LO`) while assuming that there are no transition costs. Such costs, however, exist and the curves for `LD` and `LS` indicate that the total cost (including transitions) is indeed higher. For example, `LS` offers 10.35% less cost savings than `LO`, on average. The good news is that the cost of `RED-BL` solution is quite close the ideal lower bound (`LO`).

The reason for greater savings with `RED-BL` compared to the greedy solutions (`LS` and

---

<sup>2</sup>`STATIC_MIN` uses a single data center which is big enough to handle peak workload, so scaling it’s capacity would not bring any extra savings.

LD) is that, the transition costs being significant, the former does fewer state transitions in the latter. In several intervals, RED-BL choses 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<sup>3</sup>, compared to UNIFORM, while varying the activation/deactivation overhead between 0 and 1, in increments of 0.1. The lower bound on  $\sigma$  (and  $\delta$ ) 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 it's 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 4.4 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

---

<sup>3</sup>STATIC\_MIN uses a static workload mapping so it is insensitive to variations in transition costs

that is fairly close (only 3% higher, on average) than the ideal lower bound as reported by LO.

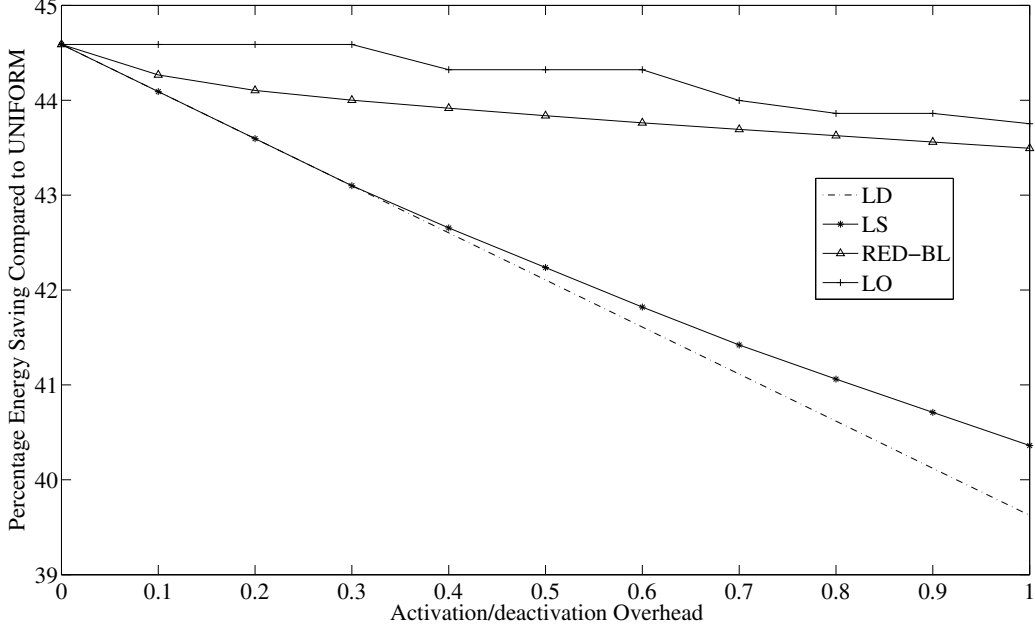


Figure 4.4: Total cost vs transition overhead

### 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 it's 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:

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

subject to:

$$x_i^j \leq c_i \quad \forall i, \forall j \quad (4.13)$$

$$\sum_{i=1}^m x_i^j = w^j \quad \forall j \quad (4.14)$$

$$p_i^j, b_i^j, s_i^j \in \{0, 1, \dots, l\} \quad \forall i, \forall j \quad (4.15)$$

$$p_i^j \geq x_i^j * l / c_i \quad \forall i, \forall j \quad (4.16)$$

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

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

$$b_i^0 = p_i^0, s_i^0 = 0 \quad \forall i \quad (4.19)$$

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  $0, 1, \dots, l$ .

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 4.10, we have 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 4.10 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

With the exception of scenario 3, 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 [69]. 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 time-scales. 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  $\gamma$  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  $\gamma$ , 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 4.7, 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

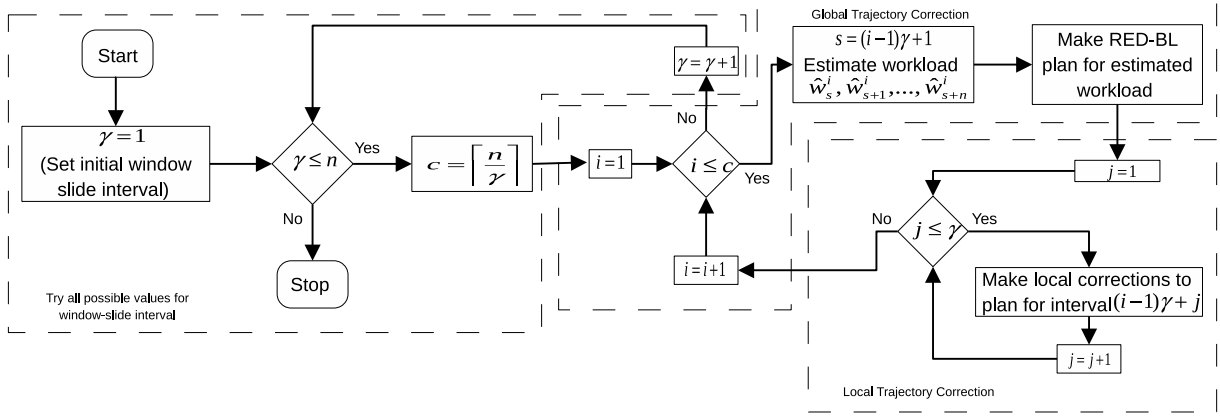


Figure 4.5: What

for the local trajectory correction step for interval  $j$  is as follows:

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

subject to:

$$x_i^j \leq c_i \quad \forall i \quad (4.21)$$

$$\sum_{i=1}^m x_i^j = w^j \quad (4.22)$$

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

$$p_i^j \geq x_i^j \quad \forall i \quad (4.24)$$

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

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

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

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

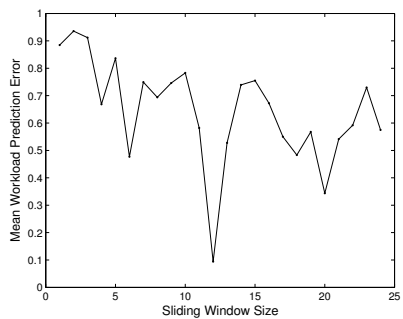


Figure 4.6: Mean absolute workload prediction error vs sliding window size

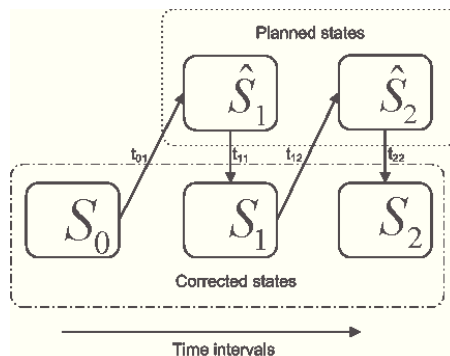


Figure 4.7: Local trajectory correction technique for three consecutive intervals

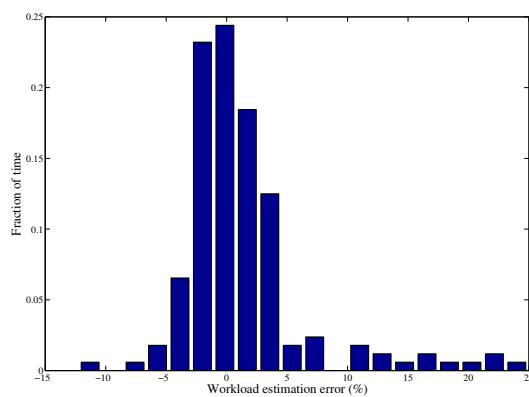


Figure 4.8: Distribution of workload prediction error for sliding window size of 12 hours

Given that the planning window size is  $n$  intervals, the possible values for  $\gamma$  are  $1, 2, \dots, \gamma$ . We experimented with all possible values for  $\gamma$ . Figure 4.5 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  $\hat{W}_1^2 = [\hat{w}_{1,1}^2, \hat{w}_{2,1}^2, \dots, \hat{w}_{n,1}^2]$ . Here,  $\hat{w}_{j,1}^2$ , for instance, represents the workload forecast for interval  $j$  during the first forecasting operation while the value of  $\gamma$  is 2. For workload forecasting, we trained an ARMA(4, 4) [70] 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  $\gamma$  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  $\hat{W}_2^2 = [\hat{w}_{3,2}^2, \hat{w}_{4,2}^2, \dots, \hat{w}_{n+2,2}^2]$ . Then, we compute the RED-BL deployment plan for intervals  $3, 4, \dots, n + 2$  as the global trajectory correction step. Since the window sliding interval size is  $\gamma$  and the number of intervals in our experiments is  $n$ , the number of times the window must slide, for a given value of  $\gamma$  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 4.6 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 4.8. 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 to 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 4.9. 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.

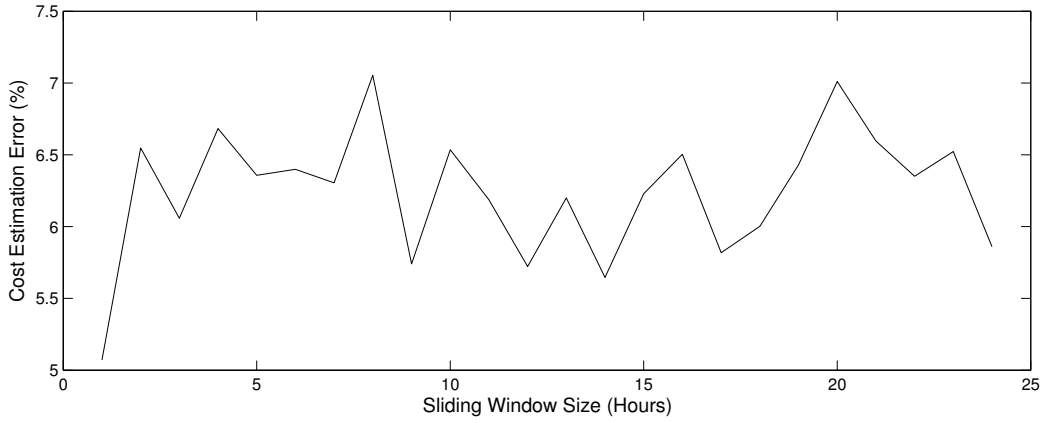


Figure 4.9: Percentage error of sliding window forecasts compared to global optimal with error-free workload

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

Chip manufacturers and computer vendors are striving to improve server 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 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.

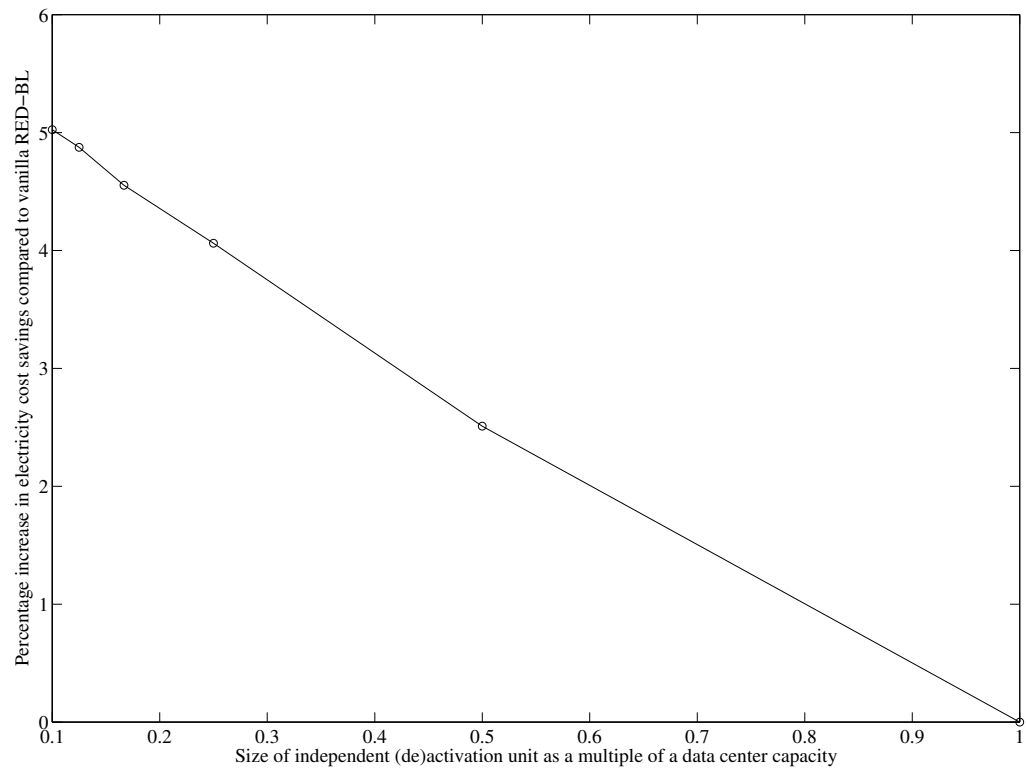


Figure 4.10: Cost saving vs (de)activation granularity

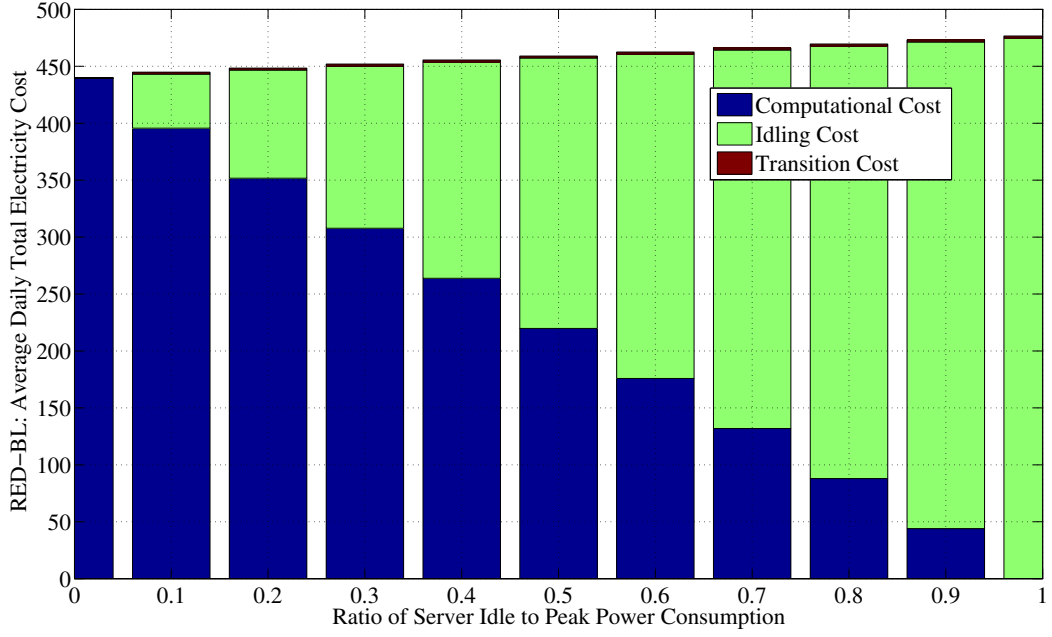


Figure 4.11: Average daily total electricity cost and it's components vs  $f$ , For  $b/s = 0.01$

Figure 4.11 shows the variation of electricity cost savings as the value of  $f$  is varied while the  $b/s$  parameteres 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 4.11 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 4.12 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

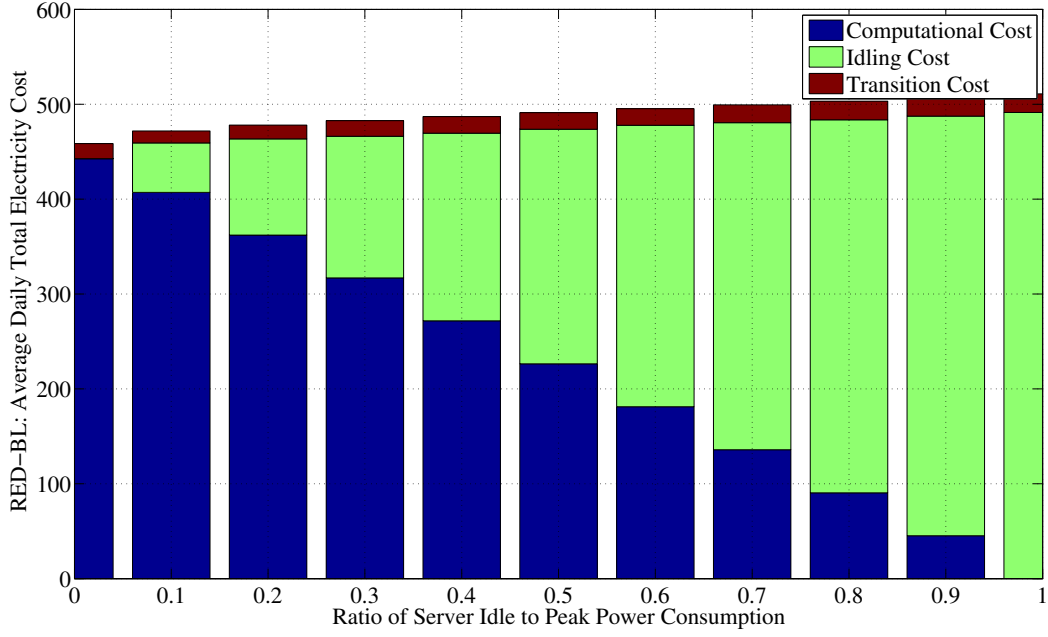


Figure 4.12: Average daily total electricity cost and it's components vs  $f$ , For  $b/s = 0.65$

solution over the planning window is characterized by an almost fixed contribution from bootup/shutdown, while the idling fraction increase almost linearly.

## 4.8 Performance of the heuristic algorithm

Figure 4.13 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



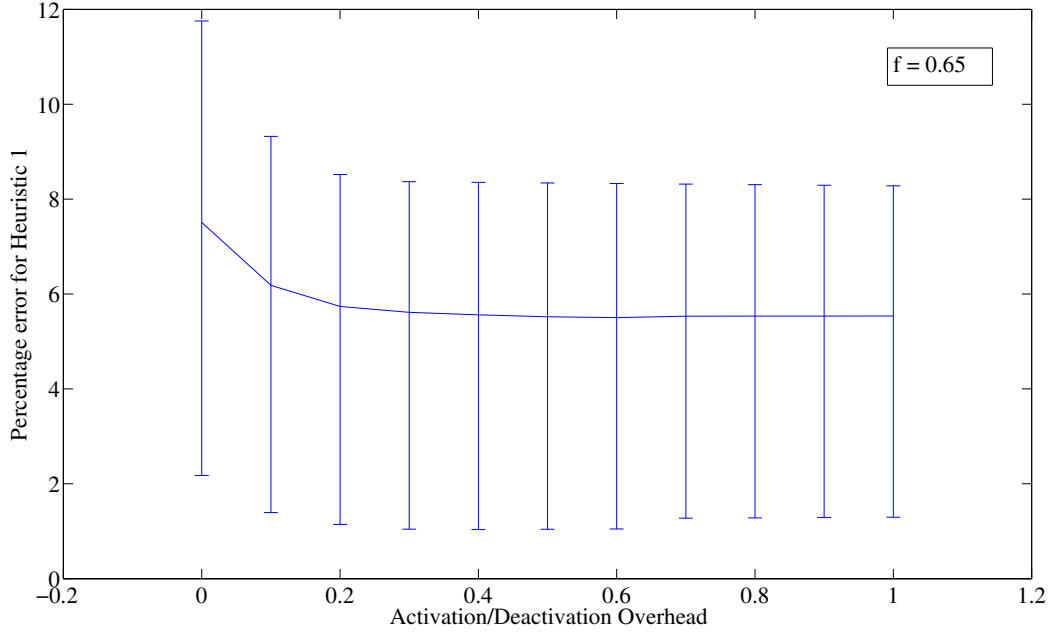


Figure 4.13: The minimum, maximum and average percentage difference between the cost of our heuristic and RED-BL

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.

## 4.9 Discussion

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 in the form 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 work 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 of electrical energy every year worldwide [23]. This not only results in significant operational expenditure, which is increasing with rising electricity and fuel prices, but is also a source of concern for ecological reasons. These concerns have motivated a lot of research aimed at reducing energy consumption in cellular networks.

In this thesis, our discussion focuses on the legacy 2G GSM cellular networks, which have a significant market share. Our focus on GSM is also due to their dominance in the typically energy-starved developing countries with a large subscriber base. These GSM networks are expected to persist for the foreseeable future due to the upgrade expenses and return on investment (ROI) concerns of the operators. It is, therefore, important to optimize such GSM networks. The energy-saving techniques proposed in this paper may be applicable to 3G and 4G networks, but we make no claims as to the effectiveness of the same.

We have seen in chapter 1 that cellular networks lack energy efficiency and which makes electricity cost minimization all the more important. We also highlighted that our thesis uses

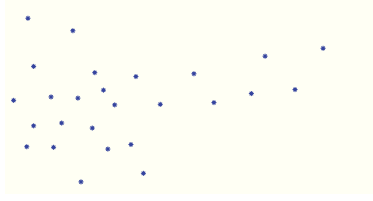


Figure 5.1: Location of the BTS sites in our dataset (spread over a  $31.25 \text{ km}^2$  urban terrain)

resource pruning and workload relocation jointly to minimize network electricity costs. The major sink of power in a cellular network are Base Transceiver Stations (BTSs), accounting for 60% to 80% of the total power consumption [22, 23, 71]. In chapter 2, we discussed that in terms of our thesis, the TRXs are the resource units in a cellular network settings because they determine the network’s traffic capacity and influence power consumption [71]. Therefore, the resource pruning strategy in a cellular network setting is deactivating presently redundant TRXs.

Also, since call volume determines a BTS’s power consumption, the workload relocation strategy is call hand-off. In present day deployments, most callers receive sufficiently strong signal simultaneously from several nearby BTSs, especially in an urban setting, some of which may have relatively lower traffic [71]. Fig. 5.2 shows coverage diversity evident in the urban data from a large cellular provider that we used in our evaluations; one can see that about half of the callers have 3 or more candidates for serving BTS. Furthermore, Fig. 5.3 shows normalized traffic at two neighboring sites in our dataset for a 24 hour period, which confirms the presence of geographic diversity in traffic. It is thus possible that by handing off some calls from a busy BTS to a less busy one, we may be able to prune more TRXs than we could without call hand-offs. Therefore, the workload relocation part of RED-BL maps onto call hand-offs.

The goal of the above discussion is to highlight the applicability of RED-BL to the cellular network setting. However, before instantiating RED-BL for the cellular network scenario, we describe some of the related work in electricity cost optimization in cellular networks.

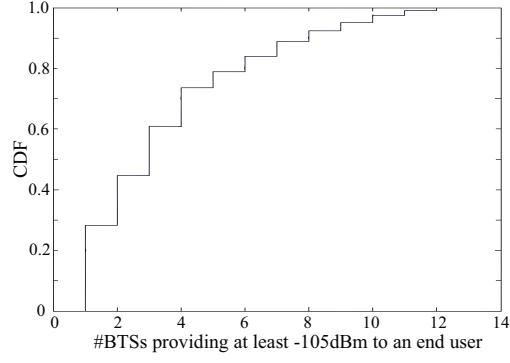


Figure 5.2: Empirical CDF of the number of potential serving BTSs for a call in our dataset (large metropolitan area)

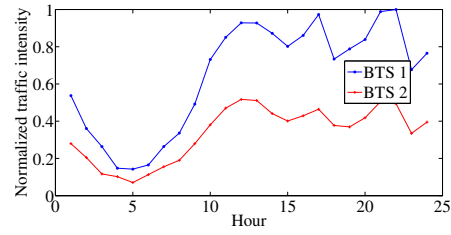


Figure 5.3: Normalized traffic for a 24-hour period at two neighboring sites in our dataset shows the geo-diversity in traffic

## 5.2 Related work

There are generally two approaches to increase cellular network energy efficiency. First, a clean-slate redesign which includes innovations in communication systems, circuits and components. This approach is not attractive for existing GSM operators, which are the most prevalent in the developing world and are expected to stay as such for several years to come, primarily due to the required expensive upgrades. A second approach is to make optimizations to the existing system and equipment to get an improvement in overall energy efficiency. Our present work is aligned with this latter philosophy.

One can improve the energy efficiency of a cellular network by adapting its “online” capacity to changes in traffic load. Recent work has proposed turning off base stations to reduce energy consumption during times of low traffic load [22, 23, 71, 72]. In such solutions, to offer the same amount of coverage, the transmit range of some of the remaining BTSs has to be increased. Our conversations with multiple network operators indicate that they are reluctant to employ such techniques citing three reasons:

- Power cycling of entire base stations is expected to reduce equipment life time.
- Turning off some BTSs may require an increased uplink power which may not be handled by many low-cost/power-limited mobile stations (MSs). This raises a risk of customer churn and is not acceptable to the operators in cut-throat competition prevalent in today’s market.
- These techniques of turning off BTSs may underestimate the increase in power needed for indoor MSs.

Our work is very similar in spirit to the concept of *frequency dimming* in [73] albeit at a different level of abstraction. A similar approach is also proposed in [74] with some rough estimates of expected savings. We, on the other hand, use site locations and 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. A key benefit of our approach is that it does not require any additional hardware and works within the GSM specifications.

Much prior work focuses primarily on recent and future generation cellular networks such as WiMax and LTE. The operational nature of GSM, the focus of our thesis, is so different from these networks that the energy efficiency improvement techniques proposed for such networks do not apply to GSM. Nonetheless, in this section, we do touch upon some of the recent work on energy efficiency in non-GSM networks as well. In [75], Deruyck et. al. have proposed a useful metric, power consumption per covered area, to measure the energy efficiency of WiMax, fixed WiMax, UMTS, HSPA and LTE. The authors rank these networks in order of energy efficiency based on their proposed metric and also investigate the impact of MIMO on energy efficiency of these networks. Good surveys of BTS energy efficiency improvement techniques for 3G networks may be found in [76, 77].

Transmit power adjustment for BTSs is critical for the operation of networks such as CDMA and WiMax, as it determines the interference level in the cells. Several researchers have investigated optimal transmit power allocation techniques, for instance see [78, 79]. Transmit power adjustment using cell-breathing was proposed in [80] with the objective of reducing energy consumption in 3G cellular networks. Han et. al. also proposed a cell-breathing based approach for cellular networks in the presence of green energy sources in [81].

Small cells such as pico and femto cells are also gaining popularity due to better energy efficiency compared to large macro-cells. Such cells are generally deployed within large organization's premises to enhance indoor coverage without significant increase in transmit power. We have not considered this type of cells because our conversations with network operators revealed that they are not common in the developing world. The interested reader



will find some investigation of energy efficiency improvement and power optimization in the presence of pico and femto cells can be found in [82, 83]

## 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. Also, these 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.

### 5.3.1 Single Base Transceiver Station (BTS)

Power consumed by a BTS, as a function of traffic load, can be well approximated as an affine function of traffic volume [71] given as  $P_1 + l(P_2 - P_1)/t_{max}$ . Here  $P_1$  and  $P_2$  are the power consumption at no load and full load, respectively,  $l$  is the number of calls presently being handled, and  $t_{max}$  is the maximum number of simultaneous calls that can be handled.

Let  $\delta$  be the traffic threshold below which the *BTS power savings* may be applied (i.e., when BTS deactivates some TRXs moving into low-power mode). Since all TRXs are identi-

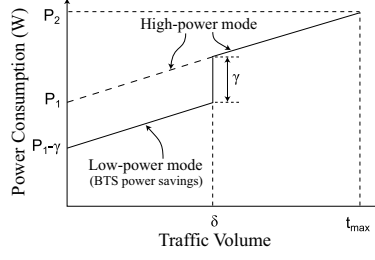


Figure 5.4: BTS power consumption model. Low-power (BTS power savings) mode is optional and kicks in at low loads.

cal, the per call increase in power consumption, and hence the slope of the power consumption profile in Fig. 5.4, remains the same in the default high-power mode, with all TRXs active, and the low-power mode, where some TRXs are deactivated. As also indicated in Fig. 5.4, the no-load power consumption drops to  $P_1 - \gamma$  in the low-power mode, where  $\gamma$  is a constant that depends on the equipment type and the number of TRXs deactivated. If  $x$  is an indicator variable which is 0 when *BTS power savings* is applied, and 1 otherwise, then the BTS power consumption may be given by  $P_1 + l(P_2 - P_1)/t_{max} - (1 - x)\gamma$  (indicated in Fig. 5.4 by the piecewise linear solid line).

### 5.3.2 Multi-BTS Cellular Setting

Consider an area with  $n$  active callers being served by  $m$  BTSs. We introduce indicator variable  $w_{i,j}$ , which is 1 if call  $i$  is being handled at BTS  $j$  and 0 otherwise. We assume availability of an  $n \times m$  matrix whose entry  $c_{i,j}$  is 1 if caller  $i$  can be served through BTS  $j$  without exceeding the uplink or downlink budgets. This information can be extracted by the data periodically transmitted by each MS comprising the received signal strength from nearby BTSs during a call. We also introduce indicator variable  $x_j$ , which is 1 if BTS  $j$  is operating in high-power mode and 0 otherwise. The total power consumption may, therefore, be given as  $\sum_{j=1}^m \left[ P_1 + \sum_{i=1}^n \frac{w_{i,j}(P_2 - P_1)}{t_{max}} - (1 - x_j)\gamma \right]$ . Using this as the objective function, we can formulate the Low-Carb optimization problem, subject to appropriate constraints.

Constant additive terms play no role in an objective function and may be omitted. The first term ( $P_1$ ) can, therefore, be dropped from the objective function. Furthermore, in order to not affect the grade of service, we will not drop any active calls. Hence, the summation over  $w_{i,j}$  is also constant and can also be excluded from the objective function. After removing the constant multiplier from the last remaining term in the objective function, Low-Carb may be stated as:

$$\text{minimize} \quad \sum_{j=1}^m x_j \quad (5.1)$$

subject to the following constraints:

$$\sum_{j=1}^m w_{i,j} = 1 \quad \forall i \quad (5.2)$$

$$w_{i,j} \leq c_{i,j} \quad \forall i, j \quad (5.3)$$

$$\sum_{i=1}^n w_{i,j} - \delta + \epsilon \leq Mx_j \quad \forall j \quad (5.4)$$

$$\sum_{i=1}^n w_{i,j} \leq t_{max} \quad \forall i \quad (5.5)$$

$$w_{i,j}, x_j \in \{0, 1\} \quad \forall i, j \quad (5.6)$$

The first constraint ensures that no active call is dropped just to save on power. The second constraint secures the uplink budget by ensuring that no call is routed to a BTS that is too far away. The third constraint picks the correct value for the decision variable  $x_j$  ( $M$  is a very large integer constant). The fourth constraint is the BTS capacity constraint, while the last constraint is the binary value constraint on the decision variables.

The Low-Carb problem itself can be shown as NP-Hard by mapping the multiple knapsacks problem [84] to it. Our formulation itself is an NP-Hard Binary Integer Program (BIP), making it intractable to solve for an operator's entire network. It could, however, be applied

separately to small disjoint network segments. Alternatively, a heuristic solution, such as the one we present in Algorithm 2, could be deployed over the entire network.

Our heuristic first partitions the set of BTSs  $B$  into disjoint sets  $B_1$  and  $B_2$  where the latter includes all the BTSs in low-power mode and the former consists of all other BTSs. The heuristic iterates over a random permutation of the BTSs in  $B_1$ . Once a BTS  $b_j$  from  $B_1$  is selected (line 3), we determine the minimum number of calls that must be handed off from  $b_j$  before it can be moved to  $B_2$ . We iterate over the calls being handled by  $b_j$  and for each such call we try to find a candidate serving BTS in  $B_2$  that may handle another call without leaving  $B_2$ . If such a BTS is found, the call is handed off to it. Calls are handed off from  $b_j$  in this manner, until it moves into  $B_2$ , or we exhaust the set of active calls with other candidate BTSs ( $b_j$  remains in  $B_1$ ).

## 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 sites are spread over a  $31.25 \text{ km}^2$  urban terrain (see Fig. 5.1). We obtained each site’s coverage prediction using a tool popular amongst the operators called Forsk Atoll. With this information, along with a caller’s location, we can determine the candidate set of BTSs for the corresponding call (the  $c_{i,j}$  parameters). Note that in this work, we do not incorporate user mobility into our model, since we are only interested in instantaneous optimization at small time scales and in determining bounds on the energy savings that Low-Carb can offer.

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 experiments mimicking a 24-hour oper-

---

**Algorithm 2** Heuristic for the Low-Carb problem

---

**Require:**  $\delta$ : the power-saving traffic threshold,

B (the set of BTSs) =  $\{b_1, b_2, \dots, b_m\}$ ,

A (the set of active calls) =  $\{a_1, a_2, \dots, a_n\}$ ,

W (current call association) =  $\{w_{i,j} = 1 \text{ if } a_i \text{ is being served through } b_j, 0 \text{ otherwise}\}$ ,

C (Possible call association matrix) =  $\{c_{i,j} = 1 \text{ if } a_i \text{ can be served through } b_j, 0 \text{ otherwise}\}$

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

```
1:  $B_1 = \text{random\_shuffle}(\{b_j | \sum_{i=1}^n w_{i,j} > \delta\})$ ;
2:  $B_2 = B - B_1$ 
3: for all  $b_j \in B_1$  do
4:    $\gamma = \sum_{i=1}^n w_{i,j} - \delta$ ;   shuffled = 0;
5:    $A^j = \{a_i | w_{i,j} = 1\}$ ;   shuffled = 0;
6:   for all  $a_k \in A^j$  do
7:     if shuffled <  $\gamma$  then
8:        $B_2^k = \{b_p | c_{k,p} = 1, b_p \in B_2\}$ ;
9:       for all  $b_p \in B_2^k$  do
10:        if  $\sum_{q=1}^n w_{q,p} < \delta$  then
11:           $w_{k,q} = 1$ ;   shuffled++;
12:           $w_{k,j} = 0$ ;   break;
13:        end if
14:      end for
15:    end if
16:  end for
17:  if  $\sum_{i=1}^n w_{i,j} \leq \delta$  then
18:     $B_1 = B_1 - \{b_j\}$ ;    $B_2 = B_2 + \{b_j\}$ 
19:  end if
20: end for
```

---

ation of a subset of a cellular network. Each experiment is a discrete event simulation of the arrival and placement of calls. Since our dataset does not include the arrival times and duration of calls, we synthetically generated this information using the assumption of Poisson call arrivals and exponentially distributed call duration with a mean of 180 seconds [85].

For every hour, the simulator determines the Poisson call arrival rate for each BTS, using Little’s Law and the BTSs traffic intensity for that hour. Using the resulting Poisson process, calls are generated such that it is equally likely for a call to be anywhere in the serving BTSs coverage area.

Our simulator tracks the call volume at every BTS on a minute’s granularity, thereby computing a time series of the power consumption (in Watts) for each BTS. This enables it to calculate three values. First, it accumulates the power consumption for each BTS over the 24 hour period, thereby computing the daily amount of energy consumed (in kWh) if no optimization is used in the network. Second, by identifying low-traffic episodes for each BTS the simulator selectively places some BTSs in power-saving mode, thereby computing the possible energy savings using BTS power-saving feature. Third, our simulator also periodically determines the instantaneous optimal call placement configuration using call hand-off, such that a maximal number of BTSs are placed in power-saving mode, thereby determining the optimal energy savings by coordinated call hand-offs and BTS power-saving.

The call placement re-optimization may be done at various frequencies. A very aggressive re-optimization regime would keep the network in an optimal state more often than a conservative one, thereby enabling greater energy savings. In order to study the scaling of energy saving with re-optimization frequency, we experimented with a range of intervals between successive optimizations, ranging from a minute to an hour. For a deployment, the re-optimization frequency that can be used would depend on the costs associated with each re-optimization. Let us now consider such costs.

First and foremost, a computational cost is incurred with each optimization. For our

dataset, an optimization run to determine the optimal state over 26 BTSs required an average running time of about 50 seconds on a Core i3 laptop with 4 GB of RAM. An optimization requiring 50 seconds would not be practical to use every minute but may be fine if used less often. For a practical deployment the computational time can be reduced by using a combination of a more powerful machine, distributed optimization and approximation algorithms.

In addition to the computational overhead, for every unit of energy saved some extra energy may be consumed in the network to perform call hand-offs or transitions into and out of BTS power-saving mode. Call hand-offs and TRX (de)activation involve signaling between a Base Station Controller (BSC), BTSs and MSs. The additional energy incurred thus, should be small, because it has been observed that variation in power consumption of network equipment with changes in traffic volume (data or control) is quite small [86]. As far as increased power consumption on MSs due to a greater number of call hand-offs is concerned, we opine that it may be negligible because the MSs energy consumption is far outweighed by that of BTSs.

### 5.4.1 Site Characteristics

All sites in our dataset had three sectors, each equipped with 6 TRXs, for a maximum of 132 simultaneous voice calls<sup>1</sup>. The GSM standard includes a provision for half-rate calls, which enables handling greater traffic at the expense of reduced voice quality by allowing a single voice channel to be shared amongst two calls, each using a half-rate codec. In this paper, we only shuffle full-rate calls around, which may be, in reality, two half-rate calls. We do not foresee any significant error arising from using this convention.

We consider a scaling down from a “6 + 6 + 6” site to a “2 + 2 + 2” site, which means

---

<sup>1</sup>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.

that  $\delta$  should be strictly less than  $t_{max}/3$  to avoid quick oscillations into and out of BTS power-saving mode due to short-term traffic variations. We have arbitrarily set  $\delta$  equal to  $\lceil t_{max}/3 \rceil - 5$ , because 5 seemed to be a good enough number compared to a sector's overall capacity and the typical utilization of a site in our datasets.

The BTS power consumption model parameters may vary from one BTS model to another. In this paper, we use three different sets of model parameters as listed in table 5.1. We now describe the sources and methods from which we obtained these models.

### Model 1

For the first model, we have used  $1.5kW$  as the maximum power consumption [5], a  $20W$  per TRX saving when scaling the BTS down [87] and a 5% swing in power consumption between no-load and full-load [71].

### Model 2

Lorincz et. al reported the single sector DC power consumption for a GSM 900 BTS with 7 TRXs [24]. We extrapolate the power consumption for a 6+6+6 site by multiplication of the DC power consumption reported in [24] by  $3 \times 6/7$ . 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, which has a typical power consumption of 1 kW [5]. We used this scaling and AC load correction to obtain the values for  $P_1$  and  $P_2$  using the minimum and maximum reported power consumption in [24]. Furthermore, the authors in [24] measured a drop of  $50W$  in power consumption when a TRX is disable, which gives us the value of  $\gamma$  as listed in table 5.1.



Parameter	Value		
	Model 1	Model 2	Model 3
$P_1$	1425	2401.8	2341.5
$P_2$	1500	3887.5	2973.9
$\gamma$	240	600	1200

Table 5.1: BTS model parameter values

### Model 3

Using the same method as for model 2 in 5.4.1, we derived the values for  $P_1$  and  $P_2$  based on the measurements for a GSM 1800 BTS reported in [24]. As for the value of  $\gamma$ , the paper reported a 100W cut in power consumption when deactivating a single TRX. The parameter values for this model are given in Table 5.1

## 5.5 Results

First, we consider the benefit of BTS power-saving alone, resulting from traffic diversity at each BTS compared to running the network in the default configuration. The percentage reduction in energy consumption is listed in TABLE 5.2. The results indicate that a saving of between 4% and 12% can be achieved in a network just by enabling BTS power savings. We note here that these results are in agreement with Ericsson’s claim of saving 10-20% energy by using BTS power-saving on Germany’s Vodafone network [88].

In absolute terms, this represents a cumulative saving of between 43 kWh and 217 kWh per day over a set of 26 BTSs. Now, consider that there are five cellular operators in Pakistan: Mobilink with more than 8500 sites [89], Ufone with more than 8000 sites [90], Zong with more than 5500 sites [90], Telenor with more than 7000 sites [91] and Warid with more than 4500 sites [90]. Overall, there were more than 31000 sites in Pakistan at the end of 2011. Extrapolating our experimental results suggests that a significant reduction in energy consumption is possible country-wide in Pakistan (and any other country with

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 5.2: Energy savings by using BTS power savings only

similar deployment and traffic characteristics) by just deploying BTS power-saving. This is especially significant for an energy-starved developing country. As we shall see next, greater energy savings are possible if we couple periodical call shuffling with BTS power savings in the network.

### 5.5.1 Sensitivity of electricity cost savings to the duration of an optimization interval

Fig. 5.5 shows the percentage energy savings achievable by coordinated call handoff and BTS power saving. Notice that if the interval between consecutive optimizations is greater than the mean call duration of 6 minutes, the percentage savings don't change significantly, whereas for shorter inter-optimization intervals, the percentage savings rise sharply. This is due to the fact that if re-optimization is performed at a rate faster than the mean call duration, a greater number of calls is routed optimally at least once during their lifetime.

Since the three BTS models have quite different power consumption characteristics, percentage energy savings do not provide a useful metric for comparison of the relative utility of Low-Carb for these BTS models. We, therefore, also plot the absolute energy savings (in kWh) for each of the BTS models in Figure 5.6. For all three BTS models, we see the same trend in absolute energy savings as that in case of percentage energy savings.

Let us now interpret what these results mean physically in terms of ecological impact. An extrapolation of our results indicates that for Pakistan, Low-Carb offers a total energy saving

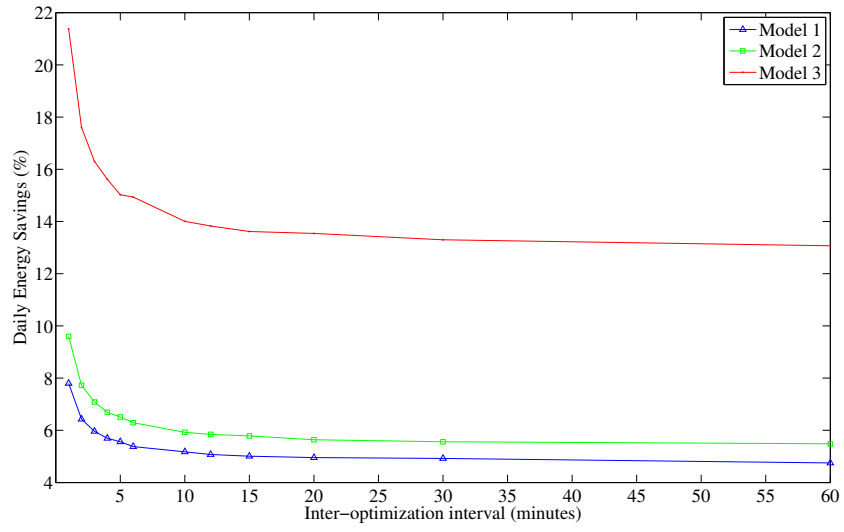


Figure 5.5: Percentage reduction in energy consumption vs inter-optimization interval

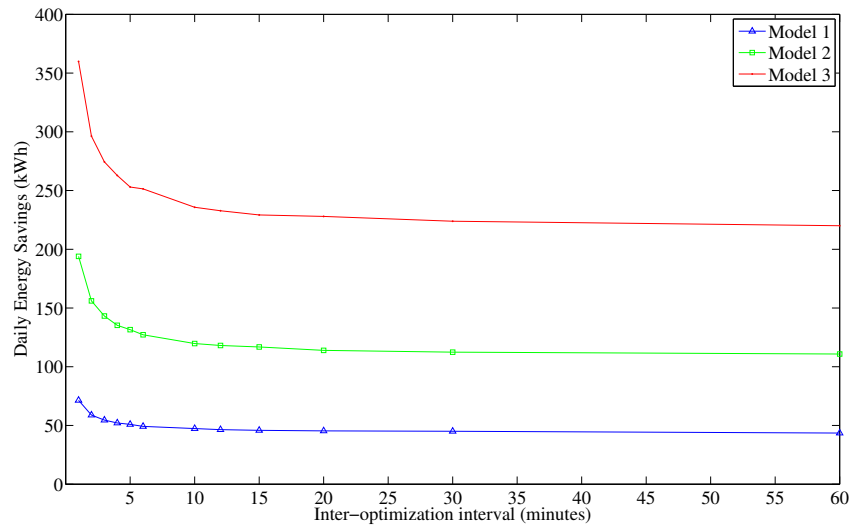


Figure 5.6: Percentage reduction in energy consumption vs inter-optimization interval

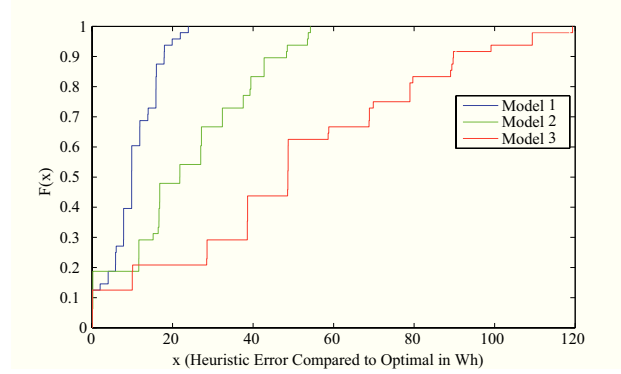


Figure 5.7: Empirical CDF of the difference between the cost offered by our heuristic compared to the optimal

of 60.72 MWh, 156.84 MWh or 301.61 MWh daily, respectively for each of the BTS models. For a small and energy-starved developing country, these energy savings are significant. Since network deployments and traffic patterns are similar in different countries, the same extrapolation can be applied to 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. One may argue that in rural settings, due to sparse deployments the energy savings potential would be lower because few calls would have multiple candidate BTSs. A counter-argument, however, is that in rural settings, call traffic is already low, which implies that BTS power-saving is applicable to most BTSs most of the time.

### 5.5.2 Comparison of our heuristic with the optimal solution

We also ran experiments for each BTS model in which the electricity cost for the optimal as well as the heuristic solution (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 Fig. 5.7. We can see in Fig. 5.7 that our heuristic is quite close to the optimal solution most of the time, especially for the Model 1 and Model 2 BTS. For Model 3 BTS, while the error is comparatively larger, since the amount of savings with the optimal solution is quite high (Fig. 5.5), the heuristic will still result in significant energy savings.

### 5.5.3 Sensitivity of electricity cost savings to the resource pruning granularity

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, i.e., at a given time, a site may be in one of six possible configurations.

For the first case, a simple extension of the optimization model given in 5.3.2 as follows:

$$\text{minimize} \quad \sum_{j=1}^m x_j + y_j \quad (5.7)$$

subject to the following constraints:

$$\sum_{j=1}^m w_{i,j} = 1 \quad \forall i \quad (5.8)$$

$$w_{i,j} \leq c_{i,j} \quad \forall i, j \quad (5.9)$$

$$\sum_{i=1}^n w_{i,j} - \delta_1 + \epsilon \leq Mx_j \quad \forall j \quad (5.10)$$

$$\sum_{i=1}^n w_{i,j} - \delta_2 + \epsilon \leq My_j \quad \forall j \quad (5.11)$$

$$\sum_{i=1}^n w_{i,j} \leq t_{max} \quad \forall i \quad (5.12)$$

$$w_{i,j}, x_j, y_j \in \{0, 1\} \quad \forall i, j \quad (5.13)$$

Here,  $x_j$  is a binary indicator variables that is 0 if the traffic at BTS  $j$  is below  $\delta_1 - \epsilon$  and 1 otherwise. Similarly,  $y_j$  is an indicator variable that is 0 if the traffic at BTS  $j$  is below  $\delta_2$  and 1 otherwise.

For the second case, we present the following slightly different optimization formulation. Consider an integer variable  $z_j$  which represents the number of TRXs active on BTS  $j$ . The power consumption at BTS  $j$  may be given by  $P_1 + p \sum_{i=1}^n w_{i,j} + \gamma z_j$ , where  $p$  is the slope of the power consumption vs traffic volume curve. The RED-BL optimization problem for the multi-BTS setting in this case is:

$$\text{minimize} \quad \sum_{j=1}^m z_j \quad (5.14)$$

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%
6-State	14.69%	25.33%	33.69%

Table 5.3: Percentage electricity savings for different granularity of resource pruning

subject to the following constraints:

$$\sum_{j=1}^m w_{i,j} = 1 \quad \forall i \quad (5.15)$$

$$w_{i,j} \leq c_{i,j} \quad \forall i, j \quad (5.16)$$

$$\sum_{i=1}^n w_{i,j} - \delta_1 + \epsilon \leq M z_j \quad \forall j \quad (5.17)$$

$$\sum_{i=1}^n w_{i,j} \leq t_{max} \quad \forall j \quad (5.18)$$

$$w_{i,j} \in \{0, 1\}; \quad z_j \in \{1, 2, 3, 4, 5, 6\} \quad \forall i, j \quad (5.19)$$

The first constraint ensures that each call is handled at exactly one BTS. The second one ensures that a call is mapped to a BTS that may handle it. The third constraint ensures that  $z_j$  picks a value at least equal to the minimum number of TRXs required at BTS  $j$ . Since the optimization is a minimization problem, the solver will pick  $z_j$  to be exactly equal to the minimum number of TRXs required for the current traffic load. The fourth constraint is the capacity constraint on the BTS. The last set of constraints specify the domains of the decision variables.

In this scenario, we conducted simulation experiments where a re-optimization was performed every six minutes using model 1 BTS. The results of these experiments are given in Table 5.3. 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 compared to the jump from 3-state to 6-state model.

### 5.5.4 Sensitivity of electricity cost savings to the margin of state-change damping

If the value of  $\epsilon$  in our optimization is set too aggressively, a BTS that is placed in low-power mode may have to be moved back to high-power mode soon afterwards due to short time scale variations in call volume. Theoretically, it is even possible that there may be several such back and forth transitions at a BTS. Such rapid state oscillations may be undesirable and to avoid these, the value of  $\epsilon$  must be set at a safe value. Furthermore, if  $\epsilon$  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  $\epsilon$  is set too conservatively, the energy savings would be smaller.

We carried out experiments to assess the impact of the value of  $\epsilon$  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<sup>1</sup>, any larger value for  $\epsilon$  did not make sense. Figure 5.8 shows the results. As expected, the percentage savings deplete almost linearly with increasing values of  $\epsilon$ .

## 5.6 Discussion

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

---

<sup>1</sup>The capacity of the 2+2+2 BTS is  $3 \times 2 \times 8 = 48$ , but 4 channels were reserved by the operator for control and broadcast channels.



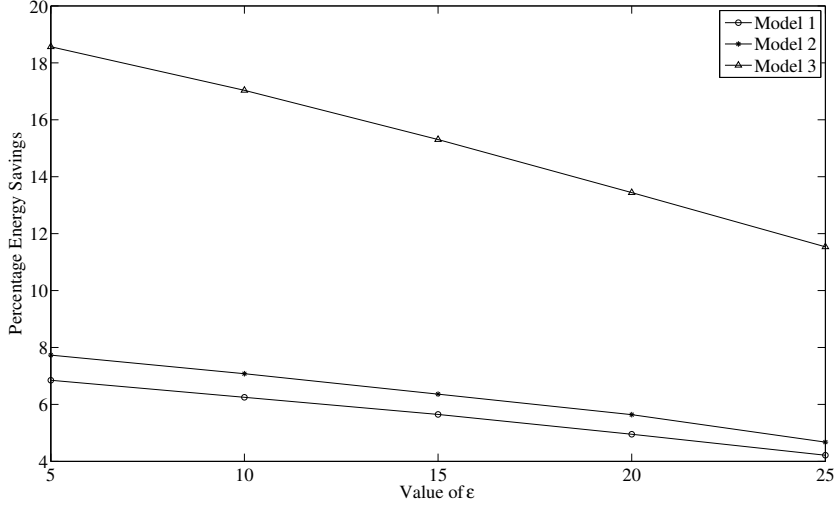


Figure 5.8: The percentage energy savings for all three BTS models considered in this paper vs the value of  $\epsilon$ , with a six minute inter-optimization interval

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 handoff, 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

### 6.1 Contributions

In this thesis, we have made the following contributions:

- We model networks as a set of resources that have user workload handling capacity. When a resource handles workload, some power is consumed. The power consumption is an affine function of workload. On this basis, we identified similarities and contrasts between two different types of networks: geo-diverse data centers and cellular networks.
- We 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 parametrized 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:

- In cellular networks, the number of workload units being handled simultaneously is much smaller compared to geo-diverse data centers. For this reason, finding the optimal fractional distribution of workload to a set of resources is practically acceptable for geo-diverse data centers but not to cellular networks. This is because a fractional mapping of workload to resources may represent a call being handled by multiple BTSs simultaneously. Meanwhile in the case of geo-diverse data centers, a reasonably small fraction of the typical cumulative hourly workload (millions of client requests or more) is still a whole number so a fractional workload distribution does not imply a single client request being split over multiple data centers.
  - 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.

## 6.2 Limitations

Like all research work, our work has some limitations. To the best of our knowledge, the following list best covers these:

- 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 due to (to the best of our knowledge) lack of models in the literature.
- We have not experimented with real-time electricity prices for geo-diverse data centers.
- We have not considered costs associated with maintaining replication across all data centers.
- 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. We have not focused on this because this is a complementary and specialized area of research which is presently receiving significant research attention.
- We have not considered the ability of GSM BTSs to handle calls at the half-rate codec, which effectively increases a BTSs call handling capacity.

## 6.3 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 inter-data center traffic, which is quite expensive. 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] S. Mbakwe, M. T. Iqbal, and A. Hsaio, “Design of a 1.5kw hybrid wind/photovoltaic power system for a telecoms base station in remote location of benin city nigeria,” in *IEEE NECEC*, November 2011.
- [6] T. Italia, “Telecom Italia Annual Report 2012,” January 2013. [Online; accessed 24-May-2013].
- [7] C. Peng, S.-B. Lee, S. Lu, H. Luo, and H. Li, “Traffic-driven power saving in operational 3g cellular networks,” in *Proceedings of the 17th annual international conference on*

- Mobile computing and networking*, MobiCom '11, (New York, NY, USA), pp. 121–132, ACM, 2011.
- [8] 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.
  - [9] 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.
  - [10] L. A. Barroso and U. Holzle, “The case for energy-proportional computing,” *Computer*, vol. 40, pp. 33–37, 2007.
  - [11] V. Inc, “Reduce Energy Costs and Go Green With VMWare Green IT Solutions,” March 2009. [Online; accessed 26-May-2013].
  - [12] 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.
  - [13] 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.
  - [14] 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.

- [15] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, “Dynamic right-sizing for power-proportional data centers,” in *IEEE INFOCOM*, 2011.
- [16] A. Qureshi, “Plugging Into Energy Market Diversity,” in *7th ACM Workshop on Hot Topics in Networks (HotNets)*, (Calgary, Canada), October 2008.
- [17] 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.
- [18] D. Abts and B. Felderman, “A guided tour of data-center networking,” *Commun. ACM*, vol. 55, pp. 44–51, June 2012.
- [19] 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.
- [20] 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.
- [21] T. Berners-Lee, R. Fielding, and H. Frystyk, “Hypertext Transfer Protocol – HTTP/1.0.” RFC 1945 (Informational), May 1996.
- [22] J. T. Louhi, “Energy efficiency of modern cellular base stations,” in *IEEE INTELEC ’07*, (New York, NY, USA), pp. 475–476, IEEE, 2007.
- [23] E. Oh, B. Krishnamachari, X. Liu, and Z. Niu, “Toward dynamic energy-efficient operation of cellular network infrastructure,” in *IEEE Communications Magazine*, June 2011.



- [24] 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.
- [25] 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 Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, pp. 1797–1807 vol.3, 2004.
- [26] A. H. Land and A. G. Doig, “An automatic method of solving discrete programming problems,” *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960.
- [27] K. G. Brill, “The invisible crisis in the data center: The economic meltdown of moore’s law,” tech. rep., The Uptime Institute, 2007.
- [28] C. L. Belady, “In the data center, power and cooling costs more than the it equipment it supports,” *Electronics Cooling*, vol. 23, Jan. 2007.
- [29] S. Pelley, D. Meisner, T. F. Wenisch, and J. W. VanGilder, “Understanding and abstracting total data center power,” *Proc. 2009 Workshop on Energy Efficient Design (WEED ’09)*, June 2009.
- [30] “Five ways to reduce data center server power consumption.” White Paper, Apr. 2008.
- [31] 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.
- [32] 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.

- [33] 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*, (New York, NY, USA), pp. 303–314, ACM, 2005.
- [34] 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.
- [35] 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.
- [36] A. Qureshi, R. Weber, H. Balakrishnan, J. Gutttag, and B. Maggs, “Cutting the Electric Bill for Internet-Scale Systems,” in *ACM SIGCOMM*, (Barcelona, Spain), August 2009.
- [37] 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 *Proceedings 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.
- [38] 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.
- [39] U. Bhaskar and L. Fleischer, “Online mixed packing and covering,” *CoRR*, vol. abs/1203.6695, 2012.
- [40] 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.
- [41] M. Mazzucco, D. Dyachuk, and R. Deters, “Maximizing cloud providers revenues via energy aware allocation policies,” *CoRR*, vol. abs/1102.3058, 2011.
  - [42] 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 *Proceedings of the 3rd USENIX conference on Hot topics in cloud computing*, HotCloud’11, (Berkeley, CA, USA), pp. 14–14, USENIX Association, 2011.
  - [43] 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.
  - [44] 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.
  - [45] 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.
  - [46] 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.
  - [47] 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.

- [48] 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.
- [49] 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.
- [50] 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.
- [51] 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.
- [52] 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.
- [53] 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 international conference on Supercomputing*, ICS ’11, (New York, NY, USA), pp. 275–284, ACM, 2011.
- [54] F. Javed and N. Arshad, “On the use of linear programming in optimizing energy costs,” in *Proceedings of the 3rd International Workshop on Self-Organizing Systems*, IWSOS ’08, (Berlin, Heidelberg), pp. 305–310, Springer-Verlag, 2008.

- [55] 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.
- [56] G. Celli and F. Pilo, “Optimal distributed generation allocation in mv distribution networks,” in *Power Industry Computer Applications, 2001. PICA 2001. Innovative Computing for Power - Electric Energy Meets the Market. 22nd IEEE Power Engineering Society International Conference on*, pp. 81 –86, 2001.
- [57] F. Javed and N. Arshad, “Adopt: An adaptive optimization framework for large-scale power distribution systems,” *Self-Adaptive and Self-Organizing Systems, IEEE International Conference on*, vol. 0, pp. 254–264, 2009.
- [58] 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*, (New York, NY, USA), pp. 159–168, ACM, 2010.
- [59] 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 *Optical Communication (ECOC), 2011 37th European Conference and Exhibition on*, pp. 1 –3, sept. 2011.
- [60] S. Lee and S. Sahu, “Efficient server consolidation considering intra-cluster traffic,” in *GLOBECOM*, pp. 1–6, 2011.
- [61] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath, “Load balancing and unbalancing for power and performance in cluster-based systems,” 2001.
- [62] 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 *INFOCOM, 2012 Proceedings IEEE*, pp. 1431 –1439, march 2012.

- [63] 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 Biennial Conference on Innovative Data Systems Research (CIDR) 2011*, pp. 261–272, January 2011.
- [64] H. Herodotou, F. Dong, and S. Babu, “No one (cluster) size fits all: automatic cluster sizing for data-intensive analytics,” in *Proceedings of the 2nd ACM Symposium on Cloud Computing*, SOCC ’11, (New York, NY, USA), pp. 18:1–18:14, ACM, 2011.
- [65] D. Aikema and R. Simmonds, “Electrical cost savings and clean energy usage potential for hpc workloads,” in *Sustainable Systems and Technology (ISSST), 2011 IEEE International Symposium on*, pp. 1–6, may 2011.
- [66] J. Pang, A. Akella, A. Shaikh, B. Krishnamurthy, and S. Seshan, “On the responsiveness of DNS-based network control,” in *IMC ’04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pp. 21–26, ACM, 2004.
- [67] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, “Delayed Internet Routing Convergence,” in *Proc. of the ACM SIGCOMM*, August 2000.
- [68] 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 conference on Internet measurement*, IMC ’08, (New York, NY, USA), pp. 43–56, ACM, 2008.
- [69] W. H. Kwon and S. H. Han, *Receding Horizon Control*. Elsevier, 2005.
- [70] G. Box, G. M. J. , and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*. Prentice-Hall, 1994.
- [71] 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.

- [72] H. He, J. Sun, Y. Wang, Q. Liu, and J. Yuan, “Cluster analysis based switching-off scheme of base stations for energy saving,” *Journal of Networks*, vol. 7, no. 3, pp. 494–501, 2012.
- [73] D. T. A. Rezgui, Krishnamurthy, and P. Pacharintanaku, “Dimming cellular networks,” in *IEEE GLOBECOM’10*, December 2010.
- [74] O. Blume, “Energy savings in mobile networks based on adaptation to traffic statistics,” *Bell Labs Technical Journal*, vol. 15, pp. 77–94, September 2010.
- [75] M. Deruyck, E. Tanghe, W. Joseph, and L. Martens, “Modelling and optimization of power consumption in wireless access networks,” *Computer Communications*, vol. 34, no. 17, pp. 2036 – 2046, 2011.
- [76] S. McLaughlin, P. M. Grant, J. Thompson, H. Haas, D. Laurenson, C. Khirallah, Y. Hou, and R. Wang, “Techniques for improving cellular radio base station energy efficiency,” *Wireless Communications, IEEE*, vol. 18, no. 5, pp. 10–17, 2011.
- [77] C. Han, T. Harrold, S. Armour, I. Krikidis, S. Videv, P. M. Grant, H. Haas, J. Thompson, I. Ku, C.-X. Wang, T. A. Le, M. Nakhai, J. Zhang, and L. Hanzo, “Green radio: radio techniques to enable energy-efficient wireless networks,” *Communications Magazine, IEEE*, vol. 49, no. 6, pp. 46–54, 2011.
- [78] V. Kavitha, S. Ramanath, and M. Debbah, “Satisfying demands in a multicellular network: A universal power allocation algorithm,” *Computer Communications*, vol. 36, no. 12, pp. 1373 – 1386, 2013.
- [79] H. Xu, X. Zhou, and Y. Chen, “Automatic uplink resource management in mobile cellular networks: A utility-based cooperative power control strategy,” *International Journal of Wireless Information Networks*, pp. 1–10, 2013.

- [80] S. Bhaumik, G. Narlikar, S. Chattopadhyay, and S. Kanugovi, “Breathe to stay cool: adjusting cell sizes to reduce energy consumption,” in *Proceedings of the first ACM SIGCOMM workshop on Green networking*, Green Networking ’10, (New York, NY, USA), pp. 41–46, ACM, 2010.
- [81] T. Han and N. Ansari, “Ice: Intelligent cell breathing to optimize the utilization of green energy,” *Communications Letters, IEEE*, vol. 16, no. 6, pp. 866–869, 2012.
- [82] R. Thakur, A. Sengupta, and C. Siva Ram Murthy, “Improving capacity and energy efficiency of femtocell based cellular network through cell biasing,” in *Modeling Optimization in Mobile, Ad Hoc Wireless Networks (WiOpt), 2013 11th International Symposium on*, pp. 436–443, 2013.
- [83] R. Estrada, A. Jarray, H. Otrok, Z. Dziong, and H. Barada, “Energy-efficient resource allocation model for ofdma macro-femtocell networks,” *Vehicular Technology, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2013.
- [84] H. Kellerer, U. Pferschy, and D. Pisinger, *Knapsack Problems*. Springer-Verlag, 2005.
- [85] M. Gerla and J. T.-C. Tsai, “Multicluster, mobile, multimedia radio network,” *Wireless Networks*, vol. 1, pp. 255–265, Aug. 1995.
- [86] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsang, and S. Wright, “Power awareness in network design and routing,” in *In Proc. IEEE INFOCOM*, 2008.
- [87] N. S. Networks, “Flexi BSC (BSC14).” <http://bit.ly/QMD805>, June 2011.
- [88] 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.
- [89] Mobilink, “Mobilink Network.” <http://bit.ly/UWXYzB>, September 2012.



- [90] Pakistan Telecommunication Authority, “Pakistan Telecommunication Authority, 2011 Annual Report.” <http://bit.ly/S8YtWS>, December 2011.
- [91] Wikipedia, “Telenor Pakistan - Wikipedia.” <http://bit.ly/R1Uyqm>, September 2012.