Impact of Data Center Infrastructure on Server Availability – Characterization, Management and Optimization

A Dissertation

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

In Partial Fulfillment

of the requirements for the Degree

Doctor of Philosophy (Computer Science)

by

Sriram Sankar

May 2014

©2014 Sriram Sankar

ABSTRACT

As cloud computing, online services and user storage continue to grow, large companies continue building data center facilities to serve end user requirements. These large data center facilities are warehouse-scale computers in their own right and the cost efficiency of such data centers is critical for both cloud and enterprise business. Data center infrastructure can be partitioned logically into IT infrastructure (server and network), Critical Environment infrastructures (power, cooling) and management infrastructure that coordinates all the other infrastructures. Although the IT component of the data center is crucial for applications to run, almost one-third of the total cost of ownership in a data center is spent towards building and operating the critical environment infrastructure. Data center operators strive to reduce the cost of the critical environment infrastructure, in order to increase the server portion of the capital expense investment. However, reduction of this cost usually comes at the expense of increase in failures or unavailability of the server infrastructure. In this work, we explore the impact of data center infrastructure on server availability – we first characterize server component failures with respect to temperature (Cooling System), evaluating the relationship between server hard disk drive failures and temperature in detail. We then evaluate power availability events and their impact on data center power provisioning (Power System). We then focus on the critical management infrastructure that coordinates all of the infrastructure, and propose a novel, low-cost, wireless-based management solution for data center management (Management System). We also present a new class of failures in data centers (Soft Failures), which results in service unavailability, but does not need actual hardware replacements.

APPROVAL SHEET

The dissertation

is submitted in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

SRIRAM SANKAR

AUTHOR

The dissertation has been read and approved by the examining committee:

Sudhanva Gurumurthi

Advisor

Kevin Skadron

Marty Humphrey

Paul F. Reynolds

Kamin Whitehouse

Mircea R. Stan

Accepted for the School of Engineering and Applied Science:

James H. Aylor

Dean, School of Engineering and Applied Science

May

2014

Acknowledgements

I would like to thank my advisor, Professor Sudhanva Gurumurthi, for his guidance and mentorship throughout these years. He motivated me to continue my PhD and supported me throughout this process. Without his encouragement and tireless efforts, this work would not have been possible. He allowed me to research with independence and appreciated my ideas, never dismissing them, providing me with all the support to pursue them successfully. He has taught me a lot and I am fortunate to have him as my research advisor. I would like to also thank my committee who have provided me with feedback and guidance on my work continuously. I would like to thank the Department of Computer Science and the University of Virginia for allowing me to work on my PhD while working in Seattle. I would like to also mention the office staff from the Department of Computer Science and SEAS, who have patiently assisted me with numerous questions on countless occasions.

I have been blessed with great friends, from undergrad in CEG, to University of Virginia and in Microsoft. They have stood by me through peace and war, thick and thin – the Single Singham gang deserves a special mention, they have been my source of strength for the PhD and beyond, and I am looking forward to many years of such friendship. I would also like to specifically mention Microsoft and the support that the company provides to its employees to pursue education goals. I would like to thank my managers throughout this period – Kushagra Vaid, Badriddine Khessib and David Gauthier, who encouraged me and provided me with technical guidance and mentorship.

My parents have always supported me in all my efforts, and I would not be where I am without them. They have supported me throughout my education creating multiple opportunities for me

from primary school to TVS Academy and DAV. From humble beginnings in our native village, Malluppatti, it is a true testament to their vision and dedication that I completed my PhD. My sister is the true doctor, she can treat people and cure them of their ailments, and I am looking forward to her future contributions to the society. My wife has been my constant source of strength and support, she moved cities to help me finish my PhD. I have spent weekends and nights working on my PhD and she has been extremely patient with the PhD journey. She took care of all the other hundred things to do, so I can focus on my work. I am lucky to have her in my life.

Finally, I would like to dedicate this dissertation to my grandparents who have been encouraging and supportive throughout my formative years. I would like to thank all my friends and family who stay closer to me in spirit though separated through geographic distance.

1 TABLE OF CONTENTS

List of Tables

List of Figures

1. INTRODUCTION

Traditional computer applications are transitioning to online services, and user data is being stored increasingly in the cloud. According to an International Data Corporation (IDC) publication, digital data is expected to grow at an exponential pace and the storage needs for cloud computing is projected to double almost every three years [28]. This shift from enterprise software to cloud services is more pronounced than ever, and large cloud providers are building massive data center facilities for coping up with this transition. Even in the HPC domain, these large data centers make it possible for ordinary customers to rent a top fifty supercomputing resource for under $2600/hour or $0.05/core hour [39]. In terms of scale and computing power, most of the large data centers are designed to provide significantly high end computing resource and a large storage capacity in one of the many variants of cloud offerings for end users, including Microsoft's Azure, Amazon's S3 and Rackspace Cloud. Cloud computing enables businesses to start quickly by offering a share of large computing resources to the customer. For the cloud providers themselves, large data centers take up a significant upfront capital investment and hence are a financial liability. In contrast to enterprise software business, cloud computing relies heavily on the efficiency of infrastructure operation for increasing financial profits. Thus, optimizing the capital expenditure and operational logistics of the data center is a valuable contribution in the cloud domain because of their immediate cost savings potential.

The large scale data centers that provide cloud computing services should not be treated just as a collection of servers, but the data center itself resembles a warehouse-scale computer [41]. For instance, a look inside Microsoft's large data center facility at Chicago [65] presents a 700,000

square foot facility with the potential to hold 300,000 servers. The facility also includes 11 diesel generators each supplying 2.8 MW of power, 11 electrical substations and power rooms, 12 chillers each with a capacity of 1260 tons and several network switches and cables. There are several inter-dependent infrastructure elements, including power, cooling, network and management infrastructures, in addition to service personnel, that keep the data center running at optimum levels. As with any large system, such data centers are subject to failures and inefficiencies in each of its component systems. Hence, managing cloud-scale data center infrastructure becomes a challenging task. *This dissertation focuses on characterizing availability issues that can impact server operation in a large scale data center as a result of inefficiencies at critical infrastructure granularities and proposes solutions that can address these issues in a cost-efficient and reliable manner compared to state-of-the-art techniques.*

Data center infrastructure can be broadly classified into server, network, power, cooling and management infrastructures as shown in Figure 1.1. Although the server and network components (termed as IT infrastructure) of a data center are important due to the fact that applications run on server and network infrastructure, the optimal working behavior of the other interacting systems is critical to normal server operation. The power, cooling and the management systems are critical systems that play a very important role in keeping the data center running at optimum levels. Each of these infrastructure elements are deeply linked with each other to ensure that the data center operates at its maximum efficiency. A mismatch in any infrastructure can lead to undesirable consequences including data center unavailability. For instance, a failure in the cooling system might shutdown servers due to high processor temperatures, and might lead to service unavailability. A circuit breaker in the power system might fail and lead to an entire row of racks becoming unavailable. Understanding the interdependency between these systems and server

availability would enable us to address the failure modes that affect server operation more effectively.

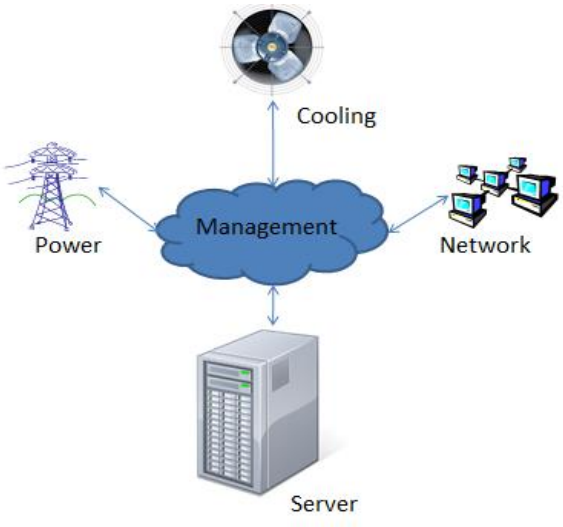


Figure 1.1: Data center infrastructures - Power, Cooling, Network, Server and Management

1.1 DATA CENTER TOTAL COST OF OWNERSHIP

In order to understand the impact of data center infrastructure on server availability, we need to understand the cost metric which is commonly used in data centers. Total Cost of Ownership defines the overall cost that a large enterprise might incur to build and operate a large data center. In addition to capital expenditure costs, the TCO model incorporates the operational costs of maintaining the data center. Hence TCO is a holistic representation of the total cost incurred for a data center. We use the TCO model [40] with the following assumptions: We use a large scale data center with 10MW critical power capacity, and a Power Usage Effectiveness (PUE) of 1.25. PUE refers to the fraction of power consumed by the entire facility including cooling divided by power consumed by IT equipment alone. A PUE closer to 1 denotes a very efficient data center facility

(1.25 PUE is typical of traditional data centers). We use a cost of $0.10c/KWhr for utility power costs. We assume that the total cost of an individual server is $2000 and each server has a typical power draw of 200W to calculate the Server Capital Expenditure. Hence this data center can host a total of 50000 servers. We assume a 5 year server amortization and a 15 year data center amortization for computing the amortization costs (Large companies amortize their capital investment over several years in order to fully realize the value of their investment). We can reconstruct the TCO chart given in this section using the above assumptions. When we reduce amortization of the servers to 3 years, the proportional cost of the TCO contributed by the server would increase. Using the same model, the proportion for server contribution increases from 43% to 61% of TCO. We assume a typical 5 year amortization number for servers in this study.

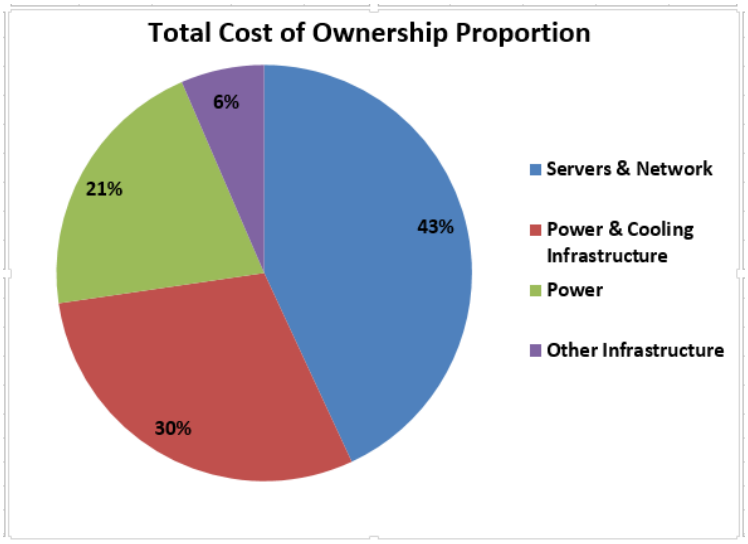


Figure 1.2: Data center Total Cost of Ownership (5 year server & 15 year infrastructure amortization)

As can be seen from Figure 1.2, 43% of the total cost of ownership of a data center is contributed by the actual cost of the server and network. This is desirable, since we want to spend as much as possible in allocating additional servers into the data center. However, ~36% of the

cost is contributed by the critical environment infrastructure. Data center designers aim to reduce this cost and make tradeoffs accordingly, so that they can deploy additional servers into the data center for the same cost. For example, one methodology is to increase the temperature at which the data center operates, and hence reduce the amount of cooling overhead. However, this methodology comes with a consequence – it increases failures in the data center due to the high temperature operation, and makes it necessary to purchase more servers or repair the servers that failed. We need to stock additional hardware components in our supply and maintain a larger team of technicians to replace failing components, thereby increasing the TCO of the data center. Analyzing this tradeoff between cooling and failures is the topic for our first major contribution in this dissertation. Similar tradeoffs exist in other systems including power, management and IT infrastructures. If we reduce cost in one component, it might increase cost in another related infrastructure, and we explore these tradeoffs throughout our contributions in the later sections.

1.2 MAJOR CONTRIBUTIONS

Failures can occur at different granularities in a cloud-scale data center infrastructure. In order to understand the impact of different critical infrastructures on failure modes in data center operation, our work focuses on characterizing failures with the help of case studies from real data centers. As part of this effort, we identify knobs in each component system that has an impact on data center operation. We utilize data collection infrastructures to gather long term data and we correlate this with actual failures from the data center. Specifically, for the impact of cooling infrastructure on server operation, we evaluate a case study of the impact of temperature on hard disk drive failures from a data center hosting more than thousands of servers with 80000 hard disk drives. Hard disk drives are one of the server components with the lowest operating limit with

respect to temperature, and are also one of the dominant failing components in the data center. With respect to the impact of power infrastructure on data center availability, we evaluate case studies from multiple data centers, with special emphasis on power quality events, and its impact on the corresponding redundant power infrastructure design present in data centers today. Using the data collected from this case study, we show that current metrics used for data center design with respect to power availability are not helpful for the cloud paradigm. We propose changes to the power infrastructure to ensure reliability and at the same time optimize cost without impacting server availability. Following this, we identify the limitations of traditional management solutions and propose that commodity wireless devices be used in cloud-scale infrastructures, providing a flexible management topology. We address protocol design for implementing a time-critical and mission-critical functionality (power capping) in data centers. We show that wireless data center management solution has an order of magnitude lower cost than existing management solutions.

To summarize, our major contributions from this dissertation are:

1. **Impact of Temperature on Hard Disk Reliability:** Previous work does not establish a clear relationship between temperature and failures. In our first contribution, we focus on data center temperature, and analyze the impact on hard disk drive failures in large data centers, with a real data center study of close to 80000 disk drives. We show that average temperature does have correlation to failures, and that variations in temperature or workload show minimal correlation to failures. We also present an Arrhenius model for hard disk reliability (Cooling System Impact). This work has been published in DSN 2011 [78] and TOS 2013 [80].
2. **Power Availability Provisioning:** In our next contribution, we focus on power infrastructures and highlight a tradeoff between power capacity utilization, power

availability and performance targets in a data center. We propose a workload-driven approach to provisioning redundant power equipment in data centers and term this new approach as *Power Availability Provisioning*. We characterize power availability data from two data centers, and present application-driven performance and power models to enable N-M redundancy mode, which is a lower-cost, unique provisioning methodology in data centers. This work has been accepted for publication in CF 2014 [78].

3. **Wireless Data Center Management (Wireless DCM):** Our next contribution deals with the management infrastructure in data centers. We propose a novel wireless-based, low-cost management solution, which also satisfies the performance and reliability requirements of data center management functions. We chose a time-critical DCM application (power capping) and present a protocol design and implementation (CapNet) that can achieve time sensitive power capping functionality. We show that our solution has potential for 12X-18X (an order of magnitude) improvement in cost over existing management solutions.
4. **New Failure Modes - Soft Failures:** While traditional component failures exist in the data centers, over the course of our work, we discovered a new class of problems in the data centers. These failures caused service disruptions, resulting in a physical touch, but no actual component failed in the system. We term these events to be “soft failures”, since they caused an actual failure, but did not result in a hardware replacement in the data center. We characterized these failures, analyzing occurrence patterns and present possible approaches to solve this new problem in data centers. This work was published in CAL 2013 [81].

The organization of the rest of the dissertation is as follows. The next chapter provides a brief overview of data centers focusing on the critical infrastructure systems. Chapter 3 presents the impact of temperature on hard disk drive failures, while Chapter 4 presents the power availability provisioning methodology. Chapter 5 discusses the wireless data center management solution and Chapter 6 presents Soft Failures, a new trend in data center availability. Chapter 7 concludes this dissertation.

2 OVERVIEW OF DATA CENTER INFRASTRUCTURE

This section briefly describes each infrastructure component in order to provide background information on the systems that impact data center operation and server availability.

2.1 COOLING INFRASTRUCTURE

A traditional data center cooling infrastructure consists of several CRAC (Computer Room Air Conditioner) units or CRAH (Computer Room Air Handler) units. CRAC units are refrigerant based and are connected to condensing units outside the building, whereas CRAH units use chilled water. These systems are responsible for moving air through the data center typically by the use of large fans. There are also humidifiers and chillers that support this operation. Figure 2.1 presents a cooling solution based on CRAC units. The traditional cooling philosophy was to maintain a consistent temperature inside the data center by having stable chilled water loops and predetermined data center setpoint temperature. The air is circulated through the floor plenum where it is cooled by the water loop. This air then passes through the floor vents, where it is routed through the servers by server fans and hot air rises up to the return plenum. More recently, data centers are moving to an economical cooling option based on adiabatic cooling [35]. The underlying principle is that data centers can be cooled by outside air for most parts of the year, but for the few hours during which external air gets warmer, evaporative cooling is used through the air flow path to cool the incoming air. This methodology still uses air handlers and fans to move air through the data center.

Cooling infrastructure efficiency determines power drawn by the cooling equipment, and directly factors into PUE (Power Usage Effectiveness) of the data center (PUE is defined as the

ratio between power consumed by the data center to the power consumed by the IT infrastructure alone). Cooling also has a direct impact on server components, since each server component has temperature specifications within which it operates as expected. Outside of the specification, the behavior of the component is not guaranteed and it might even fail or shutdown causing server unavailability. In our contributions (Chapter 3), we identify hard disk drives as one of the dominant failing components, and measure the impact of temperature on HDD failures in a large data center facility.

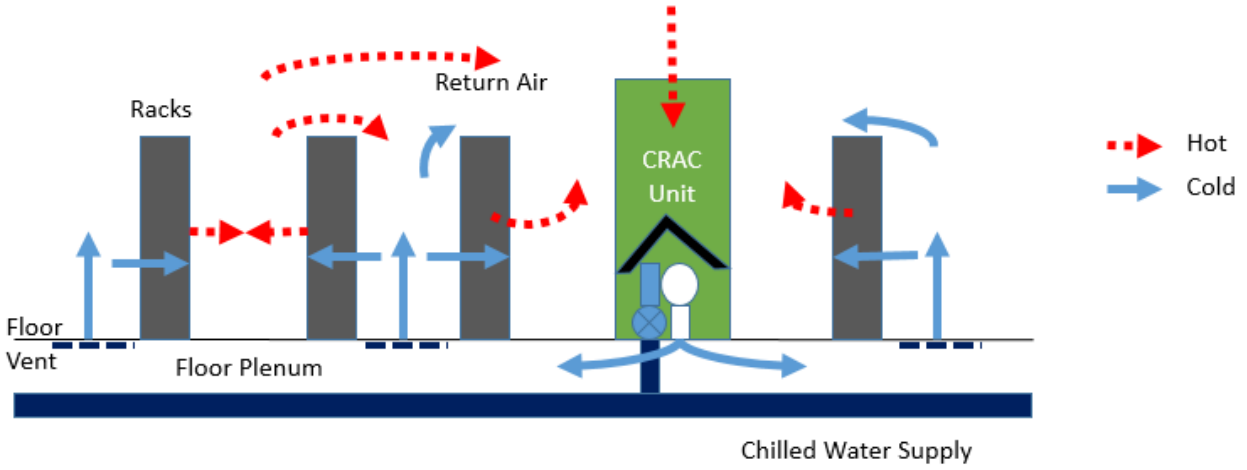


Figure 2.1: Traditional data center cooling system based on CRAC units

2.2 POWER INFRASTRUCTURE

Power infrastructure is an essential piece of data center design, and the power budget determines the number of servers that can be housed within a data center. Data center power consumption is also becoming a growing concern for several reasons, including the fact that this will be close to 2% of the entire electricity consumption of the world [58]. The power infrastructure needs to support the critical power capacity of the data center, and does this with the help of several

power infrastructure components including Electrical Substations, Transformers, Diesel Generators, Uninterruptible Power Supplies (UPS), Automatic Transfer Switches, Remote Power Panels, and the Power Cords running to each server. Figure 2.2 shows a typical power infrastructure layout in data centers. Any event impacting power delivery is crucial since it can impact the server operation directly. A typical data center consists of two kinds of power sources: 1. Utility source (Primary) and 2. Diesel generator source (Alternate), which acts as secondary backup in the event that Utility source has a power failure. The automatic transfer switch is used to switch between these two sources. However, it takes a short duration of time (10s of seconds) for the diesel generators to start up. Hence, battery powered UPS are used to bridge this gap between the transfer from utility to diesel generators. Several remote power panels are connected to the UPS devices, and transfer power to the actual racks through PDUs (Power Distribution Units). The servers themselves get power through the PSUs (Power Supply Units).

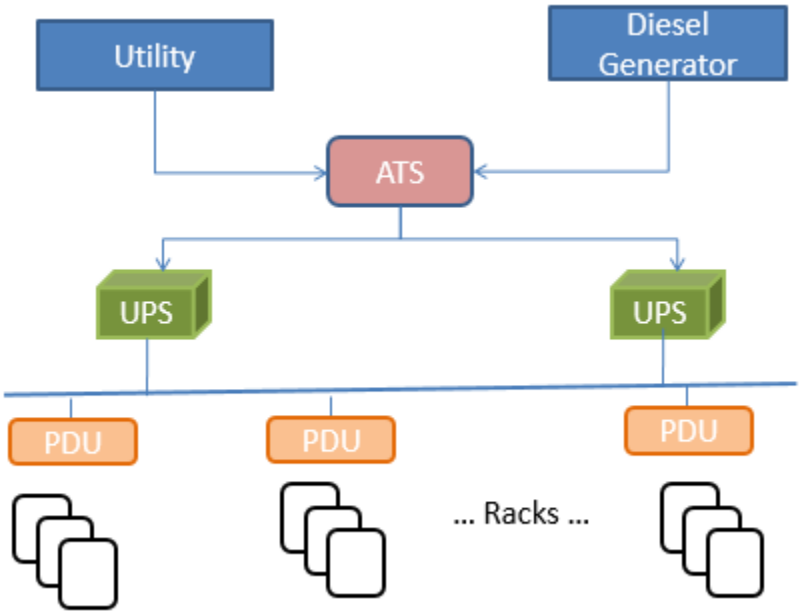


Figure 2.2: Typical power infrastructure in data centers

Within this complex power hierarchy in the data centers, there are several events that impact power quality and also eventual server operation. For instance, the Utility or the Diesel Generator can experience failures, potentially leading to the entire data center experiencing a blackout. The UPS could fail, thereby leading to a scenario where there is no switching mechanism, and leading to a power interruption of short duration. There are also sags and swells, which are brief reduction or increase in voltage typically lasting few cycles to a second, but still significant in the data center since it can impact server operation, unless the UPS can take over. There could also be transient voltage spikes that can occur due to external machinery operating on the same supply line or also from the utility supply itself. These events are typically filtered from the servers through the transformers and the UPS circuit. However, UPS and diesel generator costs are steep, and most data centers that can tolerate failures are moving to a model where they eliminate the diesel generators, and move to smaller UPS enabling better fault isolation. In our work (Chapter 4), we evaluate power events with real data and propose a novel power provisioning approach that is cost-efficient.

2.3 MANAGEMENT INFRASTRUCTURE

Traditional server management approaches in the data center can be classified into two types: 1) management through the primary network path and 2) management through an out-of-band management network. Figure 2.3 shows both the options for management in a data center. Server management can be performed through the network by issuing commands to a management agent running on the server. The primary network path is a cost efficient approach, since it does not need any additional infrastructure added for management purposes. This management approach might rely on the operating system for management functionalities (in-band) or use an external path

inside the chip to communicate with the baseboard management controller (side-band). However, the major drawback of using the primary network path is the reliance of management commands on network connectivity. If there is a failure in the network path to the server, then there is no management path to that server. This is highly undesirable, since the management system does not know if the server itself is not functional or if the network path is down. In addition to management unavailability, such failures result in a service ticket being issued, and an actual technician identifying the root cause of such failures, which prevents this from being a fully automated and reliable mode of operation. Though this approach incurs minimal capital expense, the operational expenses due to link failures could turn out to be really costly.

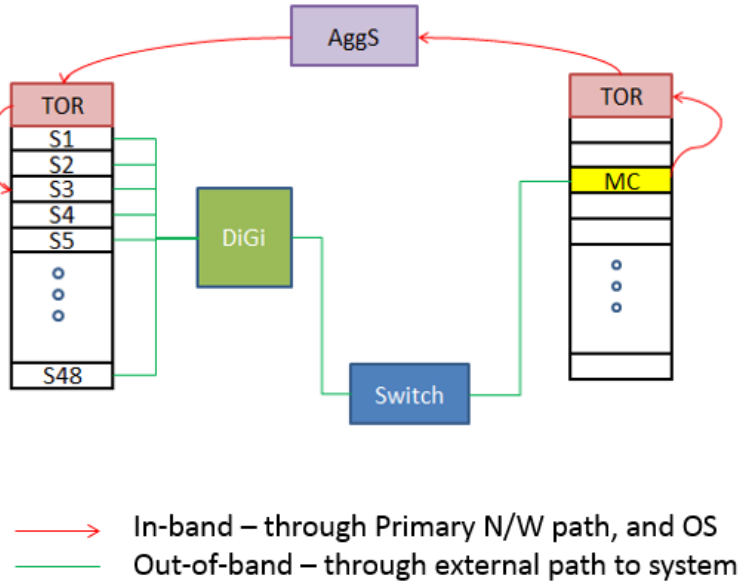


Figure 2.3: Management infrastructure in data centers (MC - Management controller, S1- S48 are servers, AggS - Aggregation Switch and TOR - Top Of Rack switch)

On the other hand, a separate out-of-band management system needs extra capital expense to be installed, but it provides a completely distinct path to control the server, even when the network link is down. This solution allows for BIOS setting manipulation, reinstallation of operating systems, and managing boot options. This methodology provides an alternate

management path, and it is a necessary solution in data centers, due to the mission-critical nature of operation for these large facilities. In our work (Chapter 5), we illustrate that traditional out-of-band management solutions in data centers are expensive and difficult to scale, and propose a flexible, order of magnitude lower cost solution based on wireless management in data centers.

2.4 IT Systems – Server & Network Infrastructure

Data centers rely on their network and server infrastructure to support the large volume of customer requests coming into the data center and the data responses that go out of the data center. In addition, to perform typical cloud computing jobs, there is a significant amount of network communication between servers even within the data center. Together, the servers and network are crucial for the applications to run.

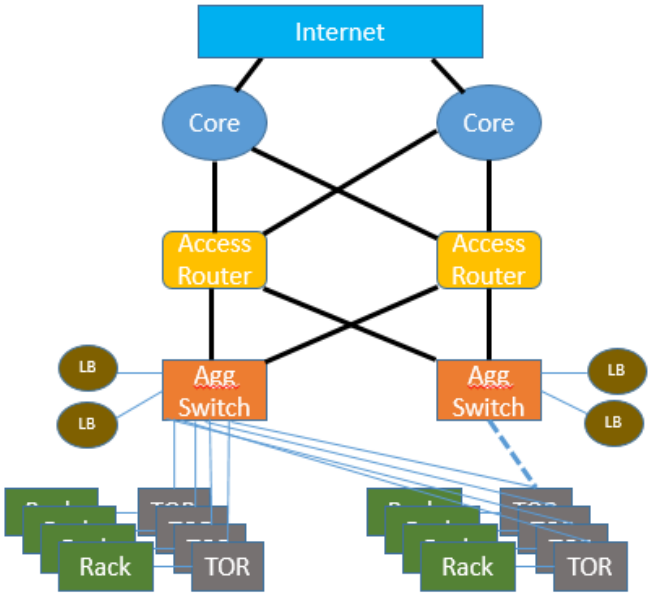


Figure 2.4: Typical data center network with racks

A typical data center network consists of top of the rack (TOR) switches at its lowest layer. The TOR is connected through Ethernet cables to the servers. These TORs are connected to

aggregation switches (Aggs), which aggregate traffic from the rack level. The Aggs then connect to several Access routers that aggregate traffic from several thousands of servers, and send it to Core switches which connect to outside internet. Failures in the network infrastructure typically affect the server operation, since communication to any server is done through this infrastructure. Network failures in a data center are addressed in previous work, where the authors provide a comprehensive overview of the different failure types and scenarios in a large data center [30]. It should be noted however, that for an external agent, network failures and server failures look alike, since if an agent is not able to reach the server, it doesn't know whether the link to the server failed or the server itself failed. Hence this becomes important to the management architecture. In general, network failures are of two types, link failures (between server and TOR, or between TOR and Aggs, etc) and device failures (TOR failure, Aggs failure, etc). Both these failure types affect communication with the server, and hence there is already certain level of redundancy designed within the system. Server design is a significant consideration in the design of the data center, and we provide detailed insights into online services server engineering in our previous work on this subject in an IEEE Micro article [59]. Detailed treatment of these two infrastructure designs itself is outside the scope of our dissertation, however we refer to existing previous work across our contributions wherever it is relevant, and illustrate knobs from this infrastructure that can help optimize data center systems holistically.

2.5 RELATED WORK

Temperature and Hard Disk Failures: Server component failures and reliability are yet to be understood completely. Previous research in this field has generated conflicting results, especially in relation to subjects like the impact of temperature on disk drive failures. With respect to large

scale installations, Gray et al [34] observed failure rates ranging from 3.3-6% in two large web properties at Microsoft. Schwartz et al [88] report failure rates of 2-6% in the drive population at the Internet Archive. Elerath et al [20] report that end-user failure rates can be as much as ten times higher than what the drive manufacturer might expect in their study on server class disk drives. Schroeder et al [86] find that in the field, annual disk replacement rates typically exceed 1%, with 2-4% common and up to 13% observed on some systems. The authors also present interesting per-component failure percentages for three different types of systems that they considered. They also report a significant overestimation of mean time to failure by manufacturers. Schroeder et al [84] in their study of failures in petascale computers, review sources of failure information for compute clusters and storage systems, and project corresponding failure rates. Others explore the tradeoffs between workload characteristics and temperature with the help of simulation [55], but do not consider reliability impacts. Our work considers the inter-relationship between workload, temperature and disk drive reliability.

One of the most closely related works to this study is by Pinheiro et al [71], which identified correlation between disk errors and SMART attributes from a large population of serial and parallel ATA drives. This work also concluded that temperature and activity levels had less correlation to disk failures and was a surprising result when compared to previous studies [14][98]. Recently, El-Sayed et al [21] show that temperature correlation to failures are weaker than expected in a diverse population of disks, and point out there might be other factors that are more dominant than temperature, whereas we try to eliminate the impact of diverse factors by selecting controlled environments. Yang et al [98] establish that a 25 C delta in temperature derates the MTTF by a factor of 2 in their study on Quantum hard disk drives. Cole et al [14] from Seagate, present thermal de-rating models showing that MTTF could degrade by close to 50% when going

from operating temperatures of 30C to 42C. Our model based on data center study matches the observations made by Cole. Our measured failure rates also exceed the AFR rates that manufacturers mention in their datasheets. Also interestingly, Vishwanath et al [93] report no correlation between failures and location of servers within a rack. We find in our case study in Section 3 that temperature does have a strong correlation to failures (within chassis, racks and across racks). We propose that temperature impacts for data center scale environments should be factored in knowing the server configuration and data center inlet temperature range.

Power Availability Provisioning: In contrast to power consumption management, provisioning is an activity that is typically performed before the servers are installed in the data center. There are several power management works that deal with shutting down servers, hibernating using power states, DVFS, and using intelligent workload migration techniques. In the realm of power provisioning itself, previous work [23][24][31][33][74] focused mainly on power capacity provisioning and rarely looked at power availability as a function of adding additional capacity to the data center. For instance, CPU utilization correlation to power utilization was shown by Fan et al. [23], and they also presented a provisioning methodology to allocate power capacity based on workload power usage. Power management at the level of blade ensembles was discussed by Ranganathan et al [74]. Recently, researchers look at software techniques to provision data center infrastructures [95], which leverage battery storage and application throttling techniques to implement provisioning. While these provisioning methodologies are orthogonal, there is very little work in understanding the relationships among availability, power capacity, and data center performance. To the best of our knowledge, our work is one of the first to propose a workload-driven power availability provisioning methodology to reduce the cost incurred in power infrastructures that also are provisioned for availability.

Wireless Data Center Management: Data center management infrastructure itself has not been a topic of considerable research previously. However, protocols for management have been implemented, including Intelligent Platform Management Interface (IPMI) [52], which is the interface standardization defined for platform management in servers. Specifically for datacenters, Intel and Microsoft also developed DCMI (Datacenter Management Interface) [15] which is a subset of IPMI and is compatible with IPMI specifications. There are several implementations of IPMI, including HP's Integrated Lights-Out [43], Dell's DRAC [4] and IBM's Remote Supervisor Adapter [46]. There are also processor side implementations of IPMI including AMI's MegaRAC [62] and Intel's Active management technology [50]. In addition, the management functionalities have been well represented. For instance, power capping is a functionality that is used to reduce the capital spending on data centers, and enterprise data centers use an over-subscription approach as studied in [23][27][61][70], which is similar to over-booking in airline reservations. Server vendors and data center solutions providers have started to offer power capping solutions [42][50]. Power capping using online self-tuning methodology [31] and feedback control algorithms [97] has been studied for individual servers. In contrast, our work concentrates on using power capping as a higher level functionality that is achieved by innovating on the infrastructure design itself. Power capping has been studied before [25][57][61][75][94][96][100], and all existing solutions rely on wired network for controller-server communication. In contrast, we focus on wireless management solution for power capping in Chapter 5.

Previous work on using wireless network in data centers exists, for example, on applications needing high bandwidth (e.g. with 60GHz radio) production data network [101]. In contrast, our work is targeted at data management functions that typically have lower bandwidth requirement while demanding real-time communication through racks. RACNet [60] is a passive

monitoring solution in the data center that monitors temperature or humidity across racks where all radios are mounted at the top of the rack. Our solution enables active control and requires communication through racks and server enclosures, and hence encounters fundamentally different challenges. RACNet also does not have real-time features. In contrast, our solution is designed to meet the real-time requirements in power capping. We believe that our work is one of the first to present an active control solution, in contrast to passive monitoring solutions that has been prevalent in this space.

Soft Failures: Most of the previous studies in the failure domain only look at actual hardware replacements to compute failure rates. For instance, a study on disk failures in large scale infrastructure at Google looks specifically only at failed disk drives [71], and a recent study at Microsoft also looks only at hardware replacements [78]. Other large scale studies that look at hardware replacements include work on memory failures at Google [87], and failures at HPC clusters [84]. Hardware reliability for data centers was studied in [93] with emphasis on failure trends and the authors note successive failure probabilities, but do not investigate transient failures in depth. Transient failures are not an unknown issue at microprocessor scale [66]. However, such a phenomenon is not typically being looked at as a systemic issue in large scale data centers. Our work illustrates this new trend, and we describe patterns and possible approaches to handle this trend in Chapter 6.

3 COOLING INFRASTRUCTURE - IMPACT OF TEMPERATURE ON HARD DISK DRIVE RELIABILITY

Server components are typically composed of commodity electrical and mechanical parts, and hence they are prone to failures. Frequent failures reduce infrastructure availability and increase the cost of data center operations. In addition, the server design in itself could be a major catalyst for a large portion of the server component failures. For instance, we found that a particular drive location in a dense storage configuration, under a fairly constant workload was continuously exposed to high temperature conditions, even under nominal inlet temperature to the server. We found a higher number of drives in this location failing more often, thereby showing strong correlation to operating conditions. Understanding the reason behind such failures enabled us to address design issues, thereby increasing the availability of machines for the particular online service. Availability of online services is a key differentiator in today's competitive market. Higher server reliability ensures that online services can have increased availability. Increasing the number of available servers also delays the need for provisioning new server deployments in data centers. New server deployments have a longer delay cycle, and might cause a high impact launch to be delayed, thereby causing significant financial damage to the enterprise. Hence, having more servers that are readily available benefits the financials of a large enterprise.

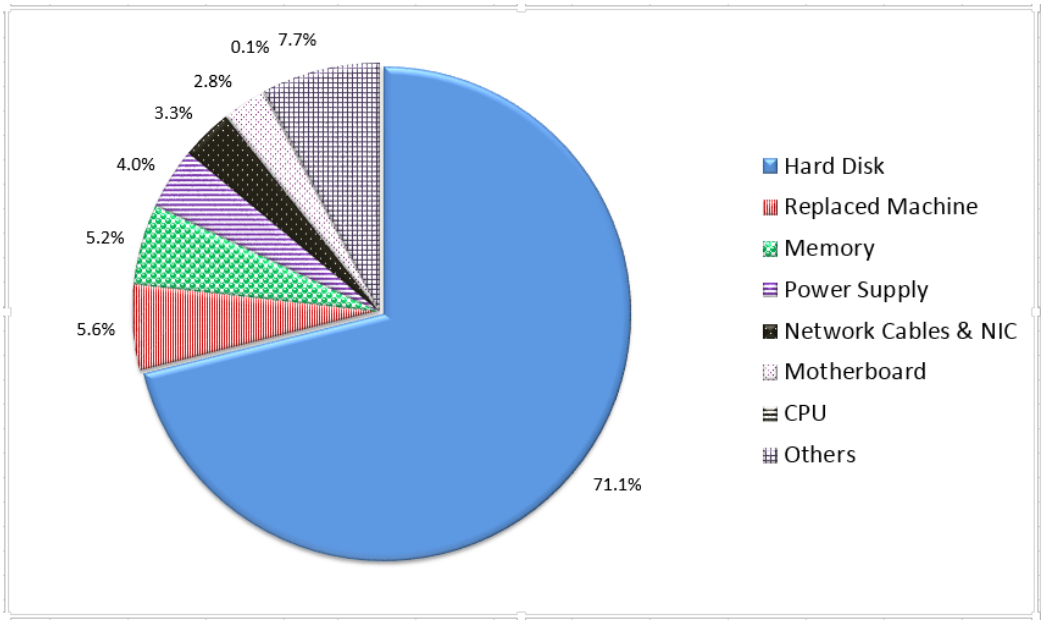


Figure 3.1: Breakdown of hardware component errors in a large data center (2 years failure data)

Server component failures have indeed been recognized as important and prior works have studied individual component reliability, such as for hard disks [71][86] and memory [87]. Figure 3.1 presents actual data on the different kinds of failure types that result in component replacements, observed over a period of two years from typical large-scale data centers housing more than 100,000 servers. We see clearly that hard disk drives account for 71% of the known failures, making them the dominant failing part. This is in part due to the mechanical moving parts of the disk drives and in part due to the extensive use of commodity SATA drives in large deployments. SATA disk drives are known for failing more often than SAS drives, but are also cheaper for storage capacity per dollar [44]. At the time of this study [79], SSDs did not contribute to a large proportion of failure rates due to the introductory state of such technology in data centers. Memory failures constitute about 5.2% of the total failures, including configuration errors. The actual percentage of memory that was replaced to correct memory failures was 4%. This is close to numbers reported by Schroeder et al [87]. Some errors like Network cable errors and NIC issues are also worth noting. It shows that hardware and network configuration issues do result in service

issues on the data center floor. The ‘others’ category includes storage/power backplane issues, wrong wiring, and other issues that do not fit into the major component buckets. Understanding dominant server component failures is critical to choosing the correct characterization work in our study.

In this section, we establish the different aspects of correlation between temperature and disk drive failures observed from the large data center case study. In addition to temperature impact at different granularities, our work quantitatively evaluates the impact of variations in temperature as measured in a live production environment. We also explore whether workload variations cause temperature behavior to be impacted, and also if workload intensity has correlation to failures observed in the data center. We conduct experimental studies to validate our observations from real data.

In summary, our major contributions in this chapter are:

1. We show strong correlation between temperature observed at different location granularities and failures. Specifically, we establish correlation between temperatures and failures observed at the following location granularities: a) inside drive locations in a server chassis, b) across server locations in a rack and c) across multiple racks in a data center.
2. Although average temperature shows a correlation to disk failures, we show that variations in temperature or workload changes do not show significant correlation to failures observed in drive locations.
3. We corroborate our findings from the data center study through an experimental evaluation and show that chassis design knobs (disk placement, fan speeds) have a larger impact on disk temperature than tuning workload knobs (intensity, different workload patterns).

4. With the help of Arrhenius equation-based temperature models and the data center cost model, we quantify the proposed benefits of temperature optimizations and increased hard disk drive reliability. We show that data center temperature control has a significant cost advantage over increased fan speeds.

3.1 EXPERIMENTAL INFRASTRUCTURE

We conduct real data center measurements and also experiments in this study, and hence it is important to understand the differences between multiple infrastructures that we use. For instance, we use large scale data centers housing several thousands of servers, and several tens of thousands of hard disk drives. We collect data from these large facilities to identify patterns from them. In addition, we have an experimental facility that houses a controlled set of servers and equipment needed to control temperature. We describe this in more detail in this section.

3.1.1 Temperature Measurement Infrastructure

We perform our data measurements on a population measuring thousands of servers and ten of thousands of hard disk drives. All the servers in this study are identical, with dual CPUs and an additional storage enclosure containing up to 40 SATA drives in a RAID 1+0 configuration. In our server chassis, we are able to fit 5 disk drive columns (3.5” SATA HDD) across the length of the server. The traditional data center racks have a cold aisle from which cold air is pulled across the server and exhausted out in the hot aisle [41]. Hence the air gets preheated by the time it reaches the interior hard disk drives and leads to higher temperatures for those hard disk drives.

The temperature measurements are collected by SMART counters (counters monitored as part of every disk drive’s self-monitoring facility) from a sensor included in the HDD enclosure in every hard disk drive. The SMART counters for temperature are logged every 20 minutes by the

array controller at a local controller log along with several other SMART counters. Every day this local server log is shipped to a central server and archived. Since the population is in a live production environment and there are various data that are collected, the duration of sampling is limited to 20 minutes on account of data storage limitations.

3.1.2 Server Test Configuration

For evaluating the impact of server chassis design parameters including placement of disk drives and fan speeds, we use a dense storage disk enclosure along with a standard enterprise server (Leading manufacturers have similar configurations [45]). This dense storage enclosure is setup to

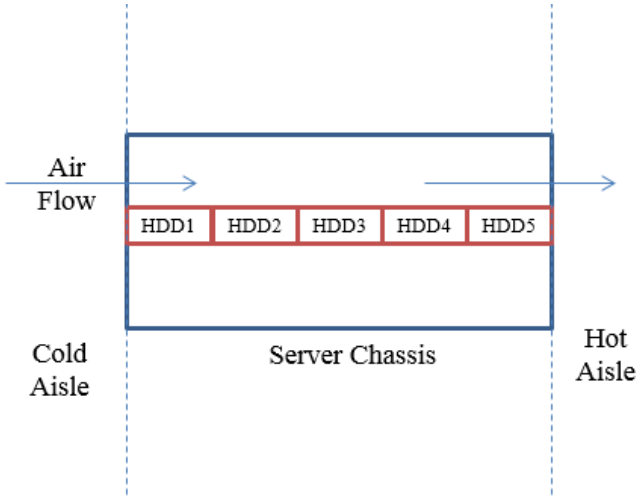


Figure 3.2: Dense Storage configuration and layout

mimic the actual production setup as close as possible. However, this does not directly reflect any production storage configurations for proprietary reasons. The test server has a controller that can log instantaneous temperature at each disk drive. The storage enclosure has five columns of hard disk drives arranged from right to left as shown in Figure 3.2. For this enterprise configuration, there are 34 disk drives present in an enclosure that can hold up to 35 disk drives. This presents an opportunity on which disk bay to leave empty, and we explore this tradeoff in later sections. The

logical drive is made up of all the 34 disk drives in a RAID 1+0 configuration that is typical of an enterprise RAID setup.

3.1.3 Workloads

For analyzing workload behavior and its resultant impact on varying temperatures, we use real data center storage workloads obtained from trace characterization. Enterprise storage systems are designed to support multiple concurrent users in order to amortize the cost of data access over a large number of users. Hence enterprise workloads are typically composed of random IO operations, with high inter-arrival rates. Table 3.1 shows the four workloads that we consider being representative of large scale data center workloads.

Table 3.1: Data center workload characteristics – Random access with short inter-arrival times.

Workload	Rd:Wr Ratio	Random %	Dominant Block Size	Average Inter-arrival (ms)
Email	1.4	83%	8K	1.48
UserContent	7.0	91%	4K	22.22
Exchange	2.0	66%	32K	0.71
Messenger	9.6	99%	8K	0.30

A denser storage solution typically acts as backend storage for applications that require large amount of data storage, like Email and OLTP applications, since denser solutions makes it possible to pack more storage in lesser space. Hence for testing such a high density storage solution, we use storage profiles of an Email backend server (Email), a large scale file system server at Microsoft (UserContent), Exchange server (Exchange) and an OLTP backend profile (Messenger) that represents user meta-data for a large online service. The trace characterization framework is based on ETW (Event Tracing for Windows) [69] and it captures the disk IO events at the OS level. This ensures that if we design a system that is configured similarly, regenerating IOs as

captured during the trace will be truly representative of a data center workload. We use publicly available disk IO generator like IOMeter [51] to replay the workload for our experiments.

From the table, we observe that all workloads have short inter-arrival times. UserContent is a file server workload with minimal storage requests, and has a larger inter-arrival time of 22.2 milliseconds between IO requests. Note that the other applications including Email, Exchange and Messenger (OLTP) workloads have less than 2 milliseconds between each IO request. Also, note that all these workloads are mostly random (66%-99%). Random IO requests require disks to seek to particular locations on the disk drive, consuming additional power as a result, and hence could result in possible increase in temperature. Since most of these workloads are random, the disk drives are continuously performing seek activity and they do not have time to shut down or save power and inter-arrival times are relatively short. In addition, seek activity is composed mainly of short-distance seeks [55] for enterprise workloads and hence there is minimal impact on temperature. We use this observation in the later sections to motivate our experimental evaluation to select different knobs that have an impact on temperature.

3.2 REAL DATA CENTER CASE STUDY

In this section, we present a case study with data collected from a live data center facility. We analyze the major determinants of hard disk failures, and explore temperature correlation in depth. The hard disk drive failures that are considered here denote actual hardware replacements as viewed from the data center management perspective. Detailed failure analysis that can identify sub-component errors or false positives (similar to manufacturer lab analysis) is not typical in such high security environments. Throughout this section, we define failures as events leading to system

downtime that was fixed by a replacement of the component in the data center floor except when specified.

3.2.1 Hard Disk Drive Failure Determinants

There are a number of factors that can influence hard disk drive failures, including age of the disk drive, utilization on the hard disk drive (general wear and tear due to use), temperature of operation, and vibration.

3.2.1.1 Age of the Disk Drive

Several previous studies have established different failure rates with respect to the age of the disk drive population [71]. A typical failure curve across age resembles a Weibull bathtub shaped curve with a large number of infant mortality, stable mid-life curve and steady increase in failures again at older age. In our study, most of the disk drives are of similar age since all the servers were deployed around similar timeframe when the data center became operational, and are past the infant mortality stage. Hence the age factor does not become a major determinant for our study. This is beneficial to help isolate the impact of other factors on failure rates in data centers.

3.2.1.2 Vibration and SMART Monitors

Dense storage can cause significant vibration; however modern hard disk drives balance internal vibration through vibration compensation techniques in the servo mechanism of the hard disk drives [36]. Vibration impact on failures is a topic of future work. We do collect several SMART data from the disk drive population, including Reallocated Sector count, Seek errors, Spin up time, ECC errors, Temperature etc. Though we see SMART counters being indicative of some failures, a predictive methodology is hard to obtain. For one of our large populations, such a methodology would have been able to account for less than 20% of all disk failures. Previous

conclusions made by Pinheiro et al. [71] also suggest that other SMART counters do not provide a confident way of predicting hard disk drive failures.

3.2.1.3 Utilization vs Temperature

The remaining two significant failure determinants are disk utilization and temperature. We need to isolate the impact of these two metrics that are location dependent. One of the primary factors that can cause more wear on the hard disk drive is disk utilization (we use utilization as a proxy for workload duty cycle), which denotes the amount of activity on the hard disk drive. According to the volume and data layout, certain disks might be more stressed than other disks (for instance, a data volume in SQL might have higher level of activity than a Backup volume). We conducted a preliminary investigation to determine which of these two metrics is correlated closely to hard disk drive failures.

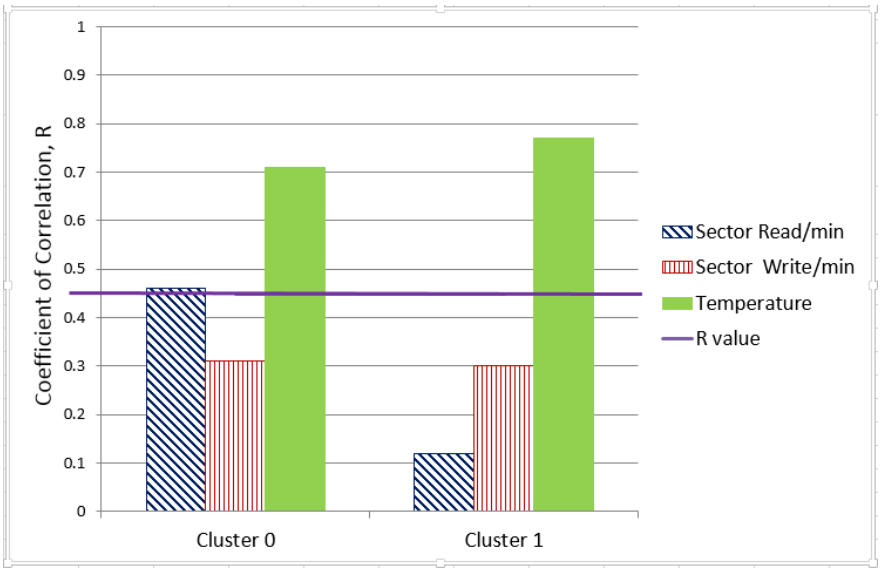


Figure 3.3: Temperature shows better correlation to HDD failures than workload utilization

Figure 3.3 presents the results of the analysis on a total of 10000 hard disk drives spread across two clusters. We correlated the 'sectors read/ minute' and 'sectors write/ minute' experienced by the disk drive in a particular location as seen by the controller over its entire lifetime, to the failures observed in that location over a year. On the other hand, we also correlated

the temperature observed in those disk locations to the number of failures. We plot the resulting coefficient of correlation, which shows that the read and write activity on the disk drives correlate minimally with the failures. However, drive temperature inside the chassis shows stronger correlation to disk failures in the particular location within the chassis (R value for temperature is above the critical R value line, at df=30 for a two-tailed test at level of significance = 0.01). Hence for the remainder of the work, we concentrate on disk drive temperature and do an in-depth temperature measurement and correlation analysis across disk drive locations inside chassis, location of a server within a rack and locations of racks in a data center.

3.2.2 Correlation of Disk Failures with Average Temperature

We present a case study where specific data center design parameters and a dense storage chassis design resulted in higher number of disk failures, under high operating temperature. The

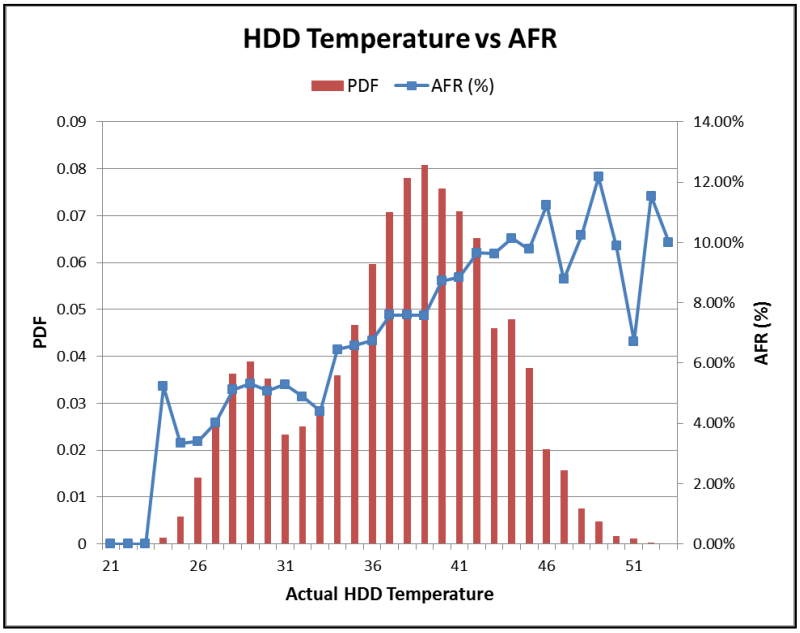


Figure 3.4: Failure rates at different hard disk drive temperatures

case study was conducted in a raised-floor data center, containing tens of thousands of hard disk drives in a dense storage server and failure data was collected for a period of 1 year.

The result of our study is surprising since earlier studies [71] establish that disk drive failures do not increase with increase in temperature in the field. Figure 3.4 shows the actual HDD temperature in increments of one degree and the corresponding AFR for our entire population. We see clearly that with increase in HDD temperature, the AFR rate increases. There are some data points at the end of the spectrum that have smaller number of samples and hence a higher skew. For the major part of the distribution (shown by PDF columns), we see that AFR steadily increases as HDD temperature increases. Interestingly, we found that certain disk locations in the heavy storage enclosure were exposed to high temperature for a longer duration even under nominal inlet operation temperatures. We also observed a significant difference between the inlet temperatures measured at different locations in the data center. In the next section, we present our analysis and observations categorized by location granularity. We divide our correlation analysis into three distinct temperature impact zones: Drive locations inside the server chassis; Server locations within a rack and Rack locations across the data center. There are different factors that come into play for each of these temperature zones. We shall discuss each in more detail in the following sections.

3.2.2.1 Correlation inside the Server Chassis

Server design is an important factor in determining the availability of machines in a data center. Depending on the placement of the hard disk drives, there could be significant variation in drive temperature. This is especially true in the case of dense storage, since cold air flows from the front of the storage enclosure to the back. Given that the workload running on the disk drives are similar (no significant duty cycle variations), we can establish the correlation if there are more failures for drives which experienced higher operating temperatures. We present the layout of a dense storage device in Figure 3.2 that was used in our case study. There are five hard disk drives

columns where HDDs are arranged one behind the other from the front of the enclosure to the back. Hence the air gets preheated by the time it reaches the interior hard disk drives and leads to high temperatures for those drives. This results in an increase in number of failures observed in that location.

Figure 3.5 (a) shows the average temperature observed in each hard disk drive column (1 through 5) across all the machines under this study. Note that the temperatures increase from 27 C in the front-most hard disk drive (HDD1) to 39 C in the fourth hard disk drive column (HDD4). This is just the average temperature measurement, and there were hard disk drives that were at temperatures greater than 45 C in hotter parts of the data center as shown in Figure 3.4. The last drive (HDD5) closer to the hot aisle has a reduced temperature due to heat dissipation at the outlet. The corresponding total failures observed across the entire server population over a period of 1 year are denoted by the AFR line. Note that we present Annual Failure Percent (which is a measured population-based value and should not be considered as the Annualized Failure Rate, which is a calculated metric that manufacturers provide) for our population that is on continuous mode of operation throughout the year (For a discussion on different annual failure rates, please see [20]). The failure rates measured here are not reflective of manufacturer quoted rates, and should be considered only as number of failures out of the population under deployment. Out of the hard disk drives that were in the front-most part of the server chassis (HDD1), only 4% failed, whereas, for the fourth hard disk drive (HDD4) around 6% of the total disks failed. This is almost 1.5X the number of failures compared to the front of the chassis. This result shows a strong correlation between temperatures observed through the SMART logs collected at the machines and the observed failures reported in this data center. In fact, the correlation coefficient measured across the entire population for (*average temperature for drive locations inside the chassis,*

number of failures) pair is **R = 0.79**, which is significantly high. Our experience with this dataset does point out that lower temperature locations do have lower failures, and as system designers it is a strong motivation for reducing temperature impact inside a chassis design. Fan speed and airflow management helps reduce such temperature impact.

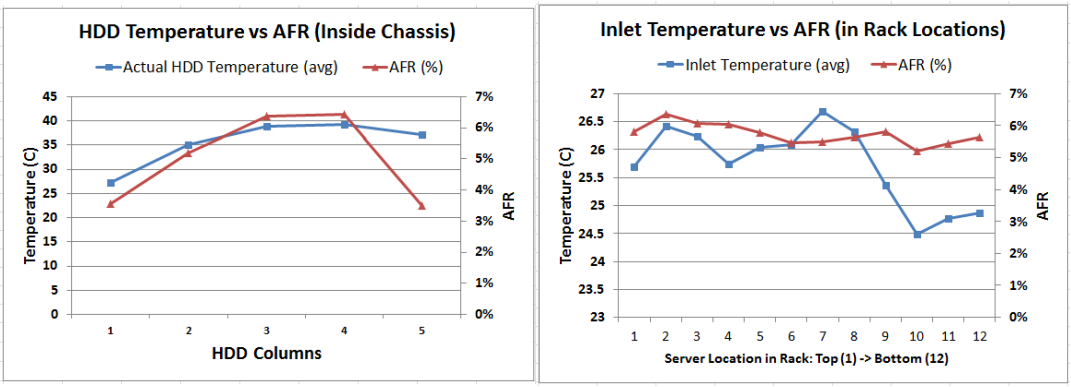
Observation: ***There is a significant correlation (r = 0.79) between actual hard drive temperature inside a server chassis design and the number of drive failures. Hence chassis design should incorporate temperature reduction optimizations.***

3.2.2.2 Correlation across Servers in a rack

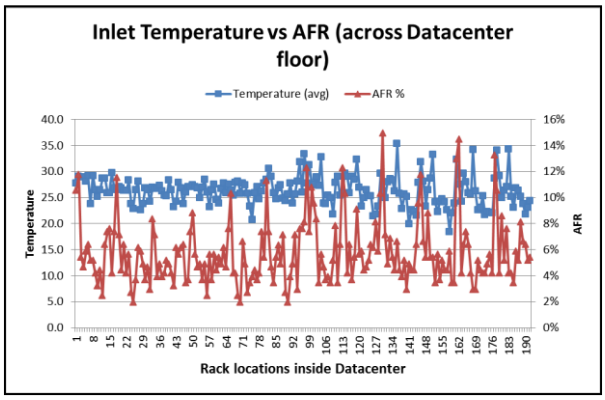
A data center rack consists of multiple server chassis arranged on top of each other. The cool air comes through vents closer to the bottom of the rack and rises upwards. It is pulled across the server as it rises up in a horizontal direction (as shown in Figure 3.2). However as it moves up through the vertical direction, there is an increase in air temperature due to heat dissipation. There are also other mechanical impacts such as the differences in air pressure (cfm) at different server locations within a rack. In this section we explore whether the server location and inlet temperature observed at each location correlates with the number of disk failures observed at that server location.

From Figure 3.5 (b), we see that the cooler servers (Location 9, 10, 11, 12) that are on the bottom of the rack have lesser number of failures (closer to 5%) as compared to hotter servers (Location 2) at 6% failure rate. This shows a strong correlation between server locations inside a rack and the number of failures. This reiterates our observation that temperature and air movement across a rack are significant determinants for server failures. The correlation coefficient computed for (*inlet temperature for server location within rack, number of failures*) pair is **R = 0.91.**

Observation: ***There is a significant correlation (R = 0.91) between the inlet temperatures observed with respect to the position of the server in the rack and number of failures for that server. The higher the average inlet temperature at a server location within a rack, the higher the number of failures.***



a) Within Server Chassis b) Within a rack



c) Across rack locations within data center

Figure 3.5: Correlation of failures across location granularities

3.2.2.3 Correlation across multiple rack location

We observed earlier that drive bay location and server location temperatures are indeed major determinants for number of failures observed in that location. Following that, we also determine whether the temperatures observed across rack locations inside the data center are

correlated to the number of failures observed. Figure 3.5 (c) presents the temperature observed at the particular rack location (averaged across the servers in the rack). Every cluster in the data center has columns of multiple racks. Each column has an inlet cold aisle and a corresponding hot aisle. Every rack has 12 servers.

One important observation from this figure is that we would expect the Temperature line to be fairly horizontal at a fixed data center set-point temperature. However this is not the case and there is significant variation in temperatures across the data center floor. This is possible due to a variety of reasons including inefficient hot aisle/cold aisle containment, other networking or server gear venting hot air into the cold aisle and hot air recirculation around the edges. There are other significant patterns observable from the figure, especially that the rise in temperature is accompanied by rise in failures, however we note that there are several places in the figure where this is not the case. However, the correlation coefficient for the entire set of data *(temperature at data center location, failures at that location)***.** There is indeed a positive correlation and is statistically significant. Also, it is clear that the lower temperature racks have lower failures and hence the motivation to be temperature-aware in data center and server design is still valid.

Observation: ***There can be varying degrees of deviation from the Data center set-point temperature in different parts of the data center floor. Hence hot and cold aisle containment solutions are needed for higher efficiency in traditional data centers.***

3.2.3 Impact of Variations in Temperature on Failures

We observed that the failures are correlated with average temperature measured at different granularities. In this section, we explore whether variations in the temperature experienced by the disk drive has any correlation with failures. Instead of just comparing variance or standard deviation which has no reference to the mean around which the variation occurs, we use the

coefficient of variation as a representative metric. This metric is a normalized measure of dispersion of a probability distribution and it computes the variation of temperatures relative to the mean ($CV = \sigma/\mu$). We saw that average temperature experienced by disk drives already has a strong correlation to failures. We also want to explore whether large variations in temperature impact failure rates.

Figure 3.6 shows the correlation between CoV (Coefficient of variation) clustered into discrete buckets (each with 0.001 CoV) and the corresponding AFR for all disks falling into this bucket. We also plot the PDF of the distribution to show temperatures with high frequencies in the distribution. As can be seen from the figure, the actual variation of temperature measurements is around 0.8% - 3.4% of the mean for most of the hard disk drives. This number in itself is relatively small, since typical average temperature ranges between 35C-40C and this variation amounts to a small deviation from this mean. This is due to the fact that in a traditional data center, inlet temperature to the servers is tightly controlled by a chilled water loop [68], and is expected to show less variation. Moreover, the temperature difference that we observe is between different disk drive locations across the chassis and rack and is not localized to each disk drive. This also agrees with our observation that workload variations (seek requests) are expected to cause minimal variation to individual disk drive temperature.

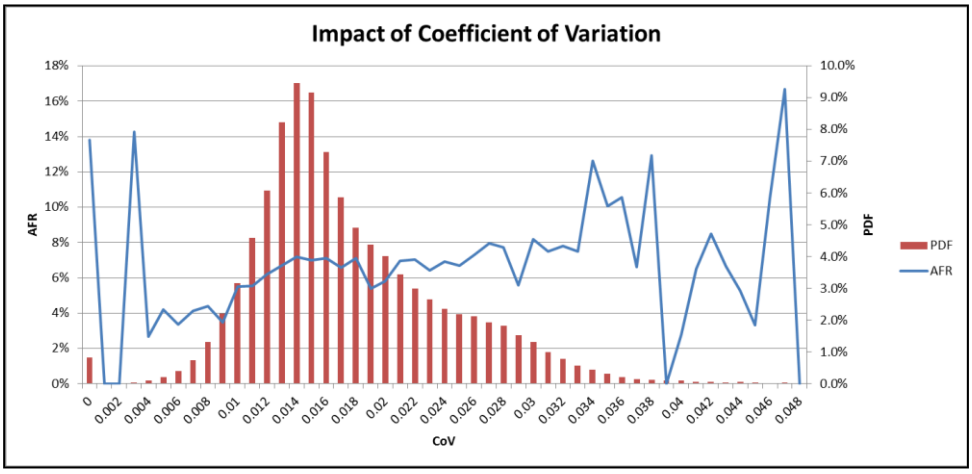


Figure 3.6: Correlation between AFR (failure rate) and CoV (Coefficient of Variation)

From the figure, we observe that there is no significant correlation between the CoV and the resulting AFR (R value of 0.21 is lower than critical value of R required for statistical correlation), though there is a slight uptrend and a positive correlation at certain CoV. For comparison purposes, correlation coefficient of *average* temperature at different chassis and rack locations with failures was in the 0.8-0.91 range. This population is from an identical server design, housing a homogeneous load-balanced data center application, and hence has little variation in terms of age, disk drive model or workload intensities.

Observation: ***This analysis shows that 1) temperature variation relative to average temperature in large data centers is minimal (less than 5%) and 2) temperature variation does not show a strong correlation to hard disk drive failures in the population under study.***

3.2.4 Impact of Workload on Temperature and Failures

In the above section, we identified that variations in temperature do not correlate with failures. However, we also want to independently see whether workload variations were the cause

of either temperature change or failures. In this section, we compare workload measurements to temperature and failure measurements separately.

3.2.4.1 Workload Intensity and Temperature

The collected data set also contains the total number of read and write operations done on the disk drive at every collection interval. This is a useful metric to have, since we can figure out the disks that were stressed more when compared to other disks. We can then correlate the observed temperature at the disk drive to see whether the disk that had a higher number of workload requests was at a higher temperature than other disks.

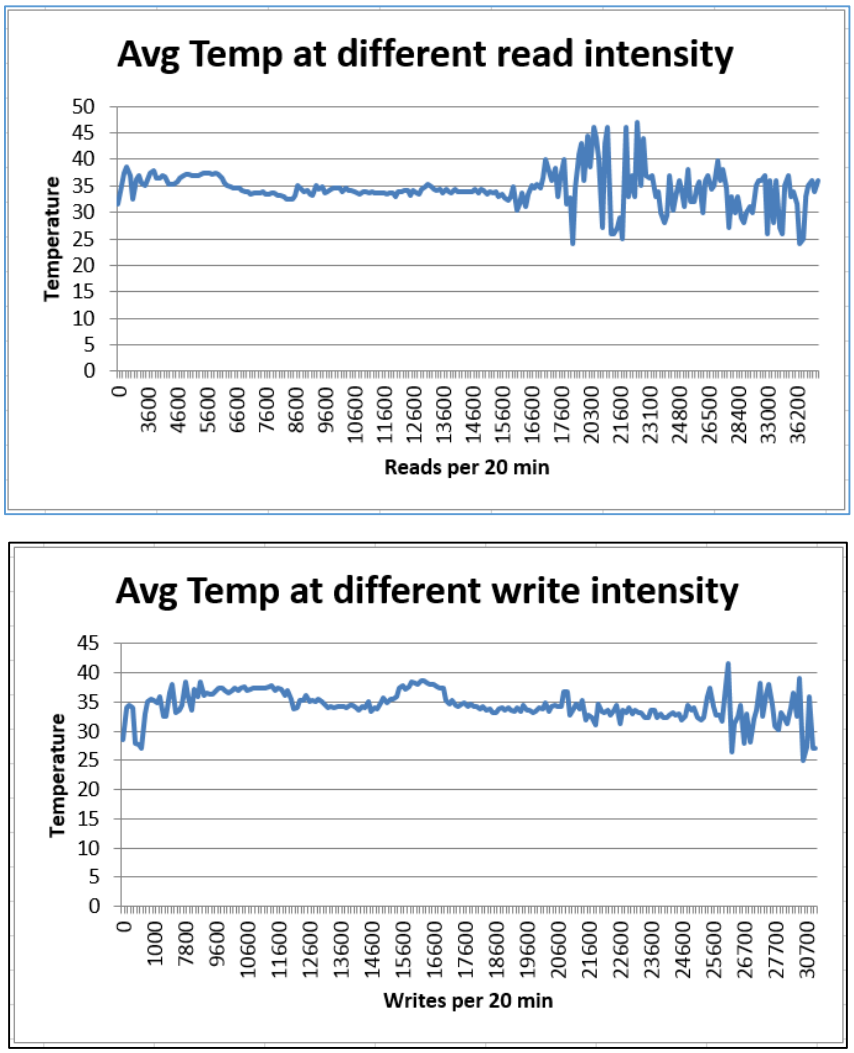


Figure 3.7: Temperature at increasing read and write intensities at all disk drives

In Figure 3.7, we plot the workload intensity at each drive and the corresponding average temperature experienced by the drive at that particular workload intensity. We plot both read and write intensities. These intensities are measured for a 20 minute interval due to data center data collection limitations. However, we expect heavily accessed disks to have consistently high access rates during the entire period of operation, since intensity is the sum of all requests over a 20 minute window. To have a sum that is large, the individual intensity measured every second (IO operations per second or IOPS) should have been large. From the figure, we are able to note that for both read and write operations, increasing intensities do not show a relative increase in temperature of the drive. As we move to the higher write and read intensities, we see temperature swings that are very high – this is due to the fact that the sample size at those high intensities is low and averaging them yields skewed temperature numbers. We show this data in the graph for completeness, but at most intensities where there is sufficient number of samples; we see no direct correlation between temperature and intensities. This confirms our earlier hypothesis that enterprise workloads have very little idle time [37] and the resulting continuous operation typically shows little or no change in temperature behavior of the disk drive [38], such that it deviates by a significant amount from the average temperature experienced throughout.

Observation: ***Workload variations do not impact temperature variations significantly for our load-balanced data center application.***

3.2.4.2 Workload Intensity and Failures

In this section, we correlate workload intensity experienced at each disk drive, with the failures experienced by disk drives. Figure 3.8 shows the correlation between average reads/ average writes and the corresponding failures for disk drives having the read/write intensity. In both the charts, X-axis plots read and write intensities measured per 20 min. Y-Axis plots the

failure rate for all disk drives that experienced that intensity. We also plot the pdf to show the distribution of read and write intensities over the measured population. We see from both the charts that there is no correlation between the read or write intensity to the failures experienced by the disk drives. This conclusively shows that workload variation in itself does not impact hard disk drive failure at data centers.

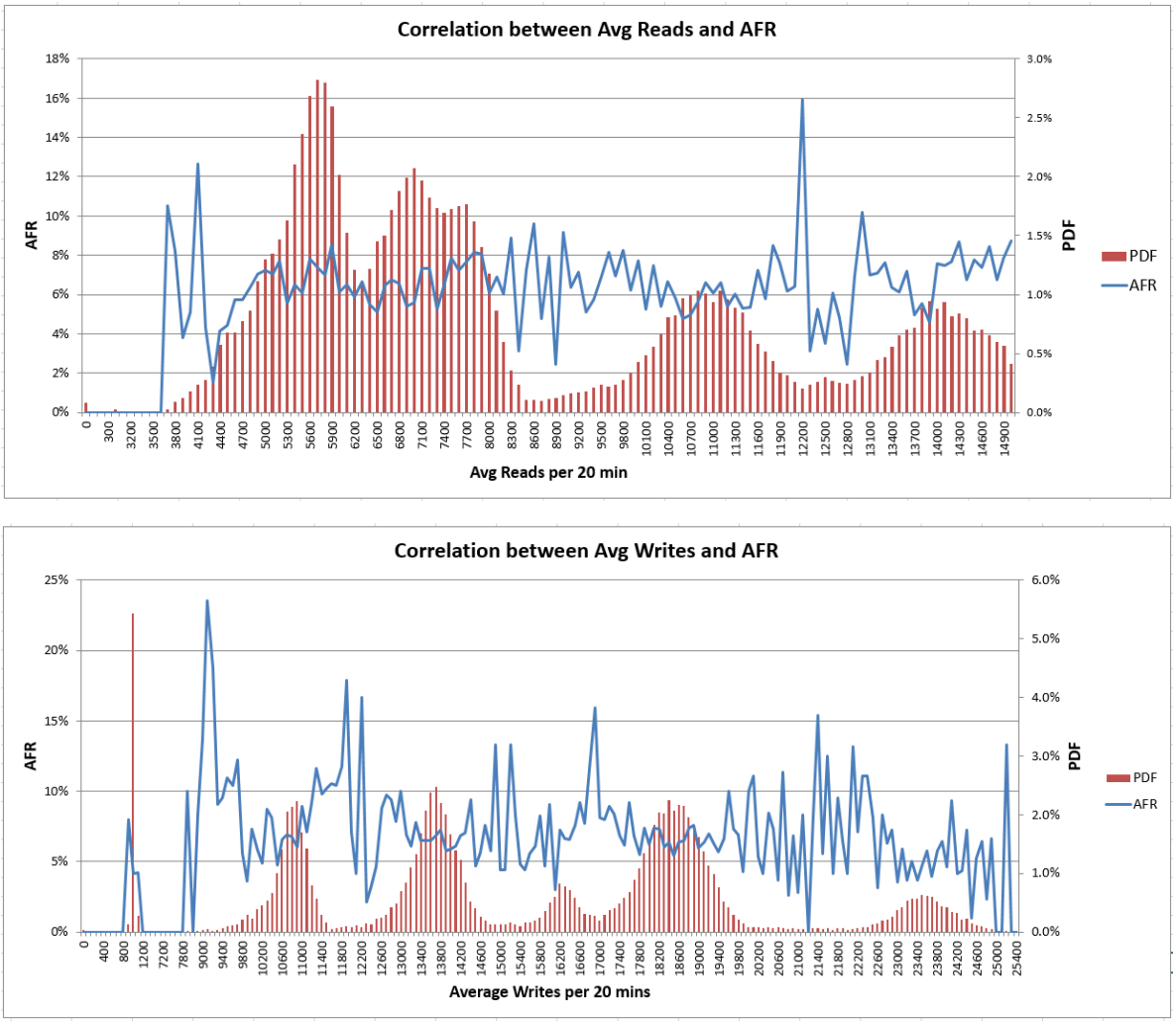


Figure 3.8: Correlation between workload intensities and failure rates

Observation: ***There is no significant correlation between workload intensities and failure rates in the data center population.***

3.2.5 Summary of Observations from Data center data

In summary, we see that average temperature has a strong correlation to disk failures at different locations inside chassis, rack and across data center floor. However, we do not observe a significant correlation between variations in temperature and disk failures. The variations in temperature are within 5% of the average and hence are not significant enough a concern for data center design. We also see that workload variations have minimal impact on temperature variations or hard disk drive failures in the data center.

3.3 EVALUATION OF TEMPERATURE CONTROL KNOBS

The results of the data center study showed significant correlation between temperature and failure rates. In this section, we evaluate the validity of our observations through controlled lab experiments. We control knobs that can influence temperature and experimentally quantify the benefit of each knob. This evaluation is done on a real system resembling the actual production system in a controlled lab environment. We evaluate the following temperature control knobs in this section:

- Workload knobs (Intensity, Different workloads)
- Chassis design knobs (Disk placement, Fan speeds)

3.3.1 Workload Knobs

In the earlier section, we saw that workload variations have minimal impact on temperature. We want to validate this with the help of an experiment, where we control two workload knobs – we modulate the workload intensity by controlling inter-arrival rates; and we also run different workloads that have different access patterns on the same experimental system. We then compare the impact of these two knobs on disk drive temperature.

3.3.1.1 Impact of Workload Intensity on disk temperature

We modulate the inter-arrival rate of the workload by delaying the time between every IO request. We simulate various inter-arrival rate from 1 ms, 10 ms, 100 ms, 1000 ms, up to 10000 ms. We also simulate an artificial workload with 0 inter-arrival time – basically, the workload sends as much requests as it can limited by the queue size specified (1024 in this case). Figure 3.9 plots the temperature measured by our thermal sensors in each of the 34 drives in our experimental setup. As can be seen from the figure, different workload intensities do not impact the drive temperature at each drive. The reason for this can be attributed to the fact that the spindle motor contributes to a significant portion of the power consumption of the disk drive [82] and as long as there is any activity on the Voice Coil Motor (VCM) that moves the read/write heads, the intensity of the operation does not have an impact on temperature.

3.3.1.2 Impact of workload patterns on disk temperature

In order to identify whether there is a difference between workload access patters we run our suite of different workloads on the experimental system. We do not change the RAID 1+0 partitions to maintain uniform infrastructure for all our experiments. Figure 3.10 shows the drive temperature for the different workloads. As we can see from the chart, there is no significant change in temperature experienced by the disk drives running different workloads. The maximum difference is a delta of 3C between Email and Messenger (OLTP) workloads. OLTP workloads have a higher read:write ratio and has a higher inter-arrival time compared to Email. They are also largely random and hence has a slightly higher seek activity that can result in the minor difference between temperatures.

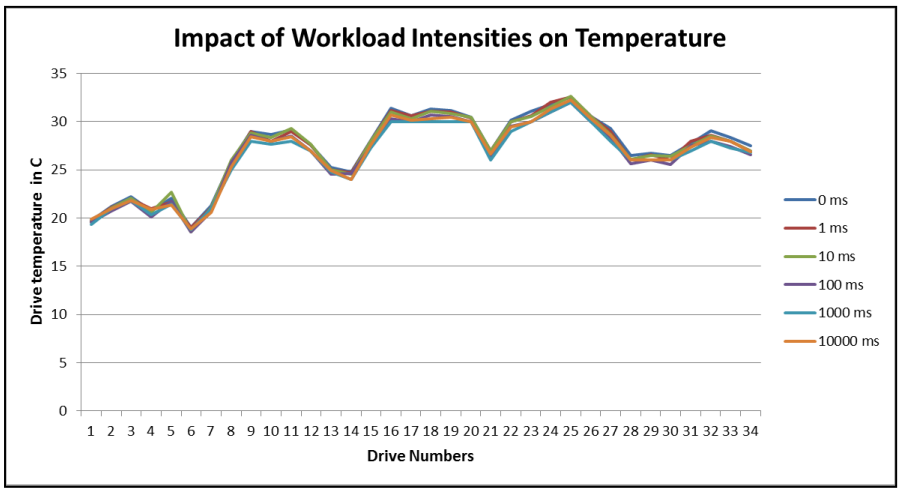


Figure 3.9: Temperature of 34 drives at different workload intensities

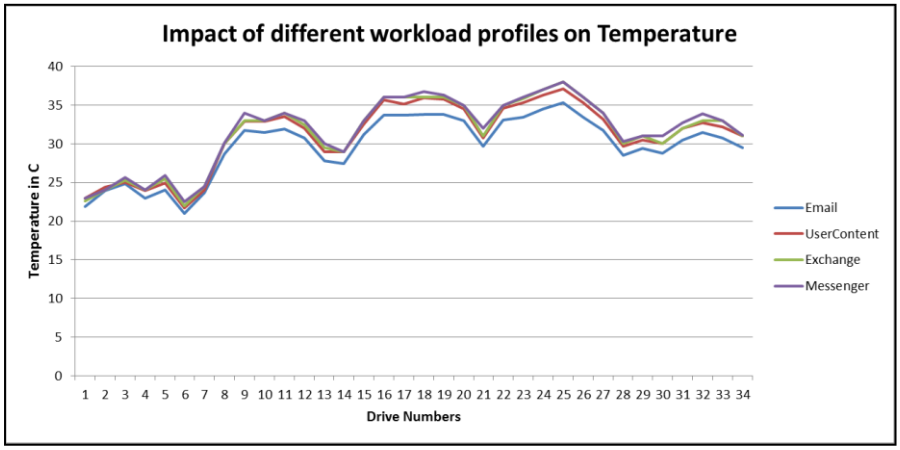


Figure 3.10: Impact of running different workload profiles on disk temperature

In this section, we saw that workload intensities or variations in actual profiles do not cause significant changes in temperature behavior at the disk drive. This agrees with our observations from our data center study that shows low correlation between workload behavior and temperature of the hard disk drives. Given this low correlation at the disk drives, we believe that investing in workload modulation to control temperature at disk drives would yield minimal benefit with respect to reducing temperature or increasing reliability. Compared to CPU temperature control using DVFS schemes or power states in processors [31], workload modulation achieves lower reduction in disk temperatures.

3.3.2 Chassis Knobs

Server chassis is composed of several components including the sheet metal casing that includes all the individual components like the CPU, motherboard, power supply, fans, memory and the hard disk drives, in addition to cables connecting the different components. The layout of each component on the chassis is deliberated and positioned in a way that optimizes the floor plan, signal integrity and cost of the overall solution. The thermal behavior of each component in the server system is impacted by the position of the system relative to inlet temperature at the cold aisle and also the cooling solution employed. CPUs have heat sinks that absorb heat produced from the processor. Hard disks however do not contain heat sinks in the typical enterprise scenario; however they are cooled by chassis level fans that move air through the chassis. The pressure difference maintained across the chassis by the rotating fans results in air flow that removes heat from the system. Typically, the components closer to the cold aisle have a lower temperature, and due to the preheating effect and the direction of air flow, the components at the back of the chassis have higher temperature. In Section 3.2.2.1, we saw the impact of difference in temperature across the chassis impact disk failures differently. In this section, we measure the impact of control knobs that can impact temperature differences across the chassis, including disk placement and fan speeds, on the temperature experienced by disk drives.

3.3.2.1 Impact of disk placements inside the chassis

In our experimental system, there are a total of 35 disk bays where disk drives could be connected. However, there is a requirement for only 34 disk drives in our system. We use the one available open slot to experiment the impact of disk placement on temperature experienced by the disk drives. We use the column positions 1 till 5 to place the empty slot in the middle of the chassis. Figure 3.11 shows the impact of an empty slot in the system. We denote the temperature of the

empty slot to be 0 in the chart. Note that whenever there is a sharp dip in the series, after 7 consecutive positions, there is another small dip in temperatures. Since there are 7 disk drives arranged in each column, the second dip in each series corresponds to the disk drive directly behind the empty slot. This experiment shows that the position of hard disk drives and empty slot influence air flow and can result in reducing temperature in storage enclosures. We see that an empty slot can reduce the temperature experienced by the disk drive behind the empty slot by close to 5C-7C. Hence based on the requirement of enterprise applications, it might be beneficial to allow empty slots with the purpose of cooling disk drives that experience a higher temperature.

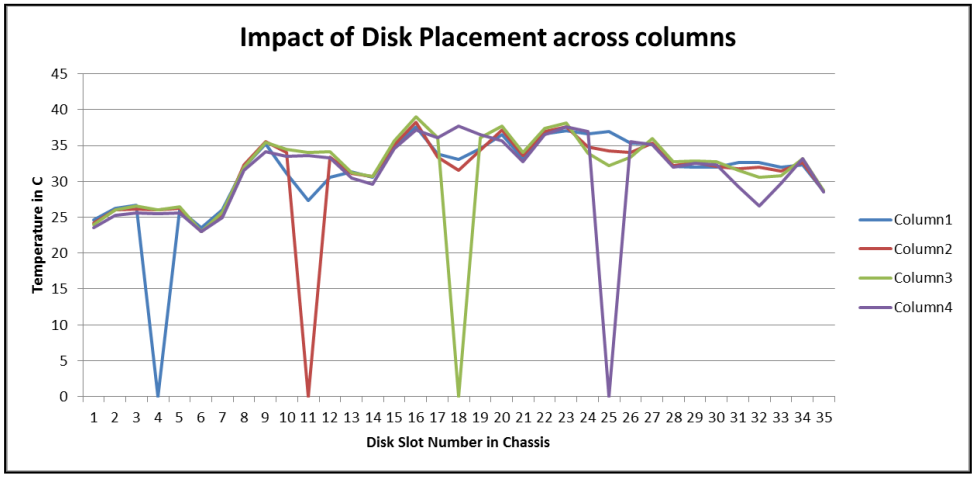


Figure 3.11: Empty slots with temperature of zero creates a dip in temperature of the drive directly behind the slot (after 7 places)

3.3.2.2 Impact of fan speeds on disk temperature

Server fans are a common solution that is used for moving cold air across the server chassis to cool hard disk drives. In this section, we measure the impact of different fan RPMs on the temperature of disk drives in our experimental setup. An increase in fan RPM results in increase in power consumed by the fans since power is proportional to the cube of the RPM. Hence we

need to evaluate the benefit of reducing temperature on reliability compared to the cost of increased power for increasing fan RPM. Figure 3.12 shows the relationship between fan speeds and temperature. In our setup, we can control the fan speed RPM from 7000 RPM (denoted by 7000-wkld) to 12000 RPM (denoted by 12000-wkld) and we increase the fan RPM in steps of 1000. We see a drop in temperature of 5C when we increase fan speed from 7000 RPM to 12000 RPM.

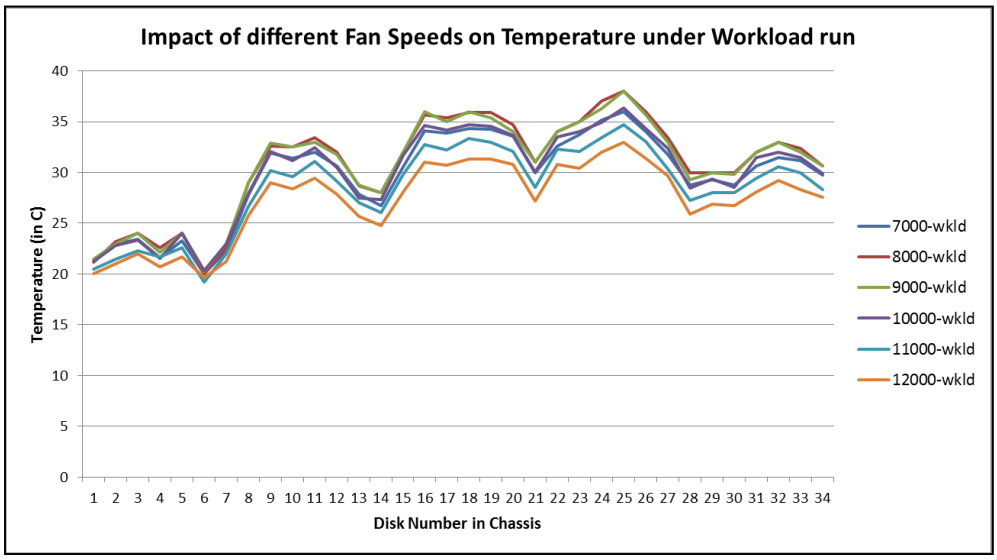


Figure 3.12: Increasing fan speeds reduces temperature of disk drives

3.4 MODEL FOR HARD DISK RELIABILITY

From our real data center study and experimental evaluation, we identify that average temperature has stronger correlation to disk failures. In order to quantify the impact of different data center inlet temperatures experienced by the servers, we had to develop a model for measuring the reliability of the hard disk drives. We used a physical Arrhenius model and estimated the activation energy based on the failures from the field. Earlier studies have estimated that duty cycle has a negative effect on AFR (higher duty cycles have higher accelerated failures) [14]. We factor

in the effect of duty cycle in the proportional multiplier for Arrhenius model in the next section. Using this model, we estimate the AFR (Annualized Failure Rate) and consider that to be a baseline for comparison between different data center inlet temperature decisions.

3.4.1 Arrhenius Model for Acceleration Factor

The failure rate due to elevated temperature is governed by the Arrhenius equation [14]. The Arrhenius acceleration factor (AF) can be expressed as:

$$AF = A * e^{\left(-\frac{E_a}{kT}\right)}$$

Where,

A, is a proportional multiplier

E_a, is the activation energy determined empirically

K, is the Boltzmann's constant that relates energy at the particle level with temperature observed at macro level

T, is the absolute temperature observed at elevated temperature points respectively

Acceleration Factor (AF) can also be expressed as the ratio between the time it took to fail under normal temperature versus the elevated temperature. Rewriting above equation,

$$Ln(t2/t1) = \left(\frac{Ea}{k}\right) * \left(1/T_2 - 1/T_1\right)$$

Where, t_2 is the time for failure with elevated temperature and t_1 is the time to failure with normal temperature.

Activation energy E_a, can be calculated from the above equation. We know empirically from Section 3.2.2, that we had almost twice the number of failures with 12 C increase in temperature. Substituting the values in the equation, we get **E_a = 0.464 eV.** We estimate the proportional multiplier (A) for the Arrhenius Acceleration Factor equation to be 1.25 based on workload duty cycle expectations. This multiplier is calculated as a function of the duty cycle expected and the duty cycle rated by the manufacturer (similar to [14]). We base our calculations on the worst case duty cycle for the workload (100%). We use the above empirically calculated value to compose the Arrhenius model for estimating Acceleration Factor at different temperatures.

Table 3.2 shows the increase in Acceleration Factor and the corresponding impact on reliability (AFR). We use the 40C row as the baseline temperature and AFR value since it is derived from typical HDD manufacturer data sheets. We see that operating the hard disk drive at 55C increases the AFR by almost twice when compared to the AFR quoted by manufacturers at 40C. The table provides a handy reference sheet for expected failures when the hard disk drive experiences a particular temperature. Given a chassis design, it is straightforward to compute the delta T observed by the hard disk drives at different location inside the chassis. We computed the delta T from SMART logs and when running a constant workload at specific temperatures. We can then use this table to estimate the failure rate for the particular chassis design, given the corresponding data center inlet temperature. Thus, this provides a methodology for selecting data center setpoint based on expected reliability.

Table 3.2: HDD temperature and corresponding AFR (40C is baseline)

HDD Temp	Acc Factor (AF)	AFR	AFR relative to 40 C
40 C	1.257	2.75	100%
41 C	1.328	2.91	106%
42 C	1.402	3.07	112%
43 C	1.480	3.24	118%
44 C	1.562	3.42	124%
45 C	1.648	3.61	131%
46 C	1.737	3.80	138%
47 C	1.831	4.01	146%
48 C	1.930	4.23	153%
49 C	2.033	4.45	162%
50 C	2.141	4.69	170%
51 C	2.254	4.94	179%
52 C	2.372	5.19	189%
53 C	2.495	5.46	198%
54 C	2.625	5.75	209%
55 C	2.759	6.04	219%

3.4.2 Application to Data Center Setpoint Selection

This section discusses the application of Table 3.2 in selecting the data center setpoint temperature. The setpoint temperature determines the chilled water temperature. The lower the setpoint temperature required, higher the energy required by the chiller units to bring down the temperature. Hence, fixing an optimal setpoint temperature by a data-driven reliability-aware approach would lead to energy conservation and better efficiency at the data center.

Table 3.3 presents two server chassis design. One design contains the HDDs in the front, and therefore is exposed to the cold aisle. The delta T between the temperatures experienced by the front HDDs and the data center setpoint temperature is minimal (1 C). The other server design consists of the inner HDDs, which has HDDs arranged one behind the other. We present only the case of the worst HDD in the design. Because of preheating, the delta T in cold temperatures is 20C. However as the air gets hotter, the chassis fans will be sped up to prevent the HDDs from overheating with a delta T of 10C. For temperatures in-between, the delta T will be assumed to

be linear. Hence at inlet of 50C, the hottest drive experiences a temperature of 60C. We assume that for temperatures below 40C there is no AFR increase and we keep that as baseline AFR and compute the relative AFR from that data point.

Table 3.3: Choosing Data center Setpoint for a) HDDs in Front, b) Buried HDDs

Inlet Temp		HDD's in Front, ΔT 1°C		Buried HDDs Design, ΔT 20°C cold de-rated to ΔT 10°C hot	
		HDD Case Temp	Relative AFR	HDD Case Temp	Relative AFR
10 C	50 F	11 C	100%	30 C	100%
15 C	59 F	16 C	100%	34 C	100%
20 C	68 F	21 C	100%	38 C	100%
25 C	77 F	26 C	100%	41 C	106%
30 C	86 F	31 C	100%	45 C	131%
35 C	95 F	36 C	100%	49 C	153%
40 C	104 F	41 C	106%	53 C	189%
45 C	113 F	46 C	138%	56 C	231%
50 C	122 F	51 C	179%	60 C	281%

As we can observe from the table, a front facing hard disk drive design experiences fewer failure events at 50C inlet temperature. However, the buried HDD design has significant increase in the relative AFR of the disk drives. Hence we need to make the decision about housing the second design in a data center more carefully. If the threshold for disk failures can be fixed, (say at 1.05X the advertised AFR rates, a 5% increase over baseline), then we need to adjust the data center setpoint inlet temperature for a data center having the second design at 25C. However, if all our servers had the first design, then the setpoint temperature could be 40C. The 15C delta between these two setpoints is a significant temperature delta to operate a data center. A 15C difference in setpoint temperature is close to 150KW difference on the data center floor. Hence it is useful to have such a methodology in place for setting data center setpoint temperature.

Observation: ***Data center setpoint temperature should be selected in a reliability-aware manner to avoid potential increases in server failures due to temperature impacts.***

3.5 COST ANALYSIS OF TEMPERATURE OPTIMIZATIONS

In the preceding sections we explored different temperature optimizations that control disk temperature. In order to quantify the cost of different optimizations, we use available power costs [40] and publicly available sources to compare the cost of optimizations. We also evaluate the cost of increased failures by using the Arrhenius model to predict failure increase with temperature. The cost calculations are a measurement of the relative differences between different options computed with assumptions as presented in Section 1.1. We use publicly available sources for estimating the cost, and this should not be viewed as the actual cost in a typical data center. A different deployment would have a different cost calculation, specific to that deployment.

In this section, we consider two optimizations: 1. Cost of fan speed increase and 2. Cost of data center chiller costs. In order to estimate the cost of fan speed increase, we identify the total power increase experimentally. The increase from 7000 RPM to 12000 RPM increases the power of the system by 137 watts and reduces temperature by 5C. We need to calculate both the power cost and the cost of increased failures to see which of the cost we should incur. For the cost analysis, we assume a typical power cost of 0.10$ per Kilowatt-hour, and a constant number of servers in the data center (we assume 10000 servers each rated at 1000 Watts to contribute to an overall power capacity of 10 Megawatts. 10 Megawatts is a typical data center size for large enterprises). To compute the increase in power cost alone,

Power cost = number of servers * increase in power * power cost (adjusted to 1 year) = **1.5 Million/ year.**

We use the Arrhenius equation to determine the difference between 5C decrease in temperature. We calculate the difference in acceleration factor and estimate the number of failing

disks in the population. We assume that the average operating temperature was 45C and a 5C decrease in temperature resulted in a 40C operation. We substitute these temperature values in the Arrhenius equation, and show that Acceleration Factor (AF) for 45C = 1.648. Compared to the AF at 40C (1.257), we see a 31% increase in Acceleration factor and hence the AFR% also increases by 31%. The average AFR quoted by disk drive vendors for enterprise class disk drives is close to 3% of the population. Assuming this value, we estimate the AFR at 45C to be 3.93%. Applying to the population of 10000 servers with 34 disk drives each, we expect an extra 3162 drives to fail every year (0.93% of overall population). The cost of replacing 3162 drives that are under warranty is minimal; however data center environments do not return disk drives to manufacturers to protect sensitive information within the disk drives. Instead, they shred the disk drives to protect data. Hence, the cost of total replacement is $632,400 (at 200$ per drive). To this number, we add additional service cost of replacement (we assume 15% of the disk cost to be the cost of service for each replacement – note that we do not have typical numbers for this service, since it is negotiated differently by each vendor for specific use case [93]). The total cost of service then becomes $94860. The total cost to the data center operator for the increased failures is $727,260.

The power cost that a data center operator would pay for a year to increase the fan speeds ($1.5 Million) is almost twice that of the cost of increased failures ($0.73 Million). However note that we do not include the cost of service downtime. For data center operators, service downtime is a critical metric, and to obtain a 1% increase in that metric, they might be willing to incur this extra cost.

Another optimization that can be used at the data center level is to change the data center setpoint temperature. This also lowers the temperature of the entire server chassis in addition to lowering the temperature for the entire data center. The power consumption of the data center

increases, and hence there is power cost associated with this knob. We compare the cost for the additional power required to increase the data center setpoint temperature. A decrease of 5C incurs additional power usage of 60 Kilowatt for a typical data center facility. This power is spent in reducing the data center chilled water temperature, and to maintain the temperature at 5C below the earlier operating temperature. Our observations correlate with previous study [67] where they consider a 4000 ton chilled water plant serving a 100,000 square feet data center at 150 W/sf. According to their results, they estimate a total power consumption of 695 Megawatt-hour annually for 2F decrease in temperature. For a 5 C decrease, we can compute the resulting power usage from their calculations (estimated at 3129 Megawatt-hour resulting in a power cost of $312,900). This cost is lesser when compared with the cost of increase in failures computed earlier ($0.73 Million).

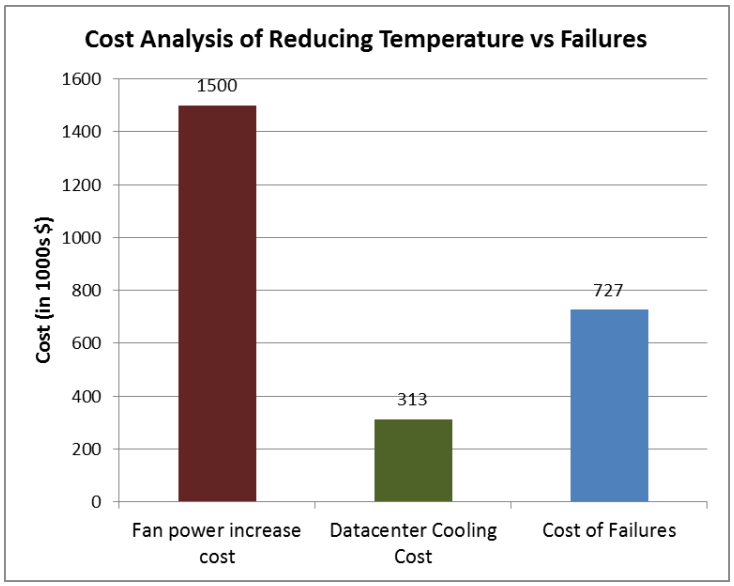


Figure 3.13: Cost comparison between reducing temperature and increase in failures

From Figure 3.13, we can see that the cost of data center cooling increase is significantly lower than the cost of server fan power increase, and is also lower than the cost of increased failures. In this case, data center setpoint temperature reduction is recommended to decrease

failures. However note that running chiller plants at lower temperatures reduces their efficiency since the temperature delta increases between the chilled loop temperature and external temperature [64] and hence this methodology should be re-evaluated for lower temperatures. In summary, there is a cost associated with increasing cooling to facilitate lower temperatures at disk drives. Similarly there is a cost associated with reducing cooling to increase data center power efficiency, attributed to the cost of the resulting increase in failures. We propose that these costs should be factored in before data center design decisions are taken.

While this section provides cost analysis for existing solutions, there has been a significant move to efficient cooling mechanisms in data centers like airside economizers, by companies like Microsoft [64] and more recently by Facebook [22]. The underlying principle behind the cooling mechanism is that outside air is cold enough for a majority of hours during the year to cool servers inside data center, and water based chiller units can be removed. During the hotter summer months, these data centers use adiabatic cooling in addition to free-air cooling [48]. This methodology of cooling data centers causes significant variations in inlet temperature and the data center setpoint temperature is not maintained at a constant level compared to traditional data centers. Every server component experiences variations in temperature according to outside temperature, in addition to relative humidity differences based on outside humidity and adiabatic cooling. This presents a completely different set of challenges in terms of quantifying reliability and is subject of future work.

3.6 SUMMARY

In this chapter, we show strong correlation between average temperature and hard disk failures. Previous works on hard disk drive failures and temperature impact are highly variant in

their claims and do not evaluate variations in temperature or the inter-relationship between workload, temperature and failures. We provide an in-depth evaluation of the impact of temperature on hard disk drive reliability, including observations from real data center case study that shows that variations in temperature or workload are not significantly correlated to failures. We also model real world data on temperature and failures and focus on the resulting impact on server design and data center cost. With experimental data, we show that certain chassis design knobs and data center knobs are better at overall temperature control, and workload knobs do not provide significant benefit to disk drive temperature control. We also provide a cost analysis that highlights the need for temperature aware server design for increased data center efficiency.

This chapter covers work published in DSN 2011 [79] and TOS 2013 [80].

4 POWER INFRASTRUCTURE – POWER AVAILABILITY PROVISIONING

Large enterprises continue to build data centers in order to respond to the growing demand in online and cloud services. Power infrastructure costs alone amount to $10-20 million per megawatt of data center capacity [40]. Utilizing the allocated power capacity efficiently results in full use of capital that is invested. Given the importance of power infrastructure for continuously running a data center, DC operators typically tend to be conservative when designing the data center. While power-capacity provisioning has received a considerable amount of focus [23][41], availability plays a key role in overall infrastructure provisioning. For instance, data center operators usually provision additional power capacity for enabling concurrent maintenance during times when certain parts of the infrastructure need to be turned off for maintenance. They also provide power backup systems that are redundant (systems that have twice the number of generators are termed 2N, while systems that have one additional generator are termed as N+1). While the generators themselves are a backup power source for the primary power source, conservative design can lead to over-provision the backup sources as well.

When provisioning the power capacity of a data center, the availability requirements are factored into the design itself. The “number of nines” is a commonly used term that denotes the overall availability of the data center [32]. A data center with 99.9 availability denotes that out of 8,760 hours in a year, the data center can be unavailable for close to 8.76 hours. A 99.999 data center facility can be expected to be unavailable for 5.256 minutes out of a year. Typically, the number of nines are computed through a reliability block diagram (RBD) [11] with component design mean time between failures (MTBF) and mean time to repair (MTTR) numbers. A data

center with a higher availability requirement tends to cost more to design and construct because it requires redundant components needed to provide that availability. The Uptime Institute, a widely recognized research and consulting organization, has published a tier mechanism [92] that classifies data centers into various classes based on availability. Tiers fall into four types starting with the basic reliability option (Tier 1), where there is no redundancy or concurrent maintenance, and proceeding to Tier 4 data centers, which feature fault tolerance and concurrent maintenance. Figure 4.1 shows the construction cost and corresponding total expected downtime of different tiers of data centers. A Tier 4 data center, with the highest possible availability among the tiers, costs almost 2.5 times more to build than a Tier 1 facility.

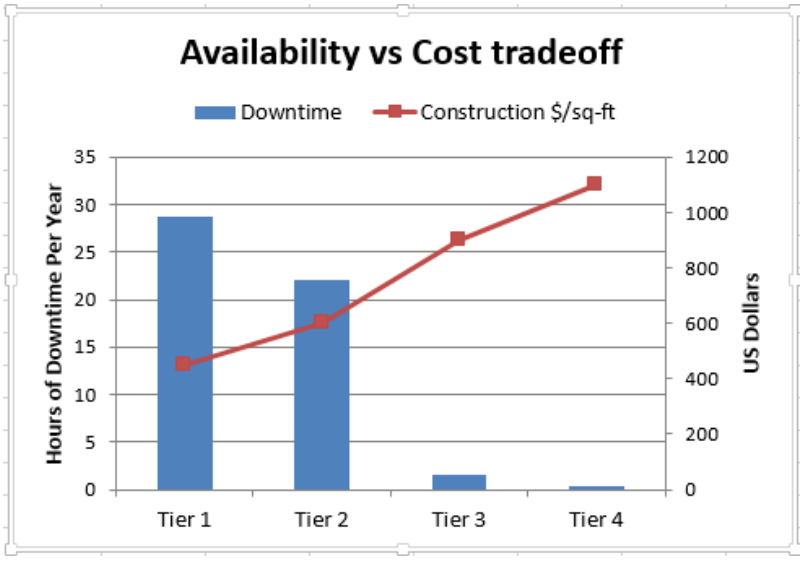


Figure 4.1: Expected downtime associated with each tier and corresponding cost to construct the data center per square foot (lower downtime implies higher availability)

However, a data center's number of nines alone is not a good operability metric for power provisioning or management because it fails to capture the impact of availability on data center performance. For instance, the number of nines is a summary metric that abstracts frequency or

duration of events: 8 hours of unavailability could have one event totaling 8 hours or 100 events of 0.08 hours each. Hence, this metric is inadequate to capture the actual distribution of the power availability events in the data center, which is important for making power provisioning decisions.

In this chapter, we make the following key contributions: Using observations from industry experience operating large facilities, we present insights into power events in real data centers. We show that a significant portion of observed power events are of short duration (within 21 seconds) and typically have varied inter-arrival times (several days between each event). We present a methodology to capture the impact of availability events on the data center design. We argue that power utilization and power availability models should be constructed and be used as workloads to guide power availability provisioning. We present a method for estimating data center performance in the presence of power events by using artificial power caps on two real data center workloads running on actual servers. We present a novel redundancy technique, N-M redundancy, which is a reduced redundancy provisioning method that contrasts with the additive redundancy policies currently used (2N or N+1). We also show the validity and cost benefits of N-M redundancy.

4.1 BACKGROUND ON POWER REDUNDANCY

4.1.1 Power Delivery Infrastructure

The power delivery infrastructure in a data center consists of a primary utility that supplies power to the entire facility (in some scenarios, more than one utility supplies power for increased redundancy). Once power is delivered from the utility, it is transferred through a utility substation, and then through a medium-voltage (MV) distribution. In some cases, smaller data centers utilize power at low-voltage (LV) levels. The power distribution includes transformers to transform to

lower voltage levels (from distribution levels to operability levels) and circuit breakers to guard against unintended power scenarios. In addition to the utility distribution, the power infrastructure also includes backup (that allows for continuous operation when the utility provider encounters a failure) and reserve distributions (that allow for concurrent maintenance). Uninterruptible power supply (UPS) systems enable the transfer of power from the utility to generators in case of a power event. The UPS have batteries that store power for a pre-designed amount of time and act as an intermediary power source for the IT equipment (servers and network devices), in order to provide

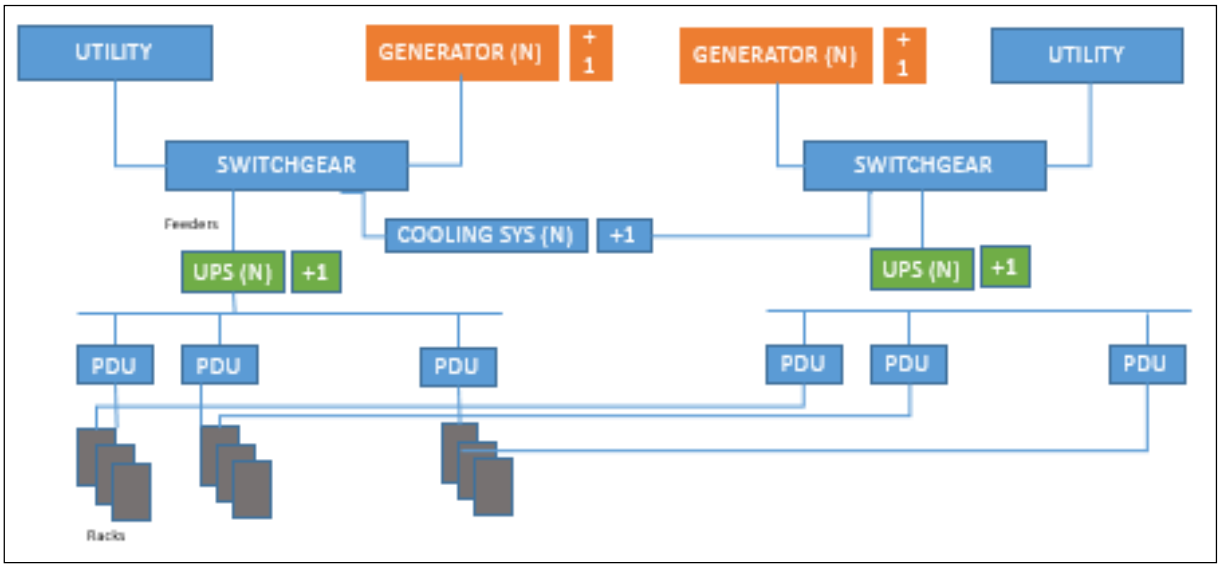


Figure 4.2: Power infrastructure with dual utility, 2N distributions, N+1 component in each distribution

time for the generators to power up and provide the continuous backup supply. The generators themselves are rated for the power capacity that they need to support and typically have a fuel tank that supplies the fuel needed for producing power to the facility. A simple non-redundant power distribution is described in Figure 2.2 in Chapter 2.

Figure 4.2 shows a highly redundant distribution that has two utilities feeding the data center. If one utility goes down, the expectation is that the other utility takes over. If the other utility also fails, there are generators that take over. This example illustrates a highly redundant

distribution path. In some implementations, the generators are shared between the two utility paths. The UPS are N+1 and are present on both distribution paths, and they also can be shared between distribution paths in certain implementations.

4.1.2 MTBF versus MTTR

During design, it is common to look at components that fail often and incorporate redundancies. However, with the shift to software resiliency and cloud services [29], applications are designed to tolerate failures because they have redundant copies and application migration built into the system. From the perspective of a data center operator, it is important to repair the failure so that applications would not be affected for a long duration. Consider the case of a cloud application designer who needs to write an application based on expected availability. The application developer needs to know the time it might take for a data center to come back online so she can decide whether to migrate services to another location. In such cases, the focus is on reducing the repair time, especially for components that are higher in the distribution hierarchy.

Table 4.1: Example MTBF hours and MTTR hours of power infrastructure components

Component	**MTBF (hours)**	**MTTR (hours)**
Dual Utility	27,712	0.56
Switchgear	5,156,470	2.40
Feeder	3,718,331	15.70
Transformer	7,895,436	5.00
Generator	13,839	13.62
UPS	499,774	6.09
PDU	55,800,000	3.10

Table 4.1 shows some of the major components in a data center and their MTBF and MTTR. If the MTBF is a large number, then the component has high reliability. If the MTTR is a large number, then the component takes a long time to repair. The numbers are derived from the IEEE Gold Book [47] based on the IEEE Standard 493, which provides failure data for industrial and commercial power systems based on surveys of industrial plants. These numbers are from a publicly available source; the actual MTBF and MTTR numbers can vary for real data center deployments. From Table 4.1, we observe that the feeder MTBF is higher than the UPS MTBF by almost an order of magnitude; however, it takes longer to fix a feeder that feeds several distributions underneath it, whereas it is easier to replace a UPS component. Another important factor in power availability design is the hierarchy of the power distribution. A failure of a component at a higher level in the hierarchy can impact a larger number of IP equipment, whereas a failure in a component lower in the hierarchy has less impact and provides greater fault isolation. Common practice is to use distributed UPS [57] at a rack level rather than a conventional centralized UPS at the low-voltage distribution level.

Observation 1: Redundancy decisions need to factor in the cost of repair (MTTR) and not just the MTBF of the components.

4.1.3 Number of Nines Metric

Typically, the number of nines is calculated by taking a power infrastructure layout (e.g., those shown in Figure 4.2) and applying series or parallel calculation of individual component availability [11], similar to circuit-resistance calculations. This methodology of constructing a RBD is used commonly in the industry, and the number of nines is a popular term for denoting availability. While this term is useful for understanding expected availability, there is no guarantee

that the actual data center performs to the design constraint. The operational behavior of the data center could be significantly different, and it could experience downtimes longer or shorter than the number of nines for which it is designed. The number of nines metric does not provide any information on the frequency or duration of events. Hence, it is imperative to talk about availability in terms of workloads and data center behavior. The rest of this chapter attempts to provide a framework to define such behaviors and the concept of availability.

Observation 2: The number of nines is a design-time metric that is insufficient for measuring the actual operational performance of a data center.

4.1.4 Types of Power Availability Events

In order to understand the impact of power events, it is necessary to understand the different types of possible events. The Information Technology Industry Council (ITIC) provides a curve that defines power-event timeliness and voltage levels for tolerances. The ITIC curve broadly captures the impact power events can have on IT equipment [54]. There are seven types of power events: transients, interruptions, sags (under-voltage), swells (over-voltage), waveform disturbances, voltage fluctuations, and frequency variations [26][89]. Based on the event and its impact on IT equipment, the regions of operation on the ITIC curve are divided into three broad categories: an area of the curve in which there is no interruption in function, an area that is the prohibited region (damage to equipment), and an area in which functionality is affected but no damage is done to the equipment. Figure 4.3 presents the ITIC curve and shows the different regions of interest.

Data center operators are concerned with the prohibited region, and thus design circuits with surge protectors and circuit breakers to prevent damage to expensive equipment. The “no

damage region" is also of interest because this is a period during which there is no damage to the equipment, nevertheless this affects the data center by causing downtime. DC operators design redundancy into distribution to protect against such events. Among the events in this region, sags up to 80% of the nominal voltage are rated to be tolerated for up to 10 seconds, while sags up to 70% of the nominal voltage are tolerated for only 0.5 seconds, beyond which the IT equipment is either turned off or switched over to some form of backup power. If there is no backup power source, lower under-voltage levels that occur for more than 20 ms result in IT equipment being powered off with no tolerance limit.

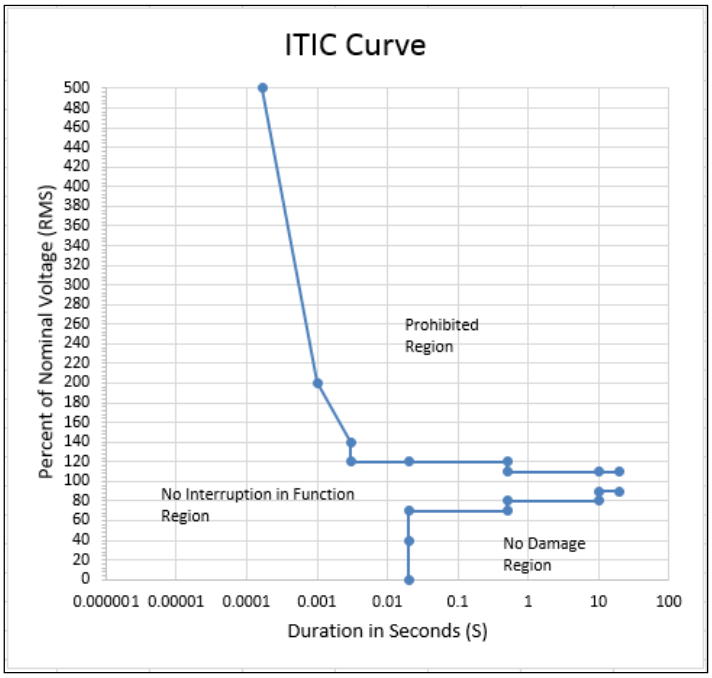


Figure 4.3: ITIC Curve showing operation region as a function of nominal voltage and duration of event

Table 4.2: Typical ride-through timing for energy storage components in the data center power infrastructure

Component	**Energy Storage**	**Duration of Ride-through**
Power Supply	Capacitor	20 ms for 50 Hz (1 cycle)
UPS	Batteries	More than 1 minute (2-5 minutes)
Generator	Diesel/Fuel	24-48 hours

While the ITIC curve specifies operating tolerances, IT equipment still has to function in the presence of power events. Hence, data center power distribution includes energy storage components that can provide energy for a specific duration for which it is provisioned. Table 4.2 shows typical times for which different power equipment in the distribution is provisioned. Power supplies that are close to the server can ride through power events that are less than 20 ms (for 50 Hz) within 80-120% of nominal voltage limits. UPS devices are provisioned to handle events longer than few seconds to minutes. However, they are designed to be short-term battery storage, enough to transition load from main utility to generator backup, and provide only transient energy to IT equipment while the generator starts up. Generators themselves can continue to operate for a long duration; however, they have a limited fuel supply (24-48 hours) that must be replenished. Each layer of redundant power is expensive to provision, and the objective of this work is to provision the redundant power equipment efficiently.

Observation 3: Sags up to 80% of nominal voltage are tolerated for 10 seconds, while sags up to 70% are tolerated for 0.5 seconds This implies that safeguards need to be put in place to handle power events that fall outside the acceptable operating ranges.

4.2 POWER WORKLOADS AND CHARACTERIZATION

To evaluate the trade-offs among power usage, power availability, and data center performance, we need to characterize and model power utilization, power availability and performance. In this section, we identify the set of workloads needed to determine an efficient methodology for power availability provisioning.

4.2.1 Power Utilization Traces

The power infrastructure carries a maximum load budget that is a design-time constraint. All the equipment, including the substation, transformers, and circuits, is sized based on this maximum supportable capacity. If the actual power utilization is lower than the maximum rated capacity of the data center, then the power infrastructure could afford to ride through a few power events. For instance, in a bank of ten generators rated for 20 MW, if one generator fails, the resultant capacity is 18 MW. If the entire facility load during the time of failure is less than 18 MW, then the generators should be able to ride through this failure event. Hence, it is important to understand the workload power utilization. While it is desirable to use the power capacity of the data center completely, it is not always the case. The likelihood of a power availability event happening at the same time as a power peak must be evaluated. To understand power-spike behavior that would be valuable for UPS sizing, we need to consider actual power traces from real data centers. A recent study from Microsoft data center power traces [94] concludes that power consumption has potential for statistical multiplexing (the occurrence of peaks at the same time is fairly minimal, and we can multiplex different component loads together such that the sum of the total is smoothed out) as we move up the hierarchy. The same study also shows that there is significant self-similarity in the power traces using Hurst parameter analysis. This implies that power utilization traces can be regenerated for studies by observing shorter duration workloads.

Observation 4: Data center power utilization has good statistical multiplexing as we move up the hierarchy. Peaks rarely occur at the same time across multiple clusters of machines. Traces show self-similarity, which enables us to construct models and regenerate workloads.

4.2.2 Power Availability Characterization

While there have been several previous works on power utilization characterization, there is very little work on power availability characterization. In this section, we observe data from two functioning data centers: data center 1 (DC1) during a period of one year, and data center 2 (DC2) during a period of five months. We anonymize the data center location and actual date and provide normalized data to protect proprietary information. We characterize the duration of power events and the inter-arrival times.

Duration of power events: Figure 4.4 shows the cumulative distribution function (CDF) of the duration of all power events recorded at DC1 and DC2 during the period of study. If all of the events are shorter than a full AC power cycle (20 ms at 50Hz or 16.7 ms at 60 Hz), then a power supply would be able to ride-through that event, as shown in Table 4.2.

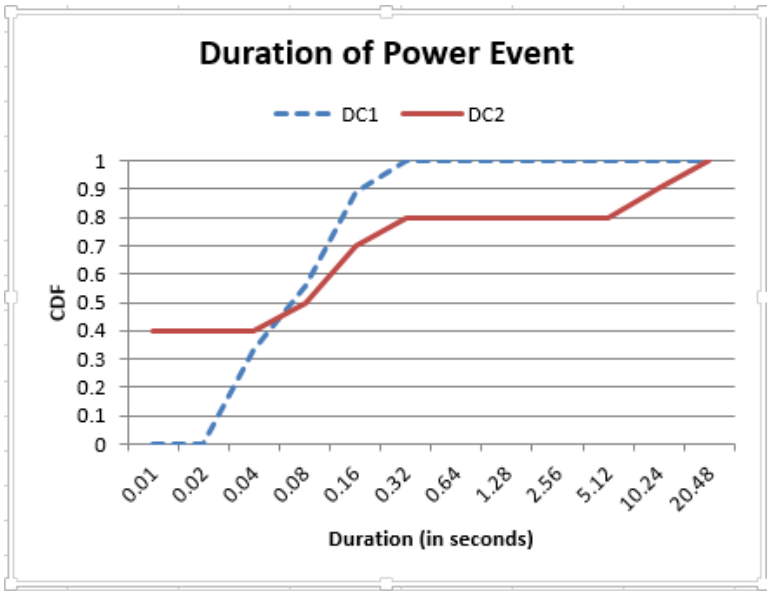


Figure 4.4: CDF of duration of power events from the two data centers

However, as we can see, almost all events at DC1 were longer than 20 ms and more than 60% of events at DC2 were longer than 20 ms. Hence, a UPS ride-through is a necessity here. The

last bucket (20.48 seconds) includes all events that are greater than that value as well, in the interest of capturing the significant portion of the population. It is interesting to note that almost all events are less than 21 seconds in duration (except for a small percentage at DC2), which is well within the tolerance limit of UPS as shown in Table 4.2. We need generators only to provide a continuous power source, and UPS can act as a temporary power source. We find that in a large majority of cases, even before the generators start up, the main utility line is back online; hence, the generators would not have been utilized. However, it is very hard to predict duration of all possible power events during operation, so generators get started assuming that there would be long power event.

Inter-arrival time of power events: Figure 4.5 shows the cumulative distribution function of the inter-arrival time for power events. The CDF of inter-arrival times is not skewed, favoring short or long durations. Instead, it is spread equally across the measured values, thereby reflecting a varying pattern of occurrence. About 40% of the power events occur within 10 days of each other, whereas the maximum inter-arrival time is within 100 days (40 days for DC2 and 100 days for DC1). From the data we analyzed, we found that no two power events happened within seconds or minutes of each other. To avoid such an occurrence, data center operators typically run on backup power (generators) for a period longer than necessary, even after the main utility is back on, to make sure that the power delivery is stable.

Observation 5: Real power events in data centers are of short duration in the sample we characterized. UPS systems would be sufficient to ride-through the duration of these power events. Generators would have seen minimal usage for such events.

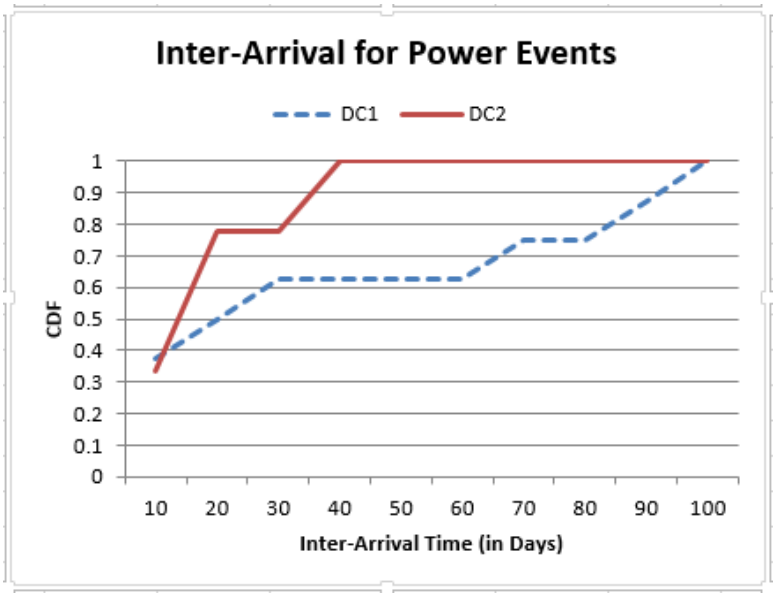


Figure 4.5: Inter-arrival time CDF for power events from two data centers

4.2.3 Putting it all together

The earlier sections provided data on power events, and observed two important qualifications: power capacity need not be utilized to its maximum limit at all times, and power availability events are of short duration. To simulate power behavior, we need to model both the spatial and temporal characteristics of power utilization and power availability. There are multiple approaches to regenerate power workloads, including replaying the trace of the workload, creating synthetic traces based on real workload properties, and using hierarchical state-based models that can represent different levels of detail [84]. The hierarchical model has been used previously to generate storage and network traces with high fidelity and efficiency [16][18]. To evaluate operational behavior of data centers, we need a method to regenerate workloads and simulate behavior. We use the duration and inter-arrival time distributions from this section in Section 4.3 to estimate the total performance impact of power availability events.

4.3 POWER AVAILABILITY PROVISIONING

A simple way of saving cost on redundancy is to reduce the capacity that offers redundancy in the infrastructure. In this section, we capture the impact of power events on performance service level agreements (SLAs) and propose a new methodology to reduce cost that is spent on provisioning generator capacity, using our observations from earlier sections. Generators are power sources that are used only when the main utility line has a failure. We propose a mode in which, during such failures, the provisioned capacity of the generators is less than the overall capacity of the data center. We call this mode N-M redundancy, since compared to traditional provisioning of N, N+1 or 2N, we actually reduce the total capacity of generators. We use M to denote the total capacity that could be reduced in percentage, compared to overall capacity. Figure 4.6 explains this concept visually for an example data center provisioned for 10 MW. We assume that each generator is a 1-MW diesel generator and the UPS capacity is N+1 (We assume that UPS are also constructed in 1 MW blocks).

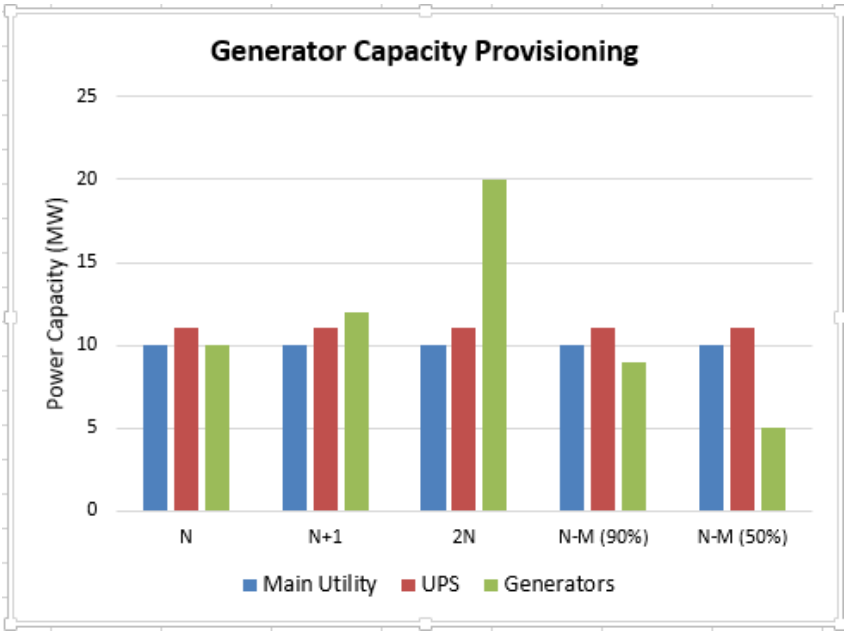


Figure 4.6: Generator capacity provisioning for different redundancy levels (lower capacity implies cost savings)

To provision less generator capacity than that provided by the main utility, we still have to be able to throttle the load to the maximum allowed capacity of the generator block when the load actually runs on the generator power source. This happens only during outage scenarios. From our characterization, we observe that the duration of power events are shorter than 20 seconds and the generator system would not get activated. However, different data centers might have different behaviors. Hence, we also need to provide a methodology for those cases when the power events are longer than 20 seconds in duration.

4.3.1 Methodology Description

From the earlier sections, we observe that all events that are greater than 20 ms cannot be tolerated by the power supply in the server. One way to address the limitations of the power supply is to augment it with a supercapacitor [6]. However, such supercapacitors are expensive and are not widely available. Hence, the power events are handled by the UPS system, while the generator starts up. One valid question is whether the UPS itself could be reduced in capacity. For the UPS to be under-sized, power throttling needs to be applied within one AC cycle (i.e., the limit of power supply). If this cannot be achieved, then the power consumed by the IT load could overrun the provisioned capacity of the UPS, cause circuit breakers to trip, and potentially bring down the entire data center. UPS can be under-provisioned only in the case when power utilization traces can show that for all times of measurement, power usage was below a certain threshold, and can never exceed that threshold from an application-level guarantee. The typical approach to reducing the power usage of servers is to use power capping [42]. However, power capping is not sufficiently fast to allow the UPS capacity to be reduced. The typical latency to enact a power capping command is in the order of 110-350 ms [5], and hence it violates the 20 ms limit of the power supply by an order of magnitude. Based on UPS capabilities and data center power

demands, for the application considered in this section, we apply our methodology only to reduce the sizing of generators. Generators take a certain amount of time to start up while the UPS provides a capacity buffer. The time provided by the UPS buffer is sufficient to implement power capping. Hence, we translate outage events into power-capping events for the duration of the outage. If the outage event happens during the time that the power utilization itself is lower, then the power capping will not have any impact on performance. However, if the power utilization is greater than the provisioned generator capacity (N-M case), then power capping will affect the performance SLA. The next section provides a model based on an actual application benchmark to characterize the relationship.

4.3.2 Performance Model

To predict performance impact during the time of a power availability event, we need to map performance curves to power-limit curves.

4.3.2.1 Performance Workloads:

For evaluating the performance impact of power limits, we consider two large scale online services workloads: WebSearch and Cosmos. These two workloads are representative of cloud-scale workloads that are deployed across hundreds and thousands of servers. For purposes of power availability provisioning, we consider the scope of cloud-scale workloads, but this methodology is also applicable for other classes of workloads like transaction processing as long as the performance impact is tolerated by the workload. Online services and cloud services are typically designed to be migrated between multiple data center locations in the event of a disruptive power event in one region, and are also tolerant to performance variations defined within acceptable thresholds. Applications that are delay sensitive might exhibit different SLA requirements, and should be characterized through a similar methodology.

WebSearch: In large scale web search, queries are distributed to many nodes as they arrive, by a top level aggregator, in the case when the aggregator is not able to serve the request from its cache. After examining its subset of index, each node returns the most relevant pages and their dynamic page rank. The content served to the user is then accessed through the sorted indices. The performance requirements of the WebSearch application is defined by QoS (Quality of Service), throughput and latency. Given a target QoS, we measure the sustainable queries served per second (QPS). The sustainable QPS also satisfies the latency constraints of the application.

Cosmos: Cosmos is a data storage and large scale data analysis application [53]. A large set of machines are available for scheduling a pool of jobs that gets executed in parallel. This workload is representative of MapReduce and Hadoop, parallel databases and other application instances processing a large set of data. Cosmos is typically CPU intensive, and has high utilization on its servers. Execution time of a job is a direct measurement of its performance, since the faster a job gets executed, several more jobs can be scheduled from the pool.

4.3.2.2 Power Capping

In order to emulate workload behavior under different power loads, we employed server level power-capping [42]. Power capping is a methodology by which the server is limited to run at a pre-defined power level [24][74][31]. Power capping could be implemented through hardware mechanisms on the server, or through the OS. Server vendors and data center solutions providers have started to offer power capping solutions [42]. Power capping using online self-tuning methodology and feedback control algorithms [97] has been studied for individual servers. For the purposes of our experiments in this work, we chose to implement system level power capping mechanism that measures total server power and caps the server to a pre-described maximum limit in operation.

4.3.2.3 Results

We run our workload at different power caps to estimate the performance at different power levels. We can then use this as a proxy to determine the overall performance during the times when the power capacity is diminished due to failure events. The % Cap is the power cap at a level that is 90% - 60% of the peak power that can be consumed by the server. We limit our experiments upto 60% cap since cap limits below that are not guaranteed in accuracy.

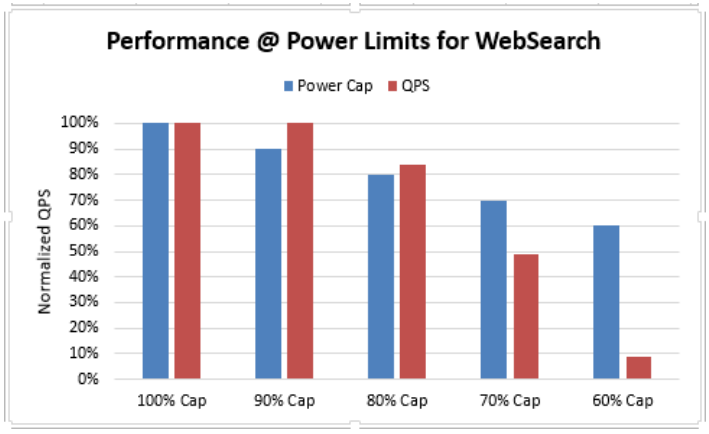


Figure 4.7: Performance model for WebSearch (QPS = queries per second, measured at power caps from 100% to 60%)

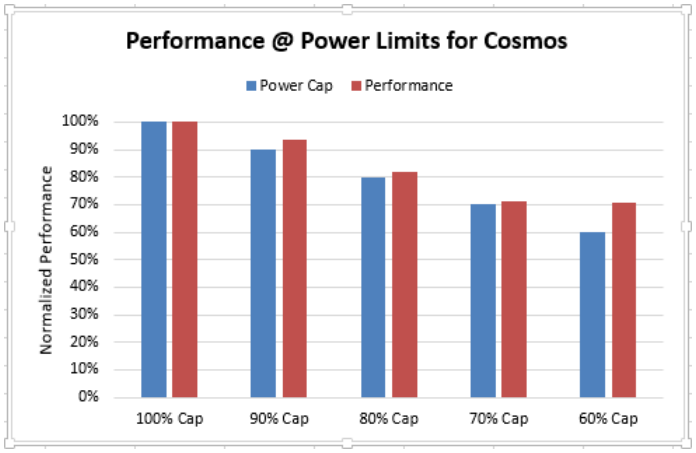


Figure 4.8: Performance model for Cosmos (Performance is normalized from execution time for Cosmos jobs)

Figure 4.7 shows the normalized sustained QPS measured at different power levels. If the data center runs at a reduced power level of 80%, we can use the data from this figure to compute

the corresponding QPS to be 83%. The figure shows that using a 90% power cap has negligent impact on the performance SLAs of this workload. Hence, reduction in data center power capacity by 10% would not affect this particular workload materially, and it is a straightforward optimization that we could make (N-M, where M=10%). This methodology assumes that the workload of interest has been profiled as shown and the performance has been benchmarked at different power levels before identifying the tolerable levels for performance degradation.

Figure 4.8 shows a similar graph for Cosmos workload, where the performance is a normalized measure of the total execution time. When compared to the WebSearch workload, Cosmos workload shows a drop in performance even for the 90% cap case. This is due to the fact that Cosmos is a map-reduce like workload, with heavy CPU load. Hence even small power caps impacts CPU utilization, and the execution time of the job. However, the drop in performance is more proportional as the power cap is reduced. For instance, the Cosmos workload has a performance of 71% at 70% power cap, whereas the WebSearch workload has a normalized performance of only 49% at the same 70% power cap limit. This shows that WebSearch is highly sensitive to lower power cap limits due to latency limitations when compared to Cosmos workload. We use both these performance models for computing the impact on performance for these respective workloads in the following section.

4.3.3 N-M Generator Capacity Sizing

The previous section showed that a 10% reduction in generator capacity would not affect the performance SLA for WebSearch when the load runs on the generator during a power event. This section determines whether we can reduce the generator capacity further and evaluate the impact of such a design on application performance. To estimate the impact of generator sizing,

we assume a hypothetical UPS sizing that is just sufficient to implement power capping at the high-end mark (350 ms) [5]. We assume that the UPS transitions to the generator successfully within this time, and that the power capping is implemented according to the generator sizing (90%, 80%, 70%, and 60%). UPS sizes assumed here are not common in actual data center provisioning, and we use the aggressive time limit (350 ms) specifically to illustrate our methodology. We use the duration distribution from Figure 4.4 to calculate event durations that would result in generators supplying power to the facility for that duration. We also use the inter-arrival time distribution obtained from Figure 4.5 to generate a time-series of those events to simulate power event frequency during a year. We then use our performance model from Figure 4.7 and Figure 4.8, to compute the impact of power events at various levels of generator capacity reduction (ranging from 90% to 60%) during the entire year. We represent the overall impact on performance as a percentage of the data center's entire performance for a full year in Figure 4.9 for WebSearch and Figure 4.10 for Cosmos (The y-axis shows the deviation but does not start from zero). As we can see, there is minimal impact given our power-event distribution and performance model for WebSearch. For 90% capacity provisioning, there is no impact. For 60% capacity provisioning, the impact is only 0.00008% of overall performance during a year for DC2. Note that these results would be different with a different distribution of power availability events. There would be no impact at DC1 because all of its events were less than 350 ms and were absorbed by the UPS energy storage capacity.

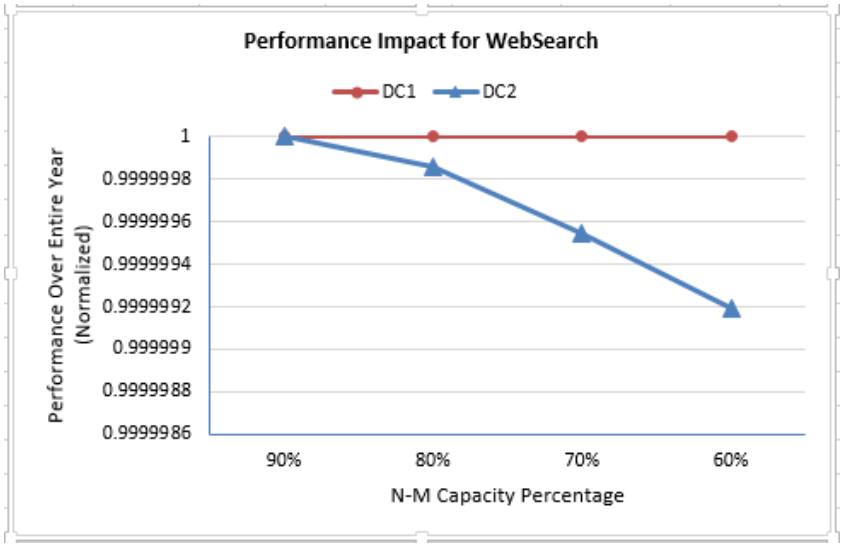


Figure 4.9: Performance impact of power events with N-M generator capacity for WebSearch

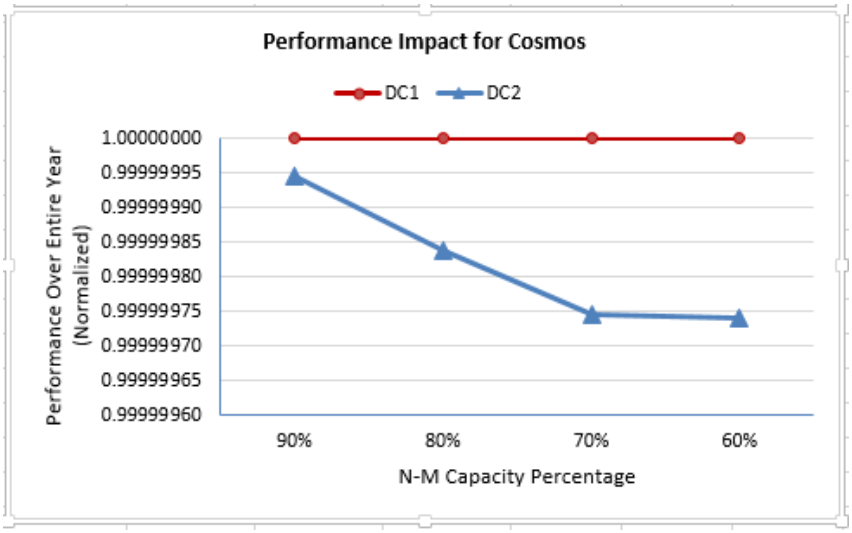


Figure 4.10: Performance impact of power events with N-M generator capacity for Cosmos

Figure 4.10 presents performance for Cosmos, however even at N-M redundancy level of 90%, this workload suffers some performance penalty. For the 60% power cap case, the penalty suffered by Cosmos workload is almost 3 times that of WebSearch workload. However given the nature of the highly parallel batch jobs that run on Cosmos machines, it is more tolerant towards

performance differences. In both these figures, we observe that performance penalties are minimal, motivating our methodology that workload profiles should be considered along with application service-level agreements (SLAs) and appropriate provisioning should be adopted for specific power workload profiles in the data center.

While the Y-axis on both these figures show very slight deviation from total performance in absolute number, it should be noted that this is normalized performance over an entire year. Moreover, online services like WebSearch and Cosmos already are impacted by performance variations at large scale due to several reasons including resource sharing, power limits and energy management policies. Developing tail-tolerant software is recommended to deal with such scenarios [16]. Hence, if the performance variation introduced by our methodology is within the tail tolerance limits designed into the application, then the SLA impact would not be significant enough to impact overall responsiveness of the workload. However, in the case of power availability provisioning the entire facility would be limited in terms of overall power budget, and hence the performance deviation would be felt across the entire workload. Therefore, the application tolerance thresholds should be taken into account when designing power limits for online services workloads.

4.3.4 Cost Impact

While the performance impact seems low from Figure 4.9 and Figure 4.10, the impact to total cost of operation of the data center could be significant. Since the data center runs online services, the impact of downtime to the service is significant. In addition, the cost of not using the servers for even a short duration is equivalent to wasting the capital that was invested in building the data center for the same duration. It is important to understand this consequence when looking at the

cost savings from N-M generator capacity proposal. However, if an application is able to tolerate short durations of reduced performance, then it could leverage the benefit of having reduced generator capacity. Assuming that generators contribute to 20% of overall power infrastructure cost [94], the total cost savings at 60% N-M capacity provisioning for a 10 MW data center is close to $8 million. In the 90% case, the total savings would be close to $2 million. Given these considerable savings, N-M redundancy option should be a serious consideration if the performance impact due to power events is within tolerable limits for the workload.

4.4 SUMMARY

In this chapter, we show that typical availability metrics are not sufficient to predict data center behavior for power availability events. We first characterize power availability events with data from two real data centers. We then provide a technique to measure performance impact due to power availability events and propose a novel redundancy mechanism (N-M redundancy) for generator capacity sizing. This chapter formed the basis for developing an extended simulator for making data center provisioning decisions [83]. We believe that a workload-driven provisioning methodology for power infrastructure is key to reducing capital cost and operational expenditure on data center infrastructure.

This chapter covers work published in CF 2014 [78].

5 MANAGEMENT INFRASTRUCTURE – WIRELESS DATA CENTER MANAGEMENT

The continuous and cost-efficient operation of a data center heavily depends on its management network and system. A typical data center management (DCM) system handles physical layer functionality, such as powering on/off a server, motherboard sensor telemetry, cooling management, and power management. Higher level management capabilities, such as system re-imaging, network configuration, (virtual) machine assignments, and server health monitoring [1][53], depend on DCM to work correctly. For this reason, the DCM has to be extremely dependable. DCM is expected to function even when the servers do not have a working OS or the data network is not configured correctly. Today's DCM is typically designed in parallel to the production data network (in other words, out of band), with a combination of Ethernet and serial connections for increased redundancy. There is a controller for a rack or a group of racks (cluster). The controllers are connected through Ethernet to a central management server. Each server within the cluster has a motherboard microcontroller (BMS) that is connected to the cluster controller via point-to-point serial connections. For redundancy reasons, every server is typically connected to two independent controllers on two different fault domains, so there is at least one way to reach the server under any single point of failure. Unfortunately, this architecture does not scale. The overall cost of management network increases super-linearly with the number of servers in a data center. At the same time, massive cabling across racks increases the probability for human errors and prolongs the server deployment latency. This chapter presents a different approach to data center management at the rack granularity, by replacing serial cable connections with low cost wireless links. Wireless networks have intrinsic advantages in this application, as we elaborate in Section 5.1. First, they are cheaper individually than wired alternatives and the cost scales

linearly with the number of servers. For 100,000 servers, the hardware material cost of a wireless solution is only 5.4%-8% the cost of a conventional serial solution (Figure 5.1). Second, they can be integrated onto the motherboard and made physically small to save precious rack space. Third, they can be self-configuring and self-repairing with the broadcast media to prevent human cabling error. Finally, they consume less power. With a small on-board battery, the DCM can continue to function even when the rack experiences a power supply failure. However, whether a wireless DCM can meet the high reliability requirement for data center operation is not obvious for several reasons.

- Wireless links within a rack may not work. The amount of sheet metals, electronics, and cables may completely shield RF signal propagation within racks.
- Although typical traffic on a DCM is low, emergency situations might need to be handled in real time, which could require the design of new protocols.

Power capping is an example of a scenario that requires emergency situation handling. Today, data center operators commonly over-subscribe the power infrastructure by installing more servers to an electrical circuit than it is rated for. The rationale is that servers seldom reach their peak at the same time. By over-subscription, the same data center infrastructure can host more servers compared to a data center with no over-subscription. In the rare event that the aggregated power consumption from all servers approaches the circuit's power capacity, some servers must be slowed down. This can be achieved through dynamic frequency and voltage scaling (DVFS) or CPU time throttling to prevent the circuit breaker from tripping.

Power capping is a critical event that must be handled in real time. A power controller must detect power surge events to send capping commands to corresponding servers. Several round trips must be finished within the trip time of a circuit breaker, usually within a few seconds. This work

studies the feasibility and advantages of using low-power wireless for DCM. We empirically evaluate IEEE 802.15.4 link qualities in server racks across two data centers, to show that the overall packet reception rate is high. We design a real-time event-driven control protocol, called CapNet, for power capping over wireless DCM. The protocol uses distributed event detection to reduce the overhead of regularly polling all nodes in the network. Hence, the network throughput can be used by other management tasks when there is no emergency. When a potential power surge is detected, the controller uses a sliding window and collision avoidance approach to gather power measurements from each node, and then issues power capping commands to a subset of the servers. We deployed a real wireless DCM on 80 wireless nodes to evaluate the real-time performance. We used real data center power traces to emulate a 480 server deployment to show that CapNet can perform as reliably and timely as wired DCM with a fraction of the cost.

5.1 THE CASE FOR WIRELESS DATA CENTER MANAGEMENT

Typical wired DCM solutions in data centers scale poorly with increase in number of servers. The serial-line based point-to-point topology incurs additional costs as more of them are connected together. We compare the costs of the wired DCM to our proposed wireless based solution (CapNet) by considering the cost of the management network, and by measuring the quality of in-rack wireless links.

5.1.1 Cost Comparison with Wired DCM

To compare the hardware cost, we consider the cost of the DiGi switches ($3917/48port [9]), controller cost (approx. $500/rack [10]), cable cost ($2/cable [7]) and additional management network switches ($3000/48port on average [8]). We do not include the labor or management costs for cabling for simplicity of costing model, but note that these costs are also significant with wired

DCMs. We assume that there are 48 servers per rack, and there can be up to 100,000 servers that need to be managed, which are typical for large data centers. For the wireless DCM based CapNet solution, we assume IEEE 802.15.4 (ZigBee) technologies for its low cost benefits. The cost of network switches at the top level layer remains unchanged, but the cost of DiGi can be significantly reduced. We assume $10 per wireless controller, which is essentially an Ethernet to ZigBee relay. For wireless receivers on the motherboard, we assume $5 per server for the RF chip and antenna, since the motherboard controller is already in place. We develop a simple cost model based on these individual costs and compute the total devices needed for implementing management over number of servers ranging from 10 to 100,000 (in order to capture how cost scales with increase in number of servers). We consider solutions across two dimensions 1) Wired vs Wireless, and 2) N-redundant vs 2N-redundant (A 2N redundant system consists of two independent switches, DiGis [9][19] and paths through the management system). Figure 5.1 shows the cost comparison across these solutions. We see that a wired N-redundant DCM solution (Wired-N) for 100,000 servers is 12.5 times the cost of a wireless N-redundant DCM solution (CapNet-N). When we increase the redundancy of the management network to 2N, the cost of a wired solutions (between Wired-2N and Wired-N) doubles, yet the wireless solution increases by 36% for 2N when compared to Wireless-N solution (mainly due to 2N controllers and 2N switches at the top level). The resulting cost of Wired-2N is 18.4 times that of CapNet-2N. Given the significant cost

difference between wired DCM and CapNet, we are motivated to explore whether the technology (IEEE 802.15.4) is indeed feasible for communication within the rack in the next section.

5.1.2 Choice of Wireless - IEEE 802.15.4

We are particularly interested in low bandwidth wireless like IEEE 802.15.4 instead of IEEE 802.11 for a number of reasons. First, the payload size for data center management is small and hence a ZigBee (IEEE 802.15.4) network bandwidth is sufficient for control plane traffic. Second, in WiFi (IEEE 802.11) there is a limit on how many nodes can be supported by an access point in the infrastructure mode. Every access point has to maintain an IP stack for every connection, and this impacts scalability in a dense deployment. Third, to support management features, the data center management system should still work when the rack is unpowered. A small backup battery can power ZigBee device longer at much higher energy efficiency. Finally, ZigBee communication stack is simpler than WiFi so the motherboard (BMC) microcontroller can remain simple. Although we do not rule out other wireless technologies, we chose to prototype with ZigBee in this work.

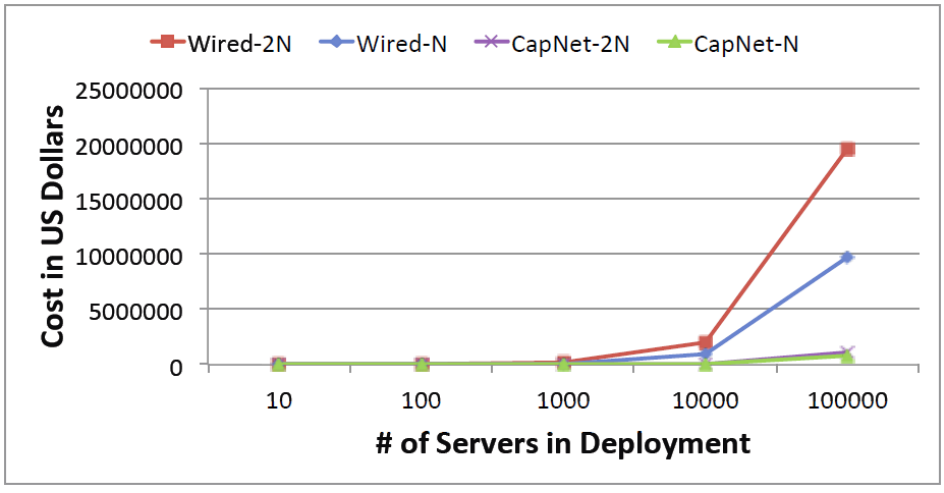


Figure 5.1: System cost comparison and scalability

5.1.3 Radio Environment inside Racks

One of the first questions to answer is whether networks like IEEE 802.15.4 can work within a rack, between a management controller and a stack of servers separated by sheet metal boxes. The sheet metals inside the enclosure are known to weaken radio signal strength, giving a harsh environment for radio propagation. RACNet [60] studied wireless characteristics in data centers, but only across racks when all radios are mounted at the top of the rack. We did not find any previous study that evaluated the signal strength within the racks through servers and sheet metal. Therefore, we first perform an in-depth 802:15:4 link layer measurement study based on in-rack radio propagation in real data center environments.

Setup: The data center used for measurement study has racks that consist of multiple chassis in which servers are housed. In all experiments, one TelosB node (also called a mote) is placed on top of the rack (ToR), inside the rack enclosure. The other motes are placed in different places in a chassis in different experiments. While measuring the downward link quality, the node on ToR is the sender and the nodes in the chassis receive. Then we reverse the sender and the receiver to measure the upward link quality. In each setup, the sender transmits packets at 4Hz. The payload size of each packet is 29 bytes. Through a week-long test capturing the long-term variability of links, we collected signal strengths and packet reception rate (PRR).

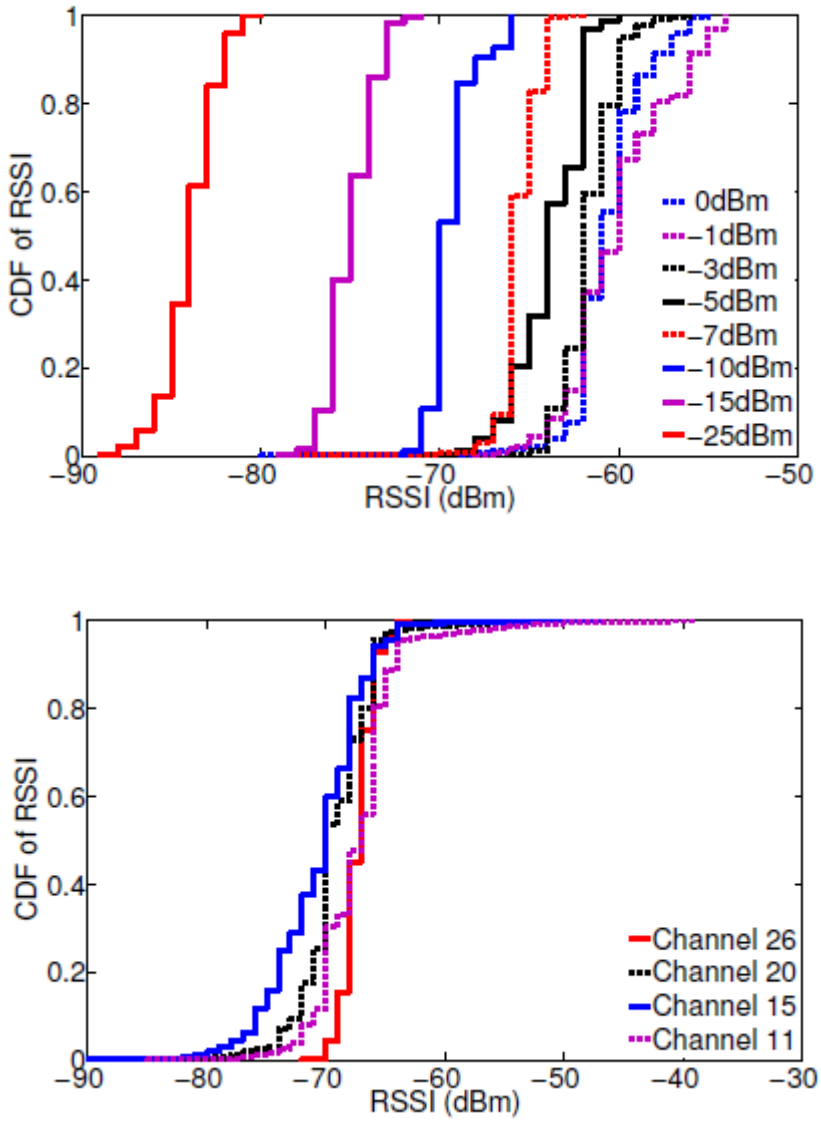


Figure 5.2: CDF of RSSI at a receiver at bottom of sled under a) different Tx Power (Top) and b) different channels (Bottom)

Results. Figure 5.2 (a) shows the cumulative distribution function (CDF) of Received Signal Strength Indicator (RSSI) values from our experiments. The values are measured at a receiver inside the bottom sled for 1000 transmissions from the node on ToR for different transmission (Tx) power using IEEE 802.15.4 channel 26. For -10dBm or higher Tx power, RSSI is greater than -70dBm in 100% cases. RSSI values in ZigBee receivers are in the range [-100, 0]. Previous study

[90] on ZigBee shows that when the RSSI is above -87dBm (approx.), PRR is at least 85%. As a result, we see that signal strength at the receiver in bottom sled is quite strong. Figure 5.2 (b) shows the CDF of RSSI values at the same receiver for 1000 transmissions from the node on ToR on different channels at Tx power of - 3dBm.

Both figures indicate a strong signal strength, and in each experiment the PRR was at least 96%. We observed similar results in all other setups of the measurement study. The measurement study reveals that low-power wireless, such as IEEE 802.15.4, is viable for communication within data center racks and that the link reliability can be sufficient for telemetry purpose. However, a DCM network must also support real-time control scenarios such as power capping, with high data reliability and timeliness of delivery. In the rest of the work, we focus on power capping scenario as an example of critical management functionality in the data center and design a network protocol for real-time control.

5.2 CapNet Design Overview

Power over-subscription and power capping are common practice to improve data center utilization. Power capping demands high reliability and low latency from the DCM network. This section gives an overview of the CapNet design.

5.2.1 The Power Capping Problem

Power infrastructure consists of huge capital investment for a data center, up to 40% of the total cost of a large data center that can cost hundreds of millions of US Dollars [40] to build. Hence, it is desirable to use the provisioned infrastructure to its maximum rated capacity. The capacity of a branch circuit is provisioned during design time, based on upstream transformer capacity during normal operation or UPS/Generator capacity when running on backup power.

When we do over-subscription [23][27][61][70], we allocate servers in a circuit exceeding the rated capacity that the circuit can support, since not all servers reach their maximum power consumption at the same time. Power capping is the mechanism to make sure that the circuit capacity limit is not exceeded during power surges and prevent power loss or potential damage to expensive equipment. Hence, there is a circuit breaker (CB) that trips to protect the expensive equipment. The peak power consumption above the power cap has a specified time limit, called a trip time, depending on the magnitude of over-subscription, as shown in Figure 5.3 for Rockwell Allen-Bradley 1489-A circuit breaker [76]. In the figure, X-axis indicates the magnitude of oversubscription and Y-axis shows corresponding trip times. If the over-subscription continues for longer than the trip time, the circuit breaker will trip and cause undesired server shutdowns and power outages disrupting data center operation. An overload condition under practical current draw trips the CB on a time scale from several hundred milliseconds to hours, depending on the magnitude of the overload.

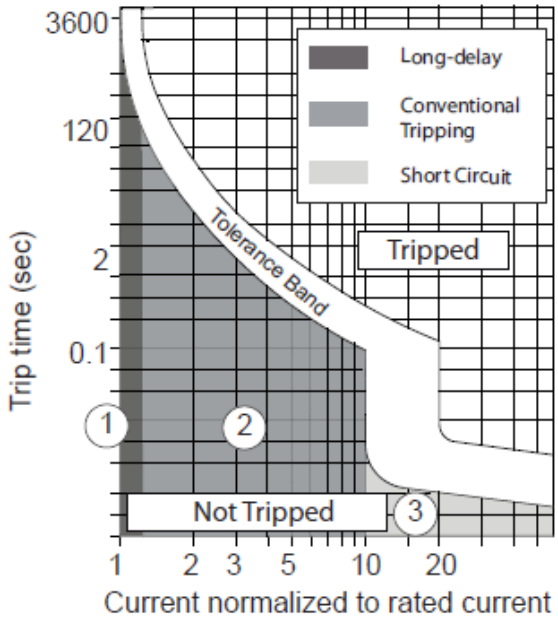


Figure 5.3: Trip curve of Rockwell Allen-Bradley 1489-A circuit breaker at 40C.

To enable power capping for a rack or collection of racks (cluster), an aggregation manager collects power consumption from all the servers and determines the cluster-level aggregated power consumption. If the aggregate consumption is over the cap, the manager generates control messages directing a subset of the servers to reduce their power consumptions through CPU frequency modulation (and voltage if using DVFS) or utilization throttling. In addition, in some graceful throttling policies, the control messages are delivered by the controller to the host OS or VMs, which introduce additional latency due to OS stack [5][61]. In order to avoid abrupt changes to application performance, the aggregation manager may change the power consumption incrementally and require multiple iterations of the feedback control loop before the cluster settles down to below the power cap [96].

5.2.2 Power Capping over Wireless DCM

Using wireless DCM does not fundamentally change the power capping architecture. All servers in a cluster incorporate a wireless transceiver that connects to the BMC microcontroller. The servers are capable of measuring their own power consumption. A cluster power capping manager can either directly measure the total power consumption using a power meter, or, to achieve fine-grained power control, aggregates the power consumption from individual servers. We focus on the second case due to its flexibility. When the aggregated power consumption approaches the circuit power capacity, the manager issues capping commands over wireless links to individual servers. The main difference comparing to a wired DCM is the broadcast wireless media and challenge of scheduling the order of communication to achieve real-time message delivery.

To reduce extra coordination and to enable spatial spectrum reuse, we assume a single IEEE 802.15.4 channel for communication inside a cluster. Using multiple channels, multiple

clusters can run in parallel. Channel allocation can be done using existing protocols that minimize inter-cluster interference (e.g. [77]). For protocol design, we focus on a single cluster of servers at first, and show its applicability through sensitivity experiments in the later part of this chapter.

5.2.3 A Naive Periodic Protocol

A naive approach for a fine-grained power capping policy is to periodically collect power consumption readings from individual servers. The period must be smaller than an upper bound of the total aggregation latency. The manager periodically computes the aggregate power. Whenever the aggregate power exceeds the cap, it generates a control message and broadcasts the message after suspending all other communications. To handle broadcast failures, it can repeat the broadcast γ times. If the control algorithm requires η iteration to reach the settling time, then after the first round capping control command is executed, the controller will again run the aggregation cycle to collect all readings from servers, and reconfirm that capping was done correctly. The procedure continues up to (η - *1*) more iterations. Upon finishing the control, it resumes the periodic aggregation again.

5.2.4 Event-Driven CapNet

Since power capping is a rare event, the naive periodic protocol is an overkill as it can saturate the wireless media by sending large number of messages periodically. This would result in other telemetry messages not getting access to network resources. CapNet employs an event-driven policy that is designed to trigger power capping control operation only when a potential power capping event is predicted. Due to the rareness and emergency nature of power surge, the network can also suspend other activities to handle power capping. It provides real-time performance and a sustainable degree of reliability without consuming a lot of network resource.

CapNet's event-driven capping protocol runs in 3-phases: distributed event detection, power aggregation, and control. The event detection phase generates an alarm implying a potential power surge that may require power capping. Only if this phase generates an alarm, CapNet runs the second phase which invokes the power aggregation protocol. If aggregated power exceeds a predefined threshold, the control algorithm on the controller broadcasts control messages requesting a subset of the servers to be capped. The servers react to the capping messages by DVFS or CPU throttling. The details of the protocol is explained in the next section.

5.2.5 Power Capping Protocol

Data center power capping is a rare event. Data centers are required to cap only in 1% to 5% of all cases [5] when they are over-provisioned. Therefore, an ideal protocol should generate significant traffic only when a power surge occurs that may require power capping. This motivates the design of an event-driven protocol.

While a global detection can be implemented by monitoring at the branch circuit level, it cannot support fine-grained and flexible power capping policies such as those based on individual server-priority or allocating power caps to individual servers based on their power consumption. When a server observes a local power surge based on its own power reading, it triggers the collection of the power consumption of all the servers to detect a potential surge in the aggregate power consumption of cluster. If a cluster-level power surge is detected, the system initiates a power capping action. The event detection policy is designed based on the characteristics of the power usage in real data centers.

Data Sets: We use workload demand traces from multiple geo-distributed data centers run by a global corporation over a period of six consecutive months. These clusters run a variety of workloads including Web-Search, Email, Map-Reduce jobs, and several other online cloud

applications, catering to millions of users around the world. The data that was collected to compose these workloads are taken from 6 consecutive months in every 2 minute intervals. While we recognize that full system power is composed of storage, memory and other components, in addition to CPUs, several previous works show that a server's utilization is highly correlated to its power consumption [12][13][23]. Hence, we use server's CPU utilization as a proxy for power consumption in all trace analyses and experiments. Any rack-level or cluster-level aggregate power is determined as the average utilization over all servers under that rack or cluster. The next subsection explains CapNet's protocol design and procedures through which it avoids false positives and false negatives through a three-phase design.

5.2.5.1 Event-Driven Protocol

The event-driven protocol consists of 3 phases as illustrated in Figure 5.4: detection, aggregation, and control. Detection phase determines if some server in the rack is over the threshold (cap) and, if so, a server generates an alarm. There may be situations where some server is over the cap, but the aggregate utilization is still under the cap. Such false positive cases are corrected through the aggregation phase, where the controller determines the aggregate power consumption. The control phase is executed only if the controller gets confirmation that it was not a false alarm. The impact of a false positive case is that the system runs into the aggregation phase which incurs additional wireless traffic. No power capping control will be performed if the power capping manager determines the aggregate power consumption of the cluster has not exceeded the power cap after the aggregation phase.

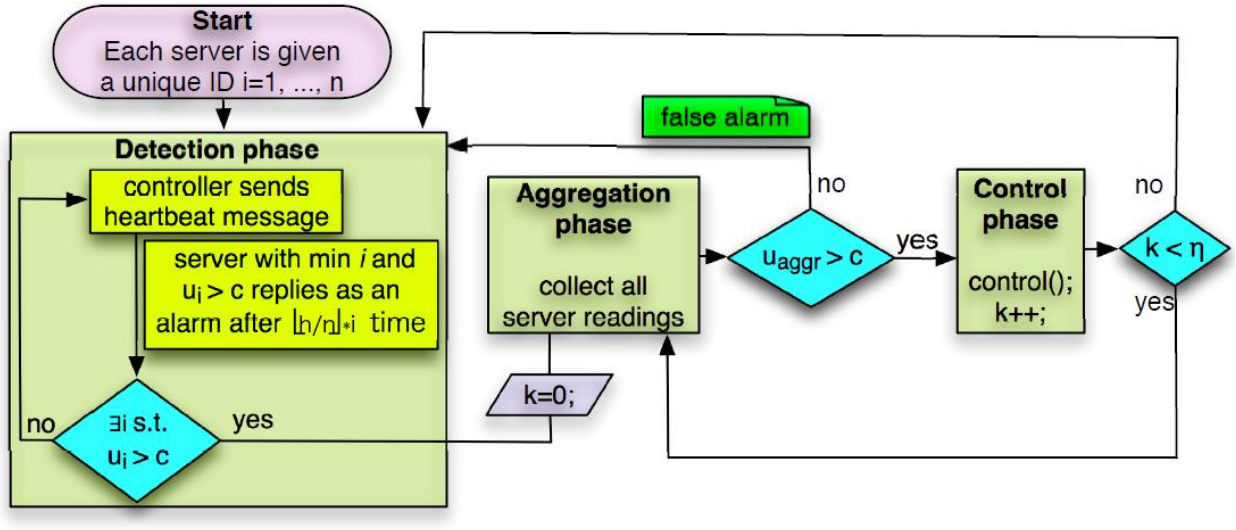


Figure 5.4: CapNet's event-driven protocol flow diagram

Our event based protocol works as follows. Each server is assigned a unique ID *i*, where *i* = 1, 2, .. , n. The Controller broadcasts a heartbeat packet at every *h* time units called *detection interval*. The detection interval of length *h* is slotted among *n* slots, with each slot length being *floor(h/n)*. The value of h is selected in a way so that a slot is long enough to accommodate one transmission and its acknowledgement. After receiving the heartbeat message, the server clocks are synchronized.

Detection phase: Each node *i*, 1 <= i <= n, takes its sample at the i-th slot i.e. at time floor(h/n) * i in the detection phase. If its reading is over the cap, it generates an alarm and sends the reading to the controller as an acknowledgement of the heartbeat message. Otherwise, it ignores the heartbeat message, and does nothing. If no server sends an alarm in the detection phase (or the sent alarm is missed due to transmission failure), the controller resumes the next phase when the current phase is over. If no servers are over the peak, there is no (alarm) transmission in the detection

phase. If the controller receives any alarm, it generates a control message (repeated up to γ times for reliability) that informs the servers that the detection phase is over. It then starts an aggregation phase and determines the aggregate power consumption, denoted by u_{aggr}, using the aggregation protocol presented as follows.

Aggregation phase. To minimize aggregation latency, CapNet adopts a sliding window based protocol to determine aggregate power. The controller uses a window of size ω. At any time, it selects ω servers (if there are fewer than ω servers whose readings are not yet collected, then it selects all of them) in a round-robin fashion who will send their readings consecutively in the next window. These ω server IDs are ordered in a message. In the beginning of the window, the controller broadcasts this message, and starts a timer of length $\tau_d + \omega\ \tau_u$ after the broadcast, where τ_d denotes the maximum downward communication time (i.e., the maximum time required for a controller's packet to be delivered to a server) and τ_u denotes the maximum upward communication time (server to controller). Upon receiving the broadcast message, any server whose ID is in order i, $1 <= i <= \omega$, in the message transmits its reading after $(i - 1)\ \tau_u$ time. Other servers ignore the message. If the timer fires or if the packets from all ω nodes are received, the controller creates the next window of ω servers that are yet to be scheduled or whose packets were missed (in the previous window). A server is scheduled in at most γ consecutive windows to handle transmission failures, where γ is the worst-case ETX (expected number of transmissions for a successful delivery) in the network. The procedure continues until all server readings are collected or if there is no server that was already retried γ times.

Control phase. Upon finishing the aggregation phase, if $u_{aggr} > c$, where c is the cap, the controller starts the control phase. The control phase generates a capping control command using a control algorithm, and then the controller broadcasts the message. To handle broadcast failures, it repeats

the broadcast γ times (since the broadcast is not acknowledged). If the control algorithm requires η-iteration, then after the capping control command is executed in the first round, the controller will again need to run the aggregation phase, and reconfirm that capping was done correctly. The procedure iterates up to (η - 1) more iterations. Upon finishing the control phase, or following an aggregation phase that started upon a false alarm, the controller resumes the detection phase again. Our protocol is particularly effective in the presence of strong intra-cluster synchrony that exists in enterprise data centers as observed in trace analysis [94]. In absence of intra-cluster synchrony in power peaks, CapNet will not cause unnecessary power capping control or more wireless traffic than a periodic protocol.

5.3 EXPERIMENTS

In this section, we present the experimental results of CapNet. The objective of the experiments is to evaluate the effectiveness and robustness of CapNet in meeting the real-time requirements of power capping under realistic data center settings. We first explain the implementation of CapNet, and describe the experimental setups followed by the detailed results.

5.3.1 Implementation

The wireless communication side of CapNet is implemented in NesC on TinyOS [91] platform. To comply with realistic data center assumptions, we have implemented the information processing and control management at the aggregation manager side. In our current implementation, wireless devices are plugged to the servers directly through their serial interface.

5.3.2 Experimental Setup

5.3.2.1 Experimental Methodology

We experiment with CapNet using TelosB motes for wireless communication. We use a total of 81 motes (1 for controller, 80 for servers). The motes communicate on behalf of the servers. When we experiment with more than 80 servers to test scalability, one mote emulates multiple servers and communicates for them. For example, when we conduct experiments with 480 servers, mote 1 emulates first 6 servers, then mote 2 emulates the next 6 servers, and so on. We conduct experiments to test the network portion of CapNet design, and emulate the control part. We place all 80 motes in racks. The controller node is placed on ToR and connected through its serial interface to a PC that works as the manager. No mote in the rack has direct line of sight with the controller. We use workload demand traces from an enterprise data center over a period of six consecutive months.

We use these server readings, and run CapNet in a trace-driven fashion. We emulate power reading from traces and send this data through the wireless motes. While the data traces are of 6 month long, our experiment does not run for actual 6 month period. When we take a subset of those traces, say for 4 weeks, the protocols skip the long time intervals where there is no peak by doing a look-ahead. For example, when we know that there is no peak between time t_1 and t_2, the protocols skip the times between t_1 and t_2. We artificially speed up the execution of the experiment, in order that we do not wait for the real time range of the traces, and finish our experiments in several days instead of 4 weeks.

5.3.2.2 Oversubscription and Trip Time

We run the experiments for different caps in various experiments, and we determine the corresponding trip times based on aggregate utilization from the Rockwell Allen-Bradley 1489-A industrial circuit breaker (Figure 5.3 shows the trip curve). X-axis shows the ratio of current draw to the rated current, and Y-axis shows the corresponding trip time. The trip curve is shown as a tolerance band. The upper curve of the band indicates upper bound (UB) trip times above which the circuit breaker trips, implying that the circuit breaker will trip if the duration of the current is longer than the UB trip time. The lower curve of the band indicates lower bound (LB) trip times under which the circuit breaker operates as expected. This band between the two curves is an area where it is non-deterministic if the circuit breaker will trip. The magnitude of oversubscription is defined as the ratio of aggregate utilization to the cap. Therefore, from the trip curve, the values in X-axis indicate the oversubscription magnitude. LB trip time could be treated as a conservative bound. In our experiments we use both LB and UB to verify the robustness of CapNet. The circuit breaker has three types of trip time behaviors that are shown as different regions below the tolerance band. Region 1 is the long-delay tripping zone with the oversubscription magnitude between 1 and 1.35, and trip time between minutes to hours. Region 2 is the conventional tripping zone with magnitude between 1.35 and 10. Region 3 is the instantaneous tripping zone (ratio > 10) that is designed to handle short-circuits.

5.3.2.3 CapNet Parameters

For all experiments, we use channel 26 and Tx power of -3dBm. The payload size of each packet sent from the nodes that emulate servers is 8 bytes. The size of payload is sufficient since these nodes need to send power consumption reading of a single server. The maximum payload size of each packet sent from the controller is 29 bytes which is the maximum default size in IEEE

802.15.4 radio stack for TelosB motes. This payload size is set large since those packets need to contain the schedules as well as control information. For the aggregation protocol, window size ω is set to 8. A larger window size can reduce aggregation latency, but requires the payload size of the controller's message to be larger (since the packet contains ω node IDs indicating the schedule for next window). In the aggregation protocol both τ_d and τ_u were set to 25ms. The controller sets its timeout using these values. These values are relatively larger compared to the maximum transmission time between two wireless devices. The time required for communication between two wireless devices is in the range of several milliseconds. But in our design the controller node is connected through its serial interface to a PC. The TelosB's serial interface does not always incur a fixed communication latency between the PC and the mote. Upon experimenting and observing a wide variation of this time, we have set τ_d and τ_u to the 25ms value.

5.3.2.4 Control Emulation

In our experiments, we emulate the control portion of the protocol. We assume that one packet is enough to contain the entire control message. To handle control broadcast failure, we repeat control broadcast for $\gamma = 2$ times. Our extensive measurement study through data center racks indicated that this is also the maximum ETX for any link between two wireless motes. Upon receiving the control broadcast message, the nodes generate an OS level latency and hardware level latency. We use the maximum and minimum OS level and hardware level time required for power capping experimented on three servers with different processors each running Windows Server 2008 R2: Intel Xeon L5520 (frequency 2.27GHz, 4 cores), Intel Xeon L5640 (frequency 2.27GHz, dual socket, 12 cores with hyper-threading), and an AMD Opteron 2373EE (frequency 2.10GHz, 8 cores with hyper-threading). The ranges of OS level and hardware level latencies are

10-50ms and 100-300ms respectively [5]. We generate OS and hardware level latencies in a single iteration by using a uniform distribution in this range.

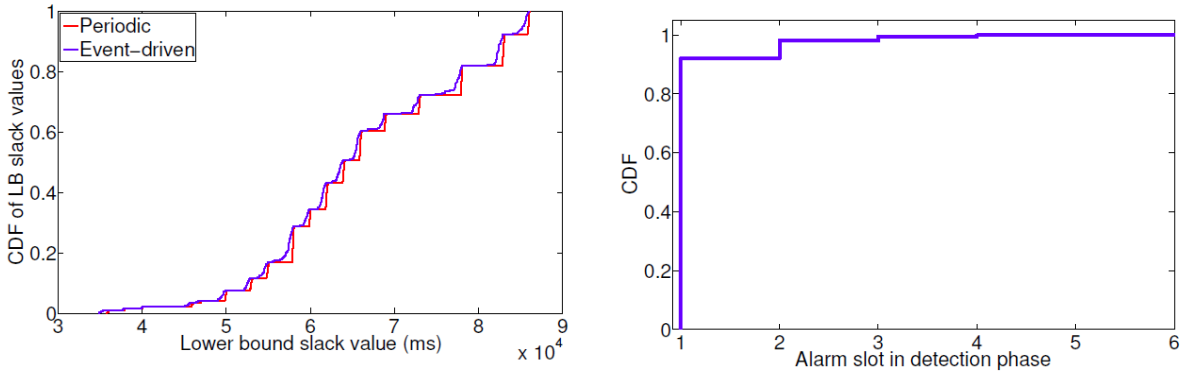
5.3.3 Results

Now we present our experimental results with CapNet's event-driven protocol. First we compare its performance with a periodic protocol and a representative CSMA/CA protocol. We then analyze its scalability in terms of number of servers. We run experiments only for the simple case, where a single iteration of control loop can achieve a sustained power level, and then we also analyze scalability in terms of number of control iterations, where multiple iterations are needed to settle to a sustained power level. We have also run our protocol under different caps and in presence of interfering clusters. In all experiments, detection phase length, *h*, was set to 100 * *n* ms, where n is the number of servers. We set this value because this makes each slot in the detection phase equal to 100ms, which is enough for receiving one alarm as well as for sending the corresponding message from the controller to the servers informing the suspension of current detection phase (when an aggregation phase is in order). Setting a larger value reduces the number of cycles of detection phase, but also reduces the granularity of monitoring. In the results, slack is defined as the difference between the trip time thresholds and the total latency required for power capping. That is, a negative value of slack implies a deadlines miss. An oversubscription is associated with both a lower bound (LB) and an upper bound (UB) trip time (Figure 5.3). Hence, we use LB slack and UB slack to define the slack calculated considering LB trip time and UB trip time, respectively. In our results, there are some cases where timing requirement can be loose, while there are cases where the requirement is very tight, and the results are shown for all cases. We particularly care for tight deadlines, and want to avoid any deadline misses.

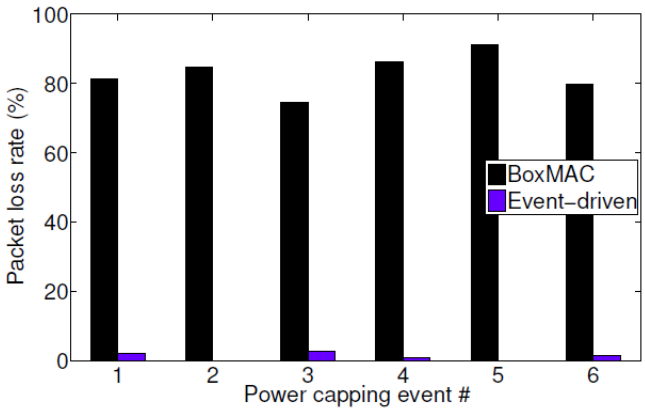
5.3.3.1 Performance Comparison with Base Lines

Figure 5.5 presents the results of our experiments with 60 servers running single-iteration control loop. We used 4 weeks long data traces for this rack. We set the 95-th percentile of all aggregate utilization values of all data points in every 2 minute intervals as its cap. The upper bound of total aggregation delay (L_{aggr}) was set as the period of the periodic protocol.

Figure 5.5 (a) shows the lower-bound (LB) slacks for both the event-driven protocol and the periodic protocol. The figure only plots the CDF for the cases where the oversubscription was above 1.5 for better resolution, as the slack is quite large for small magnitudes of oversubscription (which are not of interest). Since UB trip times are easily met, these results are obvious and not very interesting. The non-negative LB slack values for each protocol indicates that it easily meets the trip times. Hence using nonstop communication (i.e., the periodic protocol) does not benefit in practice.



(a) (b)



(c)

Figure 5.5: Performance of event-driven protocol on 60 servers for 4 weeks (a) CDF of lower bound stack (b) CDF of alarm slots (in detection phase) and (c) Packet Loss Rate

While the slacks in event-driven protocol are shorter than those in the periodic protocol because the event-driven protocol spends some time in the detection phase, in 80% cases event-driven protocol can provide a slack of more than 57.15s while the periodic protocol provides 57.88s. The difference is not significant as shown in Figure 5.5 (b), because in 90% cases among all power capping events the alarms were generated in the first slot of the detection cycle. This result actually conforms to previous trace analysis results [94] that leads to the conclusion that when the aggregate power is over the cap, a server has very high probability to generate an alarm. Therefore, the first server (i.e., the server that is assigned slot 1 of the detection cycles) in most cases generates the alarm. Only in 10% cases, the alarm was generated after the first slot of the detection phase. In 100% cases of the capping events the alarms were generated within the 6-th slot, although the phase had a total of 60 slots (for 60 servers, one slot per server).

We also evaluate the performance when BoxMAC (a CSMA/CA based protocol) is used for power capping communication under default settings in TinyOS [91]. Figure 5.5 (c) shows that

BoxMAC experiences packet loss rate over 74% while performing communication for a power capping event. This happens because all 60 nodes try to send at the same time, and the back-off period in 802.15.4 CSMA/CA under the default setting is too short, which leads to frequent repeated collisions. Since we lose most of the packets, we do not consider latency under CSMA/CA. Increasing the back-off period reduces collisions but results in long communication delays. In subsequent experiments, we exclude comparison with CSMA/CA as it does not fit for power capping.

CapNet's event-driven protocol may also generate false alarms. In this experiment, among the total alarms, about 45.56% alarms are false alarms while the remaining 54.44% alarms were real power capping events. Typically alarms are generated rarely, since power capping is triggered only in approximately 5% of all cases. The biggest advantage is that in the rest of the times detection phases simply encounter no alarms, i.e., no server makes any transmission. Thus all those cycles go without any transmission from the servers. In this experiment, 94.16% of the total detection phases did not have any transmission from the servers. Therefore, if we compare with the periodic protocol that needs to continue communication always in the network, the event-driven protocol suppresses transmissions at least by 94.16% while the real-time performance of two protocols are similar.

5.3.3.2 Scalability in Terms of Number of Servers

In our data traces each rack has at most 60 active servers. To test with more servers, we combine multiple racks in the same cluster since they have similar pattern of power consumption [94]. For sake of experimentation time, in all subsequent experiments we set cap at 98-th percentile (that would result in a smaller number of capping events). The lower bound slack distribution are shown in Figure 5.6 for 120, 240, and 480 servers by merging 2, 4, and 8 racks, respectively (for

single iteration capping). The results show that for single iteration the deadlines are easily met for even 480 servers (since in each setup, 100% of all slack values are positive). Hence we move to more complicated cases, and in the next experiments we increase the number of iterations.

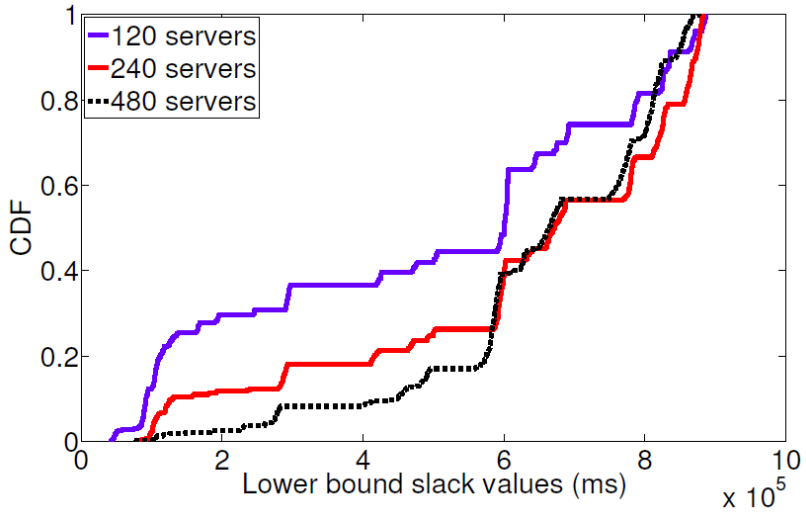
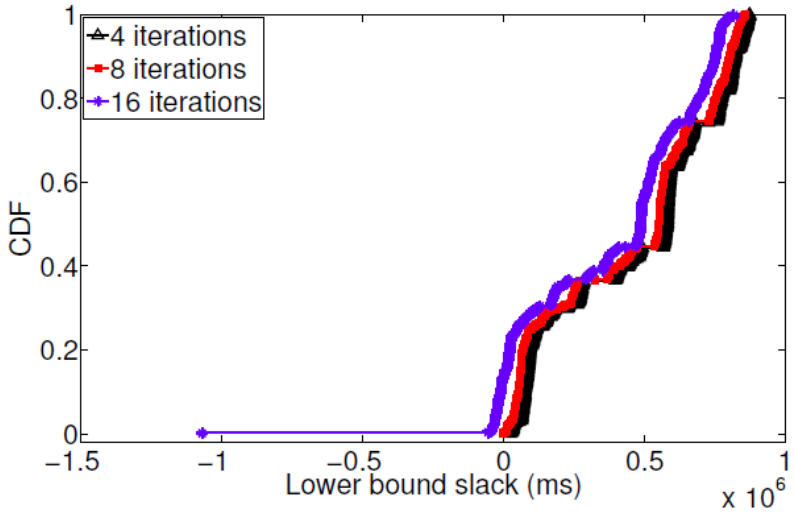


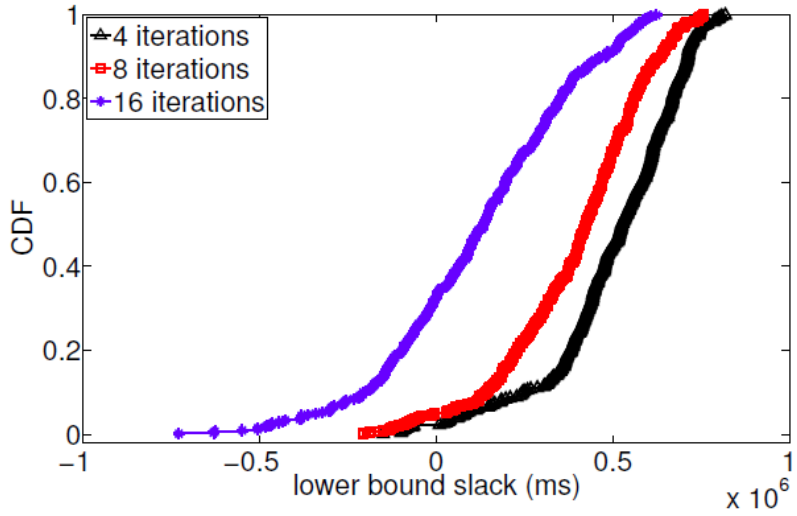
Figure 5.6: CDF of lower bound slack under event-driven protocol for increasing number of servers

5.3.3.3 Scalability in Number of Control Iterations

We consider another variation of our experiment, where multiple iterations of control loop are required to settle to a sustained power level. The number of iterations required for the rack-level loop as experimented in [96] can be up to 16 in the worst case. Hence, we now conduct experiments considering multiple numbers of control iterations (up to 16 as the worst case consideration). We plot the results in Figure 5.7 for different numbers of servers under various number of iterations.



(a)



(b)

Figure 5.7: LB slack for multi-iteration capping under event-driven protocol – (a) LB slack for 120 servers and (b) LB slack for 480 servers

As shown in Figure 5.7 (a), for 120 servers under 16-iteration case, we have 13% cases with negative slack meaning that the LB trip times were missed. However, the UB trip times were

easily met. We consider a conservative setup here because using 16-iteration as well as trying to meet the lower bound of trip times are both very restrictive considerations. For 120 servers under 8 iterations, in 0.13% cases slacks were negative. However, in 80% cases the slacks were 92.492s, 66.694s, and 22.238s for 4, 8, and 16 iterations, respectively indicating that the trip times were easily met, and the system could oversubscribe safely. For 4-iteration case, the minimum slack was 23.2s.. For 480 servers (Figure 5.7 (b)), 98.95%, 97.86%, 94.93%, and 67.2% LB trip times were met for 2, 4, 8, and 16 iterations, respectively. Figure 5.8 shows the miss rate for the different number of servers across different number of iterations. For 240 nodes, we miss deadlines in 5% cases under 8-iteration experiment and 13.94% cases under 16-iteration experiment. For all cases we met the upper bound trip times. The above results show that, even for 480 servers in a cluster, the latencies incurred in CapNet for power capping remain within the conservative latency requirements in most cases.

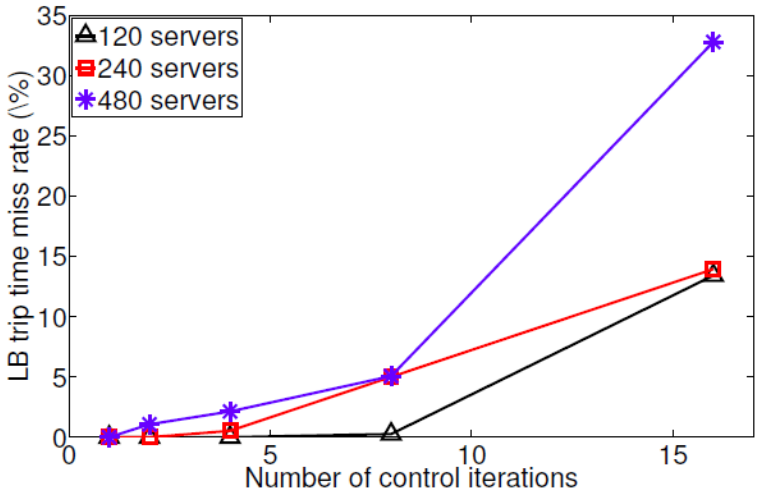
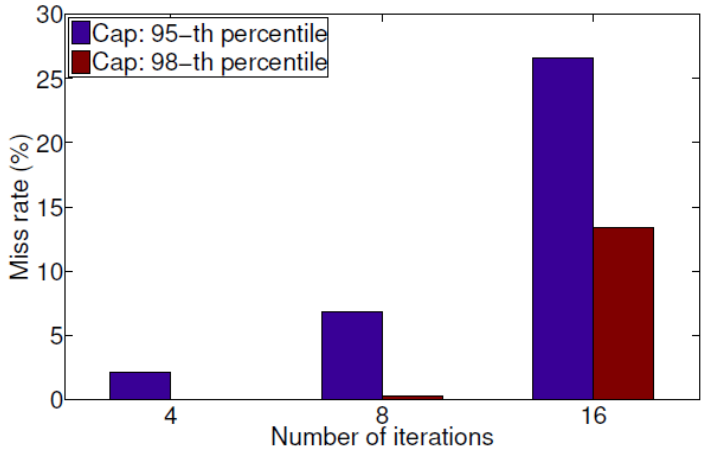
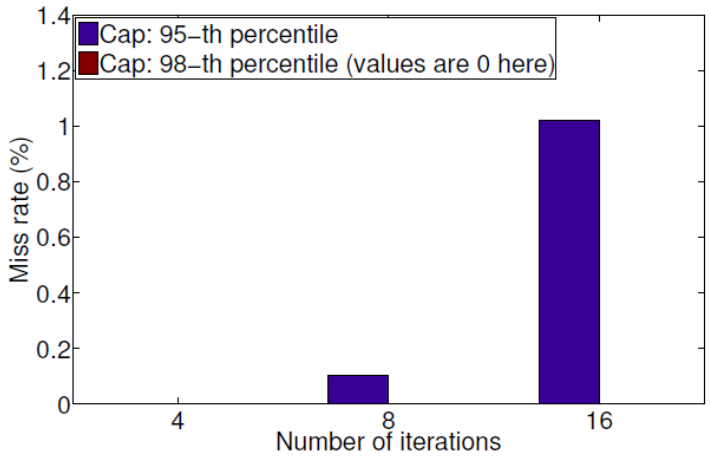


Figure 5.8: Miss rates for varying server counts for multi-iteration capping

5.3.3.4 Experiments under Varying Caps



(a)



(b)

Figure 5.9: Sensitivity to different power cap levels (120 servers, 4 weeks trace) a) LB miss rates and b) UB miss rates

In all experiments we have performed so far, CapNet was able to meet UB trip times. Now we make some setup changes to encounter scenarios where UB trip times can be smaller, by making oversubscription magnitude higher. For this purpose, we now decrease the cap, to decrease the trip times so as to simulate scenarios that miss upper bound trip times to in order to validate

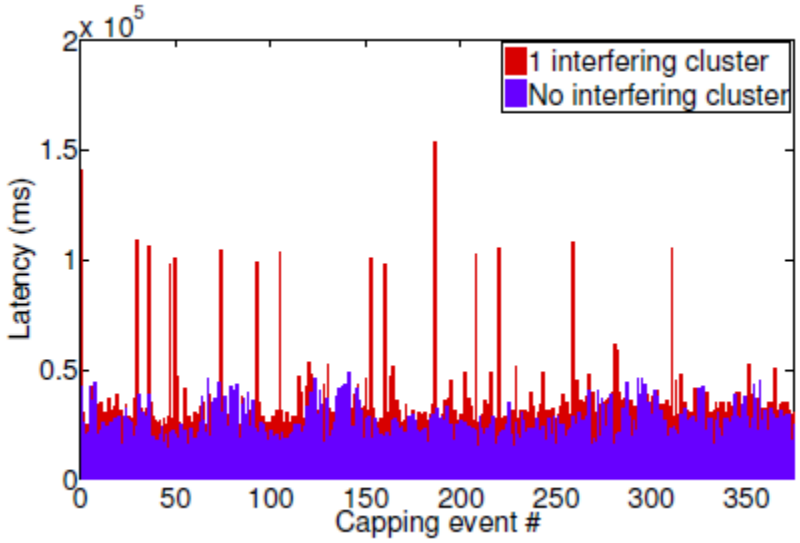
the robustness of the protocol. We set the 95-th percentile of aggregate utilization as the cap in our experiments. This would give the previous capping events shorter deadlines since a smaller cap implies a larger magnitude of oversubscription. We ran these experiments with 120 servers and their 4 week data traces.

Figure 5.9 shows that we miss more LB trip times and miss some UB trip times as well since the deadlines have become shorter. However, UB trip times are missed only in 0.11% and 1.02% cases under 8 and 16 iterations, respectively, while LB deadlines were missed in 2.14%, 6.84%, and 26.56% cases under 4, 8, and 16 iterations. These results demonstrate the robustness of our solution for larger magnitudes of oversubscription.

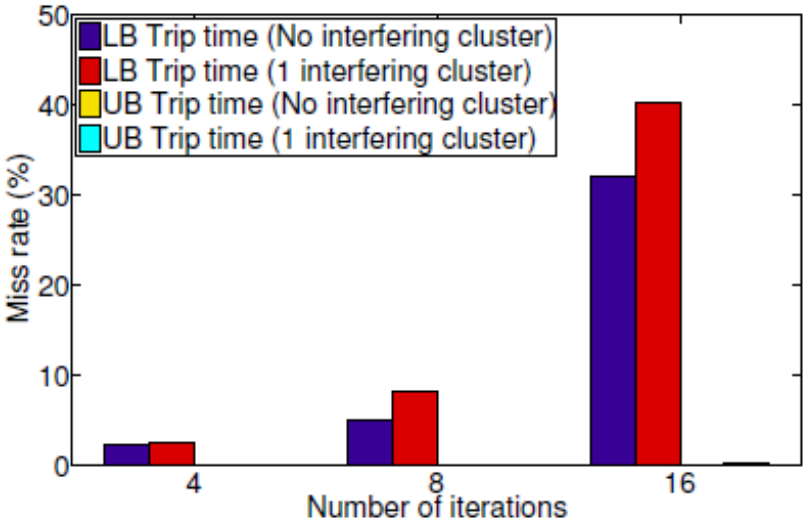
5.3.3.5 Experiments in Presence of Multiple Clusters

In this section we perform an experiment to observe the performance of the event-driven protocol under an interfering cluster. We mimic an interfering cluster of 480 servers in the following way. We select a nearby cluster and place a pair of motes in the rack: one at the ToR and another inside the rack. We set their transmission power at maximum (0dBm). The mote at the ToR represents its controller and carries on a pattern of communication like a real controller to control 480 servers. The mote inside the rack responds as if it were connected to each of 480 servers. Specifically, the controller executes a detection phase of 100 * 480ms, and the receiver node in the rack randomly selects a slot between 1 and 480. On the selected slot, it generates an alarm with probability 5% since capping happens in no more than 5% cases. Whenever the controller receives the alarm, it generates a burst of communication in the pattern like what it would have done for 480 servers. After finishing this pattern of communication it resumes the

detection phase. We run the main cluster (system used for experiment) using 4 weeks data traces, and plot the results in Figure 5.10.



(a)



(b)

Figure 5.10: Sensitivity to interfering cluster - (a) Capping latencies observed and (b) Miss rates

Figure 5.10 (a) shows the latencies for different capping events, both under interference and without interference (when there was no other cluster). Under interfering cluster, the delays mostly increase. The event-driven protocol experiences packet loss and uses retransmission for those losses, thereby increasing network delays. While the maximum increase was 124.63s, in 80% of the cases the increase was less than 15.089s. We noticed that the large increase happened because of the loss of detection message. At the beginning of capping if a detection message is lost, then the next cycle starts after 48s. Figure 5.10 (b) presents the miss rate for the events under interfering clusters. Among 375 events, 4 broadcasts were lost at some server even after 2 repetitions, resulting in control failure in 1.06% cases. This value became 0 in multi-iteration cases. For multi-iteration cases, at least one of the control broadcasts was successful. However, as the delay due to transmission failure and recovery increased in detection phase, we experienced capping failure. For 16-iteration case, we missed the upper bound of trip time in 40.27% cases and lower bound of trip times in 32.08% cases. For 4 iteration case, the miss rate was 5.06% and 8.26% only. And for 2-iteration they are only 2.13% and 2.4% which are very marginal. The result indicates that even under interference, CapNet demonstrates robustness in meeting the real-time requirements of power capping.

5.4 ENGINEERING LIMITATIONS

CapNet may face a number of engineering challenges in real world deployments. While our work addresses feasibility, protocol design and implementation, challenges like security, EMI and fault tolerance needs to be addressed.

Fault Tolerance. One important challenge is handling the failure of power capping manager in a cluster. To address this challenge, power capping managers can be connected among themselves

either through a different band or through a wired backbone. As a result, when some controller node fails, a nearby controller can take over its servers. This work focuses on communication within a single cluster. DCM fault detection, isolation, and node migration need to be studied in future work.

Security. Another challenge is the security of the management system itself. Since the system relies on wireless control, it is theoretically possible that someone might be able to maliciously tap into the wireless network and take control of the data center. There are two typical approaches to handle this security issue: First, the signal itself should be attenuated by the time it reaches outside the building. We can identify secure locations inside the data center from which the controller can communicate, and identify a signature for the controllers which would be known to the server machines. We can also use possible shielding methodologies that attenuate the radio propogation within the boundaries of the data center. This is common practice especially with respect to intentional attacks and is part of security systems in critical data center facilities [55]. However, compared to the physical approach, there also exist software based approaches to handle security. It is possible to encrypt wireless messages, for example, using MoteAODV (+AES) [2]. The performance impact of using AES on top of the Mote infrastructure is critical for designing time-sensitive responses from this system. The performance penalty due to encryption might be impactful when we use AES for all management communication. Hence in real implementations physical delimitation is recommended. We could also use an hybrid approach where security is achieved by both physical isolation and also through AES for certain communications with higher security levels. There are previous research works in this domain especially with SCADA architectures [48] that could be leveraged for implementing security practices for data centers when wireless is used inside data centers.

EMI & Compliance. While less emphasized in research studies, a practical concern of introducing wireless communications in data centers is that they do not adversely impact other devices. There are FCC certified IEEE 802.15.4 circuit design available (e.g. [63]). Previous work has also used WiFi and ZigBee in live data centers for monitoring purposes [60].

5.5 SUMMARY

In this chapter, we make a case for implementing wireless based management for data centers. We have designed CapNet, a low-cost, real-time wireless sensor network for data centers and validated its feasibility for power capping. We deployed and evaluated CapNet in two data centers using 6-month long data traces from real data centers. We show that CapNet can meet the latency requirements of power capping for a wide variety of scenarios and present engineering limitations that needs to be addressed for a real deployment. We also show that CapNet has potential for an order of magnitude (12 times-18 times) lower cost than existing wired management solutions. Our novel work in this area represents a promising step towards using low-cost commodity wireless networks for time-critical, close-loop control in data centers.

6 NEW FAILURE TRENDS IN DATA CENTERS

Large enterprises house hundreds of thousands of servers in globally distributed data centers to support the growth in online services and cloud computing. The cost of providing such services is a significant expenditure for these enterprises, and hence infrastructure efficiency is a key value consideration. As with any large system, a data center experiences failures at different levels of its architecture, ranging from critical power infrastructure to server components. The data center itself becomes a warehouse-scale computer [41] and failures can cause service unavailability for hosted services. Among component failures, some result in actual replacement of hardware components. However, there is another class of data center failures that does not result in actual replacement while still causing service downtime. This is an important class of failures to look at, because they can cause service unavailability, and incur the cost for a technician to identify and address the reason for failure. We classify these failures as ***soft failures***. This term has already been used in literature for transient failures in CPUs and memory (for example, particle strikes) [66], but it is uncommon in the context of large data centers. In this work, we analyze failure data collected from large-scale data center deployments housing tens of thousands of commodity servers. We then identify trends in fixes made for the failures and highlight the soft failures issue. We then propose methodologies that can help increase the overall availability for the data center by masking soft failures.

Previous studies have looked at hardware errors in large data centers [71][78][87][84]. However, there has been minimal work that looked at soft failures. This study characterizes soft failures in large clusters of machines using data collected over a year of observation. We attempt to answer some non-trivial questions including, the downtime experienced by a machine when it

experiences a soft failure, probability of a machine having a soft failure once it has already experienced soft failures, the categories of most likely fixes following a soft failure, average number of days to the next failure event following a soft failure, and time histograms describing other interesting trends.

Following our characterization of soft failures, we present a discussion on data center architecture and services that these failures affect, and propose possible ways of addressing these failures. Since the underlying root causes of such failures are not fully understood, our objective in this early work is to highlight the issue of soft failures, and identify possible research avenues for potentially significant solutions to this problem.

6.1 SOFT FAILURE CHARACTERIZATION

6.1.1 Percentage of Soft Failures

Figure 6.1 shows the percentage of soft failures in a population of machines belonging to the same cluster, which contains tens of thousands of machines. These clusters include servers hosting representative large scale online services like Websearch, Cosmos and Email [59]. The servers are designed for scale-out scenarios with cost optimization around dual processors and enterprise SATA disk drives. The automated management layer [53] classifies failures and pushes a machine from healthy operational state to a repair state, where it is triaged before actual repair is done.

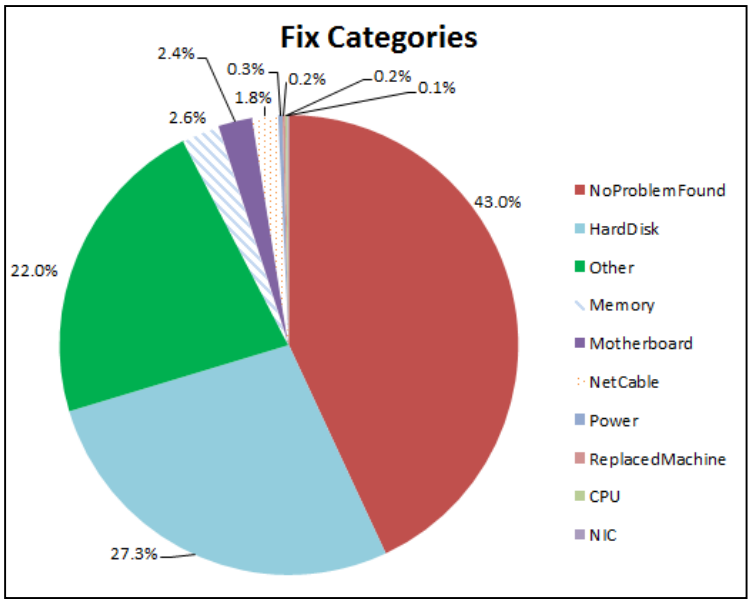


Figure 6.1: Percentage of Soft failures in data centers

As we can see from the figure, 43% of the total fixes performed in this cluster belong to the category *No Problem Found*, which assigns these failures to a category that describes machines reporting a problem, even when no actual failures occurred. In most of these cases, hard-power recycling (as contrasted to an autonomous OS-driven power recycle) fixes some portion of the problem. In other cases, physically reseating the hard disk drives or the memory DIMMs resolves the issue due to transient configuration issues with these components. Figure 6.1 also shows other categories of actual failures in a data center; the next biggest category is due to hard disk drives. Hard disk drive failures is one of the most common cases of data center server failures [71]. A significant portion of the fixes fall under *Other*, which includes firmware fixes, fan cages, SATA cables, and other components in servers that are not listed. Other component failures are minimal compared to these big contributors, and the most common error cause involves no actual hardware issue.

6.1.2 Downtime due to Soft Failures

To understand downtime statistics, it is crucial to understand the service model in large data centers. Most data centers that host online and cloud computing services are highly redundant. It is rare that an application has a mission-critical dependency on one server or just a single instance application model (in which one instance failure means that the single instance and data associated

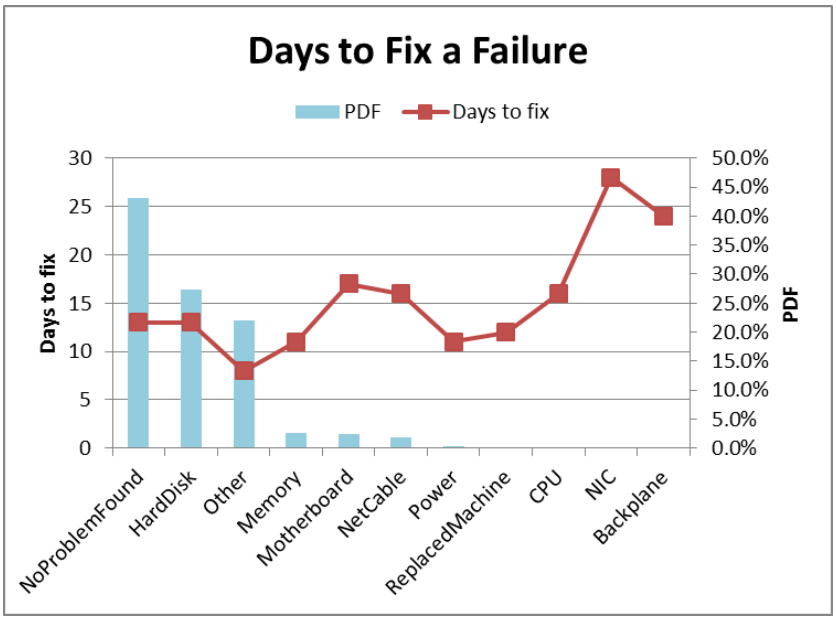


Figure 6.2: Average days for fixing a failure

with it have to be recovered). Hence, fixing failures in a data center occurs by a batch processing model, where several failures are fixed at a time. This saves the amount of time a technician has to visit a data center, and decreases the load when scaling up to several thousands of servers.

Figure 6.2 shows the average number of days it takes to fix failures belonging to different categories. This measure could be taken as a relative metric, to see how *No Problem Found (NPF)* fixes compare to failures that are fixed by hard disk replacement. As can be seen from the figure, *No Problem Found* failures take the longest time to fix: there was a failure, however the technician is not able to identify the failure correctly, and thus failed to return it to healthy state. In most other

cases, the average time to fix the failure is significantly shorter than *No Problem Found* case, because it is easy to identify the component where the failure occurs. NIC and Backplane are exceptions, because those repairs involve taking out the entire enclosure.

6.1.3 Recurrent Soft Failure Probability

The probability of a machine having another soft failure, after encountering a specified number of soft failures is plotted in Figure 6.3. As can be observed from the figure, there is no significant increase in the probability of another failure, until the third occurrence following the first occurrence of a soft failure, however, after that, there is a sharp increase in probability of soft failures. In fact after the fifth consecutive soft failure, one out of every two machines encounters another soft failure. This is a very useful statistic to have, since we can formulate machine repair strategies using this probability.

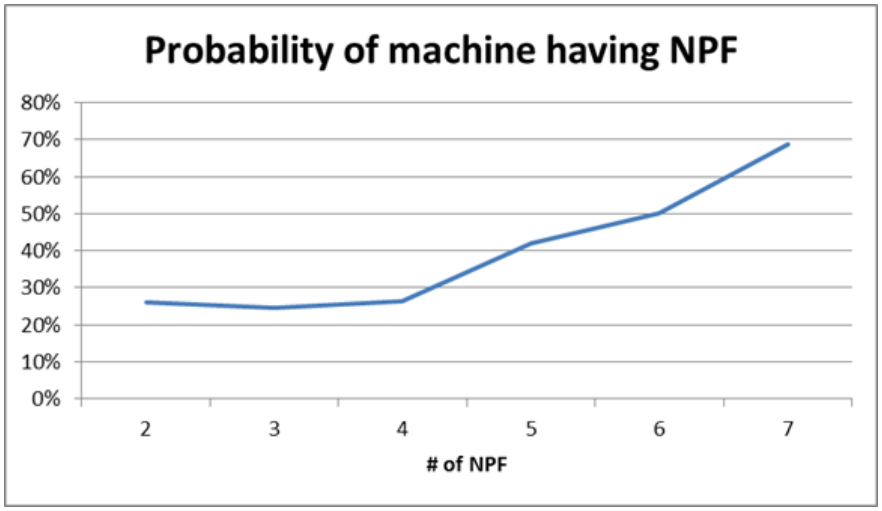


Figure 6.3: Probability of machines to have recurrent No Problem Found failures

6.1.4 Next Fix After a Soft Failure

In this section, we identify the likelihood of the next fix after a soft failure fix categorized as *No Problem Found*. Essentially, we want to identify whether a subsequent fix on the same machine yielded a different fix, or we remained unable to diagnose the issue, and classified it as a soft failure. Figure 6.4 shows the immediate next fix following a *No Problem Found* fix. The most immediate fix after an initial *No Problem Found* ends up being another *No Problem Found*, which is the most likely cause of the recurrent behavior we saw in Figure 6.3.

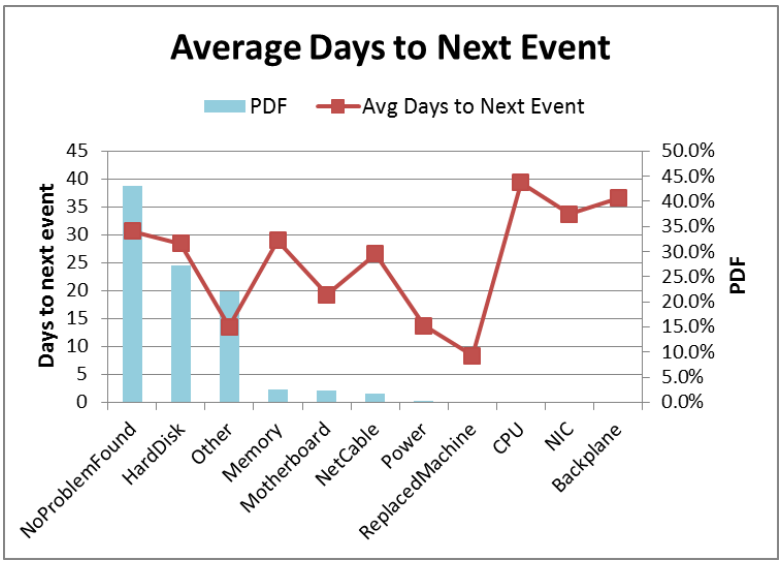


Figure 6.4: Next fix and Average days to next fix, following an NPF event

In addition to plotting the next fix after a *No Problem Found* event on the left vertical axis, we plot the average days to the next event on the right vertical axis. This is a measure of the efficiency of the fix: if the fix were made correctly, then it would take a lot longer for a subsequent failure to happen. Soft failures seem to happen at around the same frequency as hard disk drives, and do not show any particularly interesting inter-arrival behavior.

6.1.5 Probability of Soft Failures Leading to an Actual Failure

It is natural to investigate the correlation between the number of soft failures in a machine and actual failures experienced by the machines. In order to evaluate this question, we identify the count of *No Problem Found* failures before an actual hardware fault in a machine. We see in Figure 6.5, that the probability of the first event being a soft failure is much higher than the probability of having an actual failure. In all machines that had only two failures, the first failure was a soft failure 51% of the time, whereas the first failure was an actual failure only 28% of the time.

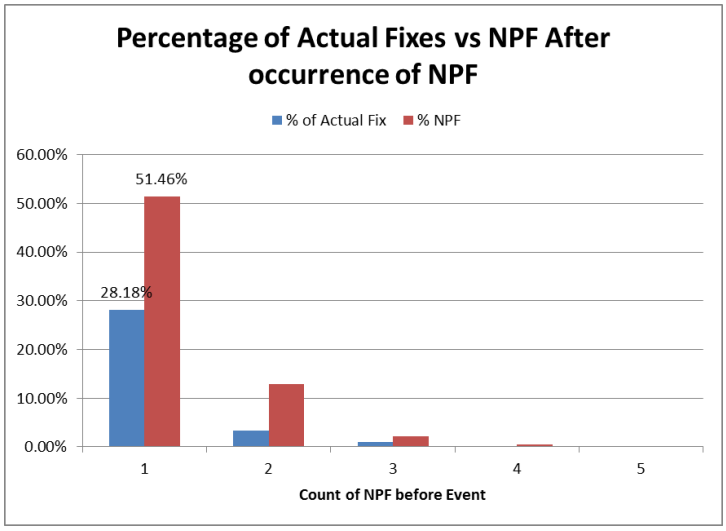


Figure 6.5: Subsequent fix type after a Soft Failure

6.2 Possible Architectural Approaches to Counter Soft Failures

As we observed from the data presented in the earlier section, Soft failures in a data center are very hard to root-cause and even harder to fix. The typical fix-only approach to this problem is an iterative approach, where technicians fix a potential failing part, and keep fixing other parts till the issue "goes away". We argue that a system-wide approach to solving this issue is needed.

In this work, we present three primary approaches to solve this issue in a large data center, 1) Process modifications based on our characterization 2) Architectural modifications to hardware components in the data center and 3) Software approaches that can identify and mask the fault. Because this is an open problem, we also discuss several possible avenues of research at the end of this section.

6.2.1 Process Modifications

From our earlier section, we observed some important characteristics of 'Soft failures'. To summarize, soft failures were likely to occur in one out of every two machines that had five consecutive failures (Figure 6.3), and these failures occur at inter-arrival rates similar to those of hard disk drives (Figure 6.4). An easy process modification is to remove the machine with five consecutive soft failures from the data center. Note that this replacement would be done specific to each environment based on cost models. Since we are dealing with a very high rate of consecutive failures, this process modification can be more efficient than leaving the machine in production. Another process modification is to increase technician training for prioritizing soft failure fixes, since these constitute the most frequent type of issues seen in the data center.

6.2.2 Hardware Modifications to Handle Soft Failures

We often find that soft failures are caused by incorrect racking and incorrect cabling of components. It is hard to identify the component that could have failed; if no components actually failed. The failures that get tagged as *No Problem Found* can be partly addressed by removing cabling from inside the server chassis. A chassis could be designed as a collection of servers inside an enclosure with a shared backplane, and plug in place hard disk drives, NIC and PCIe cards. This would help eliminate the need for several cables that are present in a server enclosure,

including SATA cables that connect hard disk drives, network cables that connect from a switch to the server, etc. Even in this case, there could be connector end-point issues that might cause *No Problem Found* errors. The growth of System-on-chip designs and introduction of non-volatile memory designs [72] in this paradigm might remove the need for external connectors in the future.

Another hardware modification that could be employed to handle soft failures at scale is to identify opportunities for provisioning components in alternate topologies. For instance, data centers currently deploy centralized UPS systems as transitory power source, while switching over to a generator backup. New topologies could deploy distributed smaller sized UPS systems [32], in order to help minimize the impact of transient failures of large components. This architectural change might increase maintenance costs, and also service costs. However it might increase the possibility of fault tolerance to both regular hardware failures and soft failures.

6.2.3 Software Approaches for Soft failures

Soft failures are like any other failure in a data center, and software and system redundancy approaches that are employed in cloud services can mask the fault from being visible at the service level. There are sub-system level error detection capabilities in processors, memory controllers and SMART counters in hard disk drives that can detect potential failure modes. However, there are very few diagnostic tests targeted to detect potential soft failures at the system level. Hence it is imperative to build a decision tree, and store the history of server behavior and repair fixes, so we can identify possible server behavior that can denote a soft fault. For instance, if a server does not fail any of the diagnostic tests, but fails to boot or image, then there might be a possible soft failure. Also, if the same server is encountering repeated transient errors, but passes all diagnostic tests, then it could be a potential candidate that is encountering soft failures. Such behavioral

identifications could be made by a management software layer similar to Microsoft's Autopilot [53] or Google's System Health Infrastructure [71].

6.3 SUMMARY

This chapter provides a characterization of a new class of failures in large data centers that we term as Soft failures. We show that Soft failures are very common in large data centers, and we also show that they are recurrent in nature. We show that this class of failures takes the longest time to fix and we also present process, hardware and software approaches to address this failure space.

This chapter covers work published in CAL 2013 [81].

7 CONCLUSIONS AND FUTURE WORK

Data center critical infrastructure is one of the often-overlooked research areas that has recently become important for managing and optimizing cloud margins. This dissertation looks at the impact of data center critical infrastructure (cooling, power and management) on server availability with real world data center studies, and creates the foundation for understanding the multiple inter-dependent infrastructures inside the data center. Chapter 3 of this dissertation looked at the impact of temperature on hard disk failures, focusing on the cooling sub-system of the data center. Previous studies have been inconclusive on the impact of data center temperature on hard disk drive failures. Our work establishes a strong correlation between average temperature at different location granularities in the data center and the failures observed at those locations. We also show that variations in temperature or workload behavior does not exhibit a strong correlation to hard disk failures. We also present a cost analysis between data center level optimizations and server fan speeds, using real cost data from data centers. The relationship that we established and the models we presented can be used by data center designers everywhere to create an environment with reduced failure cost, and increased overall TCO efficiency including cooling optimizations within the data center. Recently, large companies like Microsoft and Facebook are using air-side economizers, which use outside air for cooling. Compared to existing data center cooling solutions, the environments in these data centers are expected to vary a lot, and include more uncontrolled changes, including particulate and gaseous contaminants. Understanding the impact of the emerging cooling systems in data centers on server availability is an ongoing area of work that followed our research on temperature impact on hard disk drives. An additional dimension is to evaluate humidity and vibration impacts in these new data center architectures.

In Chapter 4, we explored power infrastructure designs in large data centers, focusing on the second component of data center critical infrastructures. We explore commonly used availability metrics in data centers and show that these metrics are inadequate to design cloud data centers. Previous work on provisioning focused heavily on power capacity, whereas in our work we illustrate the availability provisioning paradigm, where data center designers over-provision redundant equipment as well. In order to evaluate real power availability events, we characterize events from two large data centers capturing both duration and inter-arrival times. We then present a novel provisioning method, power availability provisioning, which leverages the tradeoff between power capacity utilization, power redundancy and data center performance SLAs. A workload-driven approach with software stack investments in resiliency that can be composed from less reliable hardware infrastructures [29] is a topic of interest in the cloud infrastructure domain, especially in solutions like Windows Azure or Amazon AWS. Our work presents power availability provisioning, with a view of providing the foundational elements of the infrastructure design. We expect that future work in this area builds on top of this work to include software and platform resiliency for cloud services in the presence of power faults.

Chapter 5 presents the case for wireless in data center management, and explores this technology in real data centers. We show through a cost analysis that wireless data center management has an order of magnitude lower cost potential than existing state-of-the-art wired solutions. We select power capping as an example scenario for a time-critical and high-reliability management function. We then present our design, CapNet, for power capping in data centers. Previous work in data centers using wireless has been primarily used for passive monitoring, and our work is one of the first to present an actual implementation for active control. Following our prototyping work with ZigBee, we plan to explore alternate radio opportunities for communication

inside a data center. White spaces represent the unoccupied TV spectrum, and provide tremendous opportunities for unlicensed users of wireless communication with substantial bandwidth and long transmission ranges [3][99]. Figure 7.1 shows the spectrum measured between 512MHz and 698MHz inside a real data center of dimension 50m x 50m. Considering the signal strengths below -114dBm specified by the FCC, the figure shows that more than 50% of the spectrum are whitespaces and are available across the data center, providing us with excellent availability for using White Spaces inside data centers. Hence, we are continuing our work in developing wireless networks for management traffic in data centers, by researching the applicability of Sensor Networking over White Spaces (SNOW) in data centers as future work in this area.

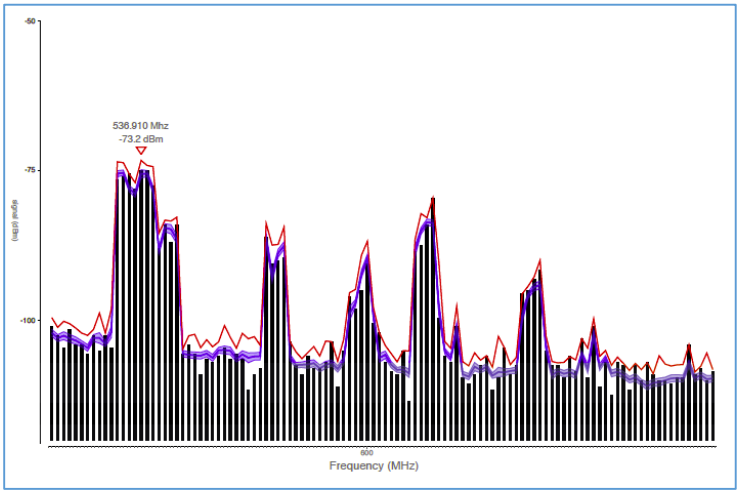


Figure 7.1: Spectrum (512-698 MHz) inside a data center

Our research on data center critical infrastructures exposed a new trend of data center failures, called Soft failures. In Chapter 6, we characterize soft failure occurrence, and event patterns associated with this new class of failures. Soft failures occur in data centers at a high frequency, but are typically ignored by previous work, which assume that events resulting in hardware replacements alone constitute actual data center failures. We expose in this chapter that this new

class of failures occurs at regular frequencies and are also recurrent in cases where they are not fixed accurately. We propose multiple early approaches to address this failure class, posing this as an open research challenge to be addressed in future work. We believe that our first work in this area would provide researchers the data and patterns needed to understand this problem space better.

In addition to future work in multiple infrastructure areas, including cooling, power and management infrastructures, we also highlight another body of work that has been motivated by this dissertation. Large companies invest significant amount of capital expenditure in building data centers, and continue to scale up these facilities. However, there is a lack of holistic simulators that capture the interaction between multiple infrastructures in addition to cost tradeoffs incurred by large enterprises. We are currently exploring the simulation framework needed to evaluate design options and simulate runtime tradeoffs in the data center [83].

This dissertation focuses on contributions in the domain of cloud-scale data centers, however, our conclusions could be applied in several domains which house a number of computing elements together as a large system, by following our methodology of characterizing availability and infrastructure tradeoffs in those domains. For instance, High Performance Computing (HPC) systems do not have the same cost tradeoffs as cloud data centers, however, they also experience reliability concerns and performance variations that impact the operation of an HPC installation [72]. Our discussions on hard disk drive reliability models and application to system design are relevant even in that domain to reduce reliability issues. However, we should characterize the tradeoffs for HPC applications more carefully, since they are not as tolerant to performance deviations when compared to cloud workloads. Financial companies and enterprises have different requirements for data centers, but they also incur additional costs for constructing and building

new data centers. Provisioning methodologies like power availability provisioning can delay the need to build another data center and can motivate sustainable development of smaller data centers. Radio communication technologies for high density servers like wireless data center management is applicable beyond the data center domain as well. Our contribution is one of the first proposals to implement active control with dense cluster of wireless nodes, which has potential for being deployed in densely populated manufacturing and industrial applications. An approach that utilizes characterization methodologies in new domains can leverage the contributions from this dissertation and further the research in increasing infrastructure efficiency.

BIBLIOGRAPHY

[1] Al-Fares, M., Loukissas, A., & Vahdat, A. (2008, August). A scalable, commodity data center network architecture. In *ACM SIGCOMM Computer Communication Review* (Vol. 38, No. 4, pp. 63-74). ACM.

[2] Backes, W., & Cordasco, J. (2010). MoteAODV–an AODV implementation for TinyOS 2.0. In *Information Security Theory and Practices. Security and Privacy of Pervasive Systems and Smart Devices* (pp. 154-169). Springer Berlin Heidelberg.

[3] Bahl, P., Chandra, R., Moscibroda, T., Murty, R., & Welsh, M. (2009). White space networking with wi-fi like connectivity. *ACM SIGCOMM Computer Communication Review*, *39*(4), 27-38.

[4] Bell, D., Osborne, L., and McGary, J. (2002). Remote Systems Management Using the Dell Remote Access Card.

[5] Bhattacharya, A. A., Culler, D., Kansal, A., Govindan, S., & Sankar, S. (2013). The need for speed and stability in data center power capping. *Sustainable Computing: Informatics and Systems*, *3*(3), 183-193.

[6] Casadei, D., Grandi, G., & Rossi, C. (2002). A supercapacitor-based power conditioning system for power quality improvement and uninterruptible power supply. In *Industrial Electronics, 2002. ISIE 2002. Proceedings of the 2002 IEEE International Symposium on* (Vol. 4, pp. 1247-1252). IEEE.

[7] CDW-G RJ45 patch cable

[8] CDW-G Fixed Management Switches

[9] CDW-G Digi Passport 48 console server

[10] CDW-G x-86 based Server Management

[11] Cepin, M. (2011). Assessment of power system reliability. *Methods and Applications*.

[12] Chen, G., He, W., Liu, J., Nath, S., Rigas, L., Xiao, L., & Zhao, F. (2008, April). Energy-Aware Server Provisioning and Load Dispatching for Connection-Intensive Internet Services. In *NSDI* (Vol. 8, pp. 337-350).

[13] Choi, J., Govindan, S., Urgaonkar, B., & Sivasubramaniam, A. (2008, September). Profiling, prediction, and capping of power consumption in consolidated environments. In *Modeling, Analysis and Simulation of Computers and Telecommunication Systems, 2008. MASCOTS 2008. IEEE International Symposium on* (pp. 1-10). IEEE.

[14] Cole, G. (2000). Estimating drive reliability in desktop computers and consumer electronics systems. *Seagate Technology Paper TP*, *338*.

[15] DCMI Specification http://www.intel.com/technology/product/DCMI/index.htm

[16] Dean, J., & Barroso, L. A. (2013). The tail at scale. *Communications of the ACM*, *56*(2), 74-80.

[17] Delimitrou, C., Sankar, S., Kansal, A., & Kozyrakis, C. (2012, November). ECHO: Recreating network traffic maps for data centers with tens of thousands of servers. In *Workload Characterization (IISWC), 2012 IEEE International Symposium on* (pp. 14-24). IEEE.

[18] Delimitrou, C., Sankar, S., Vaid, K., & Kozyrakis, C. (2011, November). Decoupling data center studies from access to large-scale applications: A modeling approach for storage workloads. In *Workload Characterization (IISWC), 2011 IEEE International Symposium on* (pp. 51-60). IEEE.

[19] Digi International 70001908 32-Port Console Port Management Device (www.amazon.com)

[20] Elerath, J. G., & Shah, S. (2004, January). Server class disk drives: how reliable are they?. In *Reliability and Maintainability, 2004 Annual Symposium-RAMS* (pp. 151-156). IEEE.

[21] El-Sayed, N., Stefanovici, I. A., Amvrosiadis, G., Hwang, A. A., & Schroeder, B. (2012). Temperature management in data centers: why some (might) like it hot. *ACM SIGMETRICS Performance Evaluation Review*, *40*(1), 163-174.

[22] Facebook 2011. Open compute project at Facebook http://opencompute.org/

[23] Fan, X., Weber, W. D., & Barroso, L. A. (2007, June). Power provisioning for a warehouse-sized computer. In *ACM SIGARCH Computer Architecture News* (Vol. 35, No. 2, pp. 13-23). ACM.

[24] Felter, W., Rajamani, K., Keller, T., & Rusu, C. (2005, June). A performance-conserving approach for reducing peak power consumption in server systems. In *Proceedings of the 19th annual international conference on Supercomputing* (pp. 293-302). ACM.

[25] Femal, M. E., & Freeh, V. W. (2005). Safe overprovisioning: Using power limits to increase aggregate throughput. In *Power-Aware Computer Systems* (pp. 150-164). Springer Berlin Heidelberg.

[26] Fortenbery, B. (2007). Power Quality in Internet Data Centers. PQ TechWatch.

[27] Fu, X., Wang, X., & Lefurgy, C. (2011, June). How much power oversubscription is safe and allowed in data centers. In *Proceedings of the 8th ACM international conference on Autonomic computing* (pp. 21-30). ACM.

[28] Gantz, J., & Reinsel, D. (2010). The digital universe decade-are you ready. *IDC iView*.

[29] Gauthier, D. (2011). Software Reigns in Microsoft's Cloud-Scale Data Centers. Global Foundation Services, Microsoft.

[30] Gill, P., Jain, N., & Nagappan, N. (2011, August). Understanding network failures in data centers: measurement, analysis, and implications. In *ACM SIGCOMM Computer Communication Review* (Vol. 41, No. 4, pp. 350-361). ACM.

[31] Govindan, S., Lefurgy, C., & Dholakia, A. (2009, June). Using on-line power modeling for server power capping. In *Proceedings of the 2009 Workshop on Energy-Efficient Design (WEED)*.

[32] Govindan, S., Wang, D., Chen, L., Sivasubramaniam, A., & Urgaonkar, B. (2011, October). Towards realizing a low cost and highly available data center power infrastructure. In Proceedings of the 4th Workshop on Power-Aware Computing and Systems (p. 7). ACM

[33] Govindan, S., Choi, J., Urgaonkar, B., Sivasubramaniam, A., & Baldini, A. (2009, April). Statistical profiling-based techniques for effective power provisioning in data centers. In *Proceedings of the 4th ACM European conference on Computer systems* (pp. 317-330). ACM.

[34] Gray, J., & Van Ingen, C. (2007). Empirical measurements of disk failure rates and error rates. *arXiv preprint cs/0701166*.

[35] Greenberg, S., Mills, E., Tschudi, B., Rumsey, P., & Myatt, B. (2006). Best practices for data centers: Lessons learned from benchmarking 22 data centers. *Proceedings of the ACEEE Summer Study on Energy Efficiency in Buildings in Asilomar, CA. ACEEE, August*, *3*, 76-87.

[36] Guo, G., & Zhang, J. (2003). Feedforward control for reducing disk-flutter-induced track misregistration. *Magnetics, IEEE Transactions on*, *39*(4), 2103-2108.

[37] Gurumurthi, S., Zhang, J., Sivasubramaniam, A., Kandemir, M., Franke, H., Vijaykrishnan, N., & Irwin, M. J. (2003, March). Interplay of energy and performance for disk arrays running transaction processing workloads. In *Performance Analysis of Systems and Software, 2003. ISPASS. 2003 IEEE International Symposium on* (pp. 123-132). IEEE.

[38] Gurumurthi, S., Sivasubramaniam, A., & Natarajan, V. K. (2005). *Disk drive roadmap from the thermal perspective: A case for dynamic thermal management* (Vol. 33, No. 2, pp. 38-49). ACM.

[39] Hamilton, J. (2011). 42: The Answer to the Ultimate Question of Life, the Universe, and Everything.

[40] Hamilton, J. (2008). Cost of power in large-scale data centers. *Blog entry dated*, *11*, 28.

[41] Hölzle, U., & Barroso, L. A. (2009). The data center as a computer: An introduction to the design of warehouse-scale machines. *Synthesis lectures on computer architecture*, *4*(1), 1-108.

[42] HP power capping and HP dynamic power capping for ProLiant servers. Technology brief, 2nd edition. http://h20000.www2.hp.com/bc/docs/support/SupportManual/c01549455/c01549455.pdf

[43] HP iLO Management Engine. http://h18013.www1.hp.com/products/servers/management/remotemgmt. html

[44] HP White Paper. (2003). Assessing and Comparing Serial Attached SCSI and Serial ATA Hard Disk Drives and SAS interface.

[45] HP SSA70 Storage Disk Enclosure. (2011). h18006.www1.hp.com/storage/disk_storage/index.html

[46] IBM Remote Supervisor Adapter II http://www.ibm.com/support/docview.wss?uid=psg1MIGR-50116

[47] IEEE Std 493-1997: IEEE Recommended Practice for the Design of Reliable Industrial and Commercial Power Systems

[48] Igure, V. M., Laughter, S. A., & Williams, R. D. (2006). Security issues in SCADA networks. *Computers & Security*, *25*(7), 498-506.

[49] Intel Whitepaper. (2008). "Reducing Data Center Cost with an Air Economizer". August 2008.

[50] Intelligent power optimization for higher server density racks a baidu case study with intel intelligent power technology. http://www.intel.com/content/dam/doc/case-study/data-center-efficiency-xeon-baidu-case-study.pdf

[51] IOMeter 2011. IOMeter project – www.iometer.org

[52] IPMI v2.0 Specifications http://www.intel.com/design/servers/ipmi/

[53] Isard, M., Budiu, M., Yu, Y., Birrell, A., & Fetterly, D. (2007). Dryad: distributed data-parallel programs from sequential building blocks. *ACM SIGOPS Operating Systems Review*, *41*(3), 59-72.

[54] ITI (CBEMA) Curve Application Note, Technical Committee 3, Information Technology Industry Council

[55] Jackson, D. B. Intentional Electromagnetic Interference–Leapfrogging Modern Data Center Physical and Cyber Defenses.

[56] Kim, Y., Gurumurthi, S., & Sivasubramaniam, A. (2006, February). Understanding the performance-temperature interactions in disk I/O of server workloads. In *High-performance computer architecture, 2006. The twelfth international symposium on* (pp. 176-186). IEEE.

[57] Kontorinis, V., Zhang, L. E., Aksanli, B., Sampson, J., Homayoun, H., Pettis, E., Tullsen, D. T., and Rosing, T. S. 2012. Managing distributed ups energy for effective power capping in data centers. In Proceedings of the 39th Annual International Symposium on Computer Architecture (ISCA '12). IEEE Computer Society, Washington, DC, USA, 488-499.

[58] Koomey, J. (2011). Growth in data center electricity use 2005 to 2010. *The New York Times*, *49*(3).

[59] Kozyrakis, C., Kansal, A., Sankar, S., & Vaid, K. (2010). Server engineering insights for large-scale online services. *IEEE micro*, *30*(4), 8-19.

[60] Liang, C. J. M., Liu, J., Luo, L., Terzis, A., & Zhao, F. (2009, November). Racnet: a high-fidelity data center sensing network. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems* (pp. 15-28). ACM.

[61] Lim, H., Kansal, A., & Liu, J. (2011, June). Power budgeting for virtualized data centers. In *2011 USENIX Annual Technical Conference (USENIX ATC'11)* (p. 59).

[62] MegaRAC Remote Management AMI. http://www.ami.com/serviceprocessors/

[63] Microchip Site - http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en535967

[64] Miller, R. (2009). Microsoft's chiller-less data center. *Data center knowledge*.

[65] Miller, R. (2009). Inside Microsoft's Chicago Data Center. *Data center knowledge.*

[66] Mukherjee, S. (2011). *Architecture design for soft errors*. Morgan Kaufmann.

[67] Namek, R.Y., Fournier, E.. (2011). Two strategies to reduce Chiller power and Plant energy consumption in Data centers. *Data centerDynamics*.

[68] Patterson, M. K. (2008, May). The effect of data center temperature on energy efficiency. In *Thermal and Thermomechanical Phenomena in Electronic Systems, 2008. ITHERM 2008. 11th Intersociety Conference on* (pp. 1167-1174). IEEE.

[69] Park, R., & Buch, I. (2007). Improve Debugging and Performance Tuning with ETW. *MSDN Magazine,[Online],[Accessed: 01.01. 2012], Avaliable from: http://msdn. microsoft. com/en-us/magazine/cc163437. aspx*.

[70] Pelley, S., Meisner, D., Zandevakili, P., Wenisch, T. F., & Underwood, J. (2010, March). Power routing: dynamic power provisioning in the data center. In *ACM Sigplan Notices* (Vol. 45, No. 3, pp. 231-242). ACM.

[71] Pinheiro, E., Weber, W. D., & Barroso, L. A. (2007, February). Failure Trends in a Large Disk Drive Population. In *FAST* (Vol. 7, pp. 17-23).

[72] Raju, N., Gottumukkala, Y. L., Leangsuksun, C. B., Nassar, R., & Scott, S. (2006). Reliability Analysis in HPC clusters. In *Proceedings of the High Availability and Performance Computing Workshop*.

[73] Ranganathan, P., and Chang, J. (2012). (Re) Designing Data-Centric Data Centers." *Micro, IEEE 32.1 (2012):* 66-70.

[74] Ranganathan, P., Leech, P., Irwin, D., & Chase, J. (2006, June). Ensemble-level power management for dense blade servers. In *ACM SIGARCH Computer Architecture News* (Vol. 34, No. 2, pp. 66-77). IEEE Computer Society.

[75] Raghavendra, R., Ranganathan, P., Talwar, V., Wang, Z., & Zhu, X. (2008, March). No power struggles: Coordinated multi-level power management for the data center. In *ACM SIGARCH Computer Architecture News* (Vol. 36, No. 1, pp. 48-59). ACM.

[76] Rockwell Automation Inc Bulletin 1489 circuit breakers selection guide, 2010

[77] Saifullah, A., Xu, Y., Lu, C., & Chen, Y. (2013). Distributed channel allocation protocols for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*.

[78] Sankar, S., Gauthier, D., & Gurumurthi, S. (2014). Power Availability Provisioning in Large Data Centers. *Computing Frontiers 2014*

[79] Sankar, S., Shaw, M., & Vaid, K. (2011, June). Impact of temperature on hard disk drive reliability in large data centers. In *Dependable Systems & Networks (DSN), 2011 IEEE/IFIP 41st International Conference on* (pp. 530-537). IEEE..

[80] Sankar, S., Shaw, M., Vaid, K., & Gurumurthi, S. (2013). Data center Scale Evaluation of the Impact of Temperature on Hard Disk Drive Failures. *Trans. Storage 9*, 2, Article 6

[81] Sankar, S., and Gurumurthi, S. (2013). Soft Failures in Large Data centers. In *IEEE Computer Architecture Letters*, vol. 99, no. RapidPosts, p. 1, , 2013

[82] Sankar, S., Gurumurthi, S., & Stan, M. R. (2008, June). Intra-disk parallelism: an idea whose time has come. In *ACM SIGARCH Computer Architecture News* (Vol. 36, No. 3, pp. 303-314). IEEE Computer Society.

[83] Sankar, S., Kansal, A., & Liu, J. Towards a Holistic Data Center Simulator. *MODSIM 2013*

[84] Sankar, S., & Vaid, K. (2009, October). Storage characterization for unstructured data in online services applications. In *Workload Characterization, 2009. IISWC 2009. IEEE International Symposium on* (pp. 148-157). IEEE.

[85] Schroeder, B., & Gibson, G. A. (2010). A large-scale study of failures in high-performance computing systems. *Dependable and Secure Computing, IEEE Transactions on, 7*(4), 337-350.

[86] Schroeder, B., & Gibson, G. A. (2007, February). Disk failures in the real world: What does an mttf of 1, 000, 000 hours mean to you?. In *FAST* (Vol. 7, pp. 1-16).

[87] Schroeder, B., Pinheiro, E., & Weber, W. D. (2009, June). DRAM errors in the wild: a large-scale field study. In *ACM SIGMETRICS Performance Evaluation Review* (Vol. 37, No. 1, pp. 193-204). ACM.

[88] Schwarz, T., Baker, M., Bassi, S., Baumgart, B., Flagg, W., van Ingen, C., ... & Shah, M. (2006, May). Disk failure investigations at the internet archive. In *Work-in-Progess session, NASA/IEEE Conference on Mass Storage Systems and Technologies (MSST2006)*..

[89] Seymour, J. (2005), The Seven Types of Power Problems, American Power Conversion, *Schneider Electric White Paper 18*

[90] Srinivasan, K. and Philip Levis. (2006). RSSI is Under Appreciated. In *Proceedings of the Third Workshop on Embedded Networked Sensors (EmNets 2006)*.

[91] TinyOS Community Forum. http://www.tinyos.net/

[92] Turner IV, W. P., PE, J., Seader, P. E., & Brill, K. J. (2006). Tier Classification Define Site Infrastructure Performance. *Uptime Institute*, 17.

[93] Vishwanath, K. V., & Nagappan, N. (2010, June). Characterizing cloud computing hardware reliability. In *Proceedings of the 1st ACM symposium on Cloud computing* (pp. 193-204). ACM.

[94] Wang, D., Ren, C., Govindan, S., Sivasubramaniam, A., Urgaonkar, B., Kansal, A., and Vaid, K.. 2013. ACE: abstracting, characterizing and exploiting peaks and valleys in data center power consumption. *SIGMETRICS Perform. Eval. Rev. 41*, 1 (June 2013), 333-334

[95] Wang, D., Govindan, S., Sivasubramaniam, A., Kansal, A., Liu, J., & Khessib, B. (2013). *Underprovisioning Backup Power Infrastructure for Data centers*. Technical Report CSE-13-012, The Pennsylvania State University.

[96] Wang, X., Chen, M., Lefurgy, C., & Keller, T. W. (2012). Ship: A scalable hierarchical power control architecture for large-scale data centers. *Parallel and Distributed Systems, IEEE Transactions on, 23*(1), 168-176.

[97] Wang, Z., Zhu, X., McCarthy, C., Ranganathan, P., & Talwar, V. (2008, June). Feedback control algorithms for power management of servers. In *Third International Workshop on Feedback Control Implementation and Design in Computing Systems and Networks (FeBid), Annapolis, MD*.

[98] Yang, J., and Sun, F. 1999. A comprehensive review of hard-disk drive reliability. In Proceedings of the Annual Symposium on Reliability and Maintainability, pages 403 - 409, January 1999Wang, D., Ren, C., Govindan, S., Sivasubramaniam, A., Urgaonkar, B., Kansal, A., and Vaid, K.. 2013. ACE: abstracting, characterizing and exploiting peaks and valleys in data center power consumption. SIGMETRICS Perform. Eval. Rev. 41, 1 (June 2013), 333-334

[99] Ying, X., Zhang, J., Yan, L., Zhang, G., Chen, M., & Chandra, R. (2013, September). Exploring indoor white spaces in metropolises. In *Proceedings of the 19th annual international conference on Mobile computing & networking* (pp. 255-266). ACM.

[100] Zhang, Y., Wang, Y., & Wang, X. (2011, June). Capping the electricity cost of cloud-scale data centers with impacts on power markets. In *Proceedings of the 20th international symposium on High performance distributed computing* (pp. 271-272). ACM.

[101] Zhou, X., Zhang, Z., Zhu, Y., Li, Y., Kumar, S., Vahdat, A., Zhao, B. Y., & Zheng, H. (2012). Mirror mirror on the ceiling: flexible wireless links for data centers. *ACM SIGCOMM Computer Communication Review*, *42*(4), 443-454.