

# Using Batteries to Reduce the Power Costs of Internet-scale Distributed Networks

Darshan S. Palasamudram<sup>†</sup>, Ramesh K. Sitaraman<sup>‡</sup>, Bhuvan Urgaonkar<sup>†</sup>, Rahul Urgaonkar<sup>\*</sup>  
Penn. State Univ.<sup>†</sup>, UMass Amherst<sup>‡</sup>, Akamai Technologies<sup>‡</sup>, Raytheon BBN Technologies<sup>\*</sup>  
darshan.ps@cse.psu.edu<sup>†</sup>, ramesh@cs.umass.edu<sup>‡</sup>, bhuvan@cse.psu.edu<sup>†</sup>, rahul@bbn.com<sup>\*</sup>

## ABSTRACT

Modern Internet-scale distributed networks have hundreds of thousands of servers deployed in hundreds of locations and networks around the world. Canonical examples of such networks are content delivery networks (called CDNs) that we study in this paper. The operating expenses of large distributed networks are increasingly driven by the cost of supplying power to their servers. Typically, CDNs procure power through long-term contracts from co-location providers and pay on the basis of the power (KWs) provisioned for them, rather than on the basis of the energy (KWHs) actually consumed. We propose the use of batteries to reduce both the required power supply and the incurred power cost of a CDN. We provide a theoretical model and an algorithmic framework for provisioning batteries to minimize the total power supply and the total power costs of a CDN. We evaluate our battery provisioning algorithms using extensive load traces derived from Akamai's CDN to empirically study the achievable benefits. We show that batteries can provide up to 14% power savings, that would increase to 22% for more power-proportional next-generation servers, and would increase even more to 35.3% for perfectly power-proportional servers. Likewise, the cost savings, inclusive of the additional battery costs, range from 13.26% to 33.8% as servers become more power-proportional. Further, much of these savings can be achieved with a small cycle rate of one full discharge/charge cycle every three days that is conducive to satisfactory battery lifetimes. In summary, we show that a CDN can utilize batteries to significantly reduce both the total supplied power and the total power costs, thereby establishing batteries as a key element in future distributed network architecture. While we use the canonical example of a CDN, our results also apply to other similar Internet-scale distributed networks.

## Categories and Subject Descriptors

C.4 [Performance of Systems]: Modeling techniques; Design studies

## General Terms

Algorithms, Experimentation, Economics

## Keywords

Internet Content Delivery, Cloud Computing, Energy Efficiency, Energy Storage, Network Architecture

## 1. INTRODUCTION

Internet-scale services that deploy large distributed networks of servers across the globe are fundamentally transforming all aspects of human activity by enabling a wide range of applications and functionality. Canonical examples of such networks are Content Delivery Networks<sup>1</sup> (CDNs, for short) that we study in this work. CDNs deliver Web content, applications,

<sup>1</sup>For convenience, we use the term CDNs to denote delivery networks for not just Web content but also media and applications although the system architecture can be different in ways that are not material to the current work. [20] provides a detailed discussion of these architectural differences.

and streaming media over the Internet with high reliability, performance, and scalability. CDNs are an early example of a Platform-as-a-Service (PaaS) cloud that provides highly-distributed hosting and delivery services for content and application providers. For instance, a large CDN such as Akamai’s consists of over 100,000 servers distributed in over 75 countries and over 1000 networks and serves 15% to 30% of the global Web traffic [1, 14, 20]. Modern CDNs host and deliver Web sites (both static and dynamic content), Web-based and IP-based applications (such as e-commerce portals and SaaS applications), high-definition (HD) streaming media (both live and on-demand), cloud storage services, and even distributed computing where Java application components are run on the CDN’s edge platform [20].

In the quest for greater performance and scalability, CDNs deploy *clusters* of servers in data centers (i.e., colocation facilities) around the world at the “edges” of the Internet so as to be “proximal” in the network sense to users around the world. Each cluster consists of servers deployed in a specific colocation facility at a specific location. A cluster can contain anywhere from tens of servers in a small Tier-3 ISP or a university to thousands of servers in a large colocation facility in a major metro area. A CDN’s servers cooperatively deliver content and applications to optimize the performance experienced by the clients. Content and applications can typically be replicated on demand to any server of the CDN. Each client request is routed by the CDN’s load balancing system to an “optimal” server within a cluster that can serve the content with high availability and performance. In particular, the load balancing system ensures that a user is routed to a server that is live, not overloaded, and is located in a cluster that is proximal to the user in the sense of having a communication path with low latency, low loss, and high throughput.

An important consideration for the CDN is to minimize the cost incurred for content and application delivery. The cost of deploying clusters of servers around the globe includes the capital expenditure (CapEx) of procuring the servers themselves and the expenditure of operating them (OpEx). Pure-play CDNs such as Akamai and Limelight seldom own colocation facilities themselves. Instead, these CDNs rely upon colocation providers (e.g., Switch and Data) to house their servers. Similarly, these CDNs rely upon network providers (e.g., Comcast) for their network connectivity. The OpEx of a CDN typically involves multiple components<sup>2</sup>: a “rent” for the rack space for the CDN’s servers, the power cost for powering the servers, and a bandwidth cost for the traffic served to users by the CDN’s servers. Increasingly, an important component of the OpEx of large distributed systems is the cost of power. While bandwidth costs were the dominant factor in a CDN’s OpEx in the past, bandwidth prices have roughly halved each year during the past decade. For instance, it cost \$0.15 to deliver an MB in 1998 but only costs about \$0.00005 per MB today, a decrease of roughly 1.8X per year. In stark contrast, the cost of power has been rising over the past decade [11] and is now a key factor to reckon with. More importantly, power costs are expected to rise even further during the next decade and are, therefore, likely to account for an even larger share of the OpEx of large distributed networks.

Re-architecting next-generation distributed networks with energy<sup>3</sup> as a first-order principle is an important long-term goal of our work. To that end, in this paper, we formulate and examine novel techniques for reducing the power expenses that a distributed network incurs by using energy storage devices such as batteries.

## 1.1 How Batteries Can Reduce Power Costs

We summarize our key idea that we then explore in detail through the rest of the paper.

Colocation providers typically charge their customers for power using a tariff model called the *supplied power model*. In this model, CDNs pay each colocation facility for the amount of power *supplied* to (i.e., provisioned for) their server racks, not for the energy their servers actually *use*. Figure 1 shows the power demand from the CDN’s servers in a single cluster. The CDN pays for  $P_{supply}$ , irrespective of the peak power used ( $P_{peak}$ ) or the actual energy used (area under the power demand curve). For instance, if a CDN is offered a unit power price  $c_p$  of \$150 per KW per month and wants the colocation provider to provision 6.24 KW of power (208 V  $\times$  30 Amps for about 20 typical servers), then it would pay  $6.24 \times \$150 = \$936$  per month to the colocation provider. The maximum power drawn by the CDN’s servers may not exceed the supplied power at any time, and exceeding that threshold could trip circuit breakers and result in penalties. Therefore, it is customary to incorporate a small “safety margin” by procuring slightly more power supply than the expected peak demand.

Colocation providers have always had centralized batteries to provide an uninterrupted power supply (UPS) in case the power utility fails. Recent technology trends allow incorporating batteries at the rack- or server-level to implement distributed UPS [22], which improves energy conversion efficiency by requiring one less AC-DC conversion than centralized UPS. This

<sup>2</sup>We capture the cost categories that are typical. However, it should be noted some individual variations may exist, e.g., a small tier-3 network may host the CDN’s servers for free if those servers significantly decrease the upstream traffic that this network pays for.

<sup>3</sup>Minimizing energy consumption and energy costs are two complimentary but different objectives. We study costs rather than consumption, while recent work such as [18] are oriented towards minimizing CDN energy consumption.

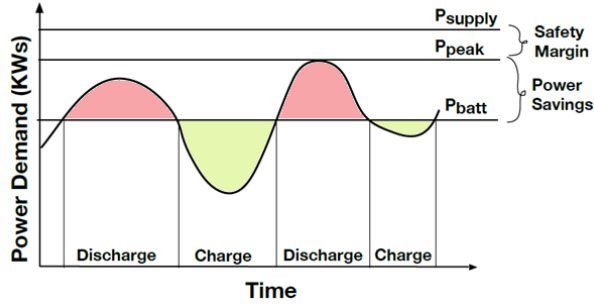


Figure 1: The power demand from servers in a CDN’s cluster. The CDN’s power cost is proportional to the supplied power ( $P_{supply}$ ), not the peak power draw ( $P_{peak}$ ) or the total energy used (area under the power demand curve). Using a battery, the power demand can be satisfied by a power supply of  $P_{batt}$  where the battery is discharged when demand is high and recharged when the demand is low, enabling the CDN to reduce the supplied power by at least  $P_{peak} - P_{batt}$ .

technological development makes it possible for the first time for a CDN to deploy batteries for its own purposes. *The key insight of our work is that the distributed UPS architecture can also be effectively used for reducing the power supply that must be provisioned for each of a CDN’s clusters, resulting in a cost reduction in the supplied power tariff model.* To take the example in Figure 1, without the use of batteries, a CDN would have to provision  $P_{supply}$  to be at least  $P_{peak}$ . However, with a battery of sufficient capacity, it might suffice to provision  $P_{supply}$  to be at least  $P_{batt}$ . Specifically, the battery could discharge and supply power when the power demand exceeds  $P_{batt}$  and it could draw power to charge back again whenever the power demand is smaller than  $P_{batt}$ . Thus, the CDN can decrease its provisioned power by at least  $(P_{peak} - P_{batt})$  by using a battery, resulting in a decrease in the electricity bill of  $c_p(P_{peak} - P_{batt})$ .

Note the key role that the supplied power tariff model plays in the above cost savings, since batteries at best do not change the total energy used, and at worst increase energy usage due to conversion losses. Cloud providers that do not require CDN-like Internet-scale network deployments in hundreds of locations and ISPs around the world have other deployment options that may result in different tariff models for power pricing. For instance, a cloud provider who needs only fewer locations for deployment can own a few big data centers and can procure energy directly from power utilities where the cost could be based on peak and/or average energy usage. Such usage-based power pricing exists but is less common in CDNs today, where pure-play CDNs typically rent (rather than own) space for their servers and procure power supply from the co-location facility rather than directly from the utility.

## 1.2 Our Contributions

We state and solve two key optimization problems in power supply and battery provisioning for a CDN: minimizing the total power supply ( $TP$ ) and minimizing the total cost ( $TC$ ). The total power supply  $TP$  (in KW) of a CDN equals  $\sum_i S_i P_i$ , where  $P_i$  is the supplied power (per server)<sup>4</sup> and  $S_i$  is the number of servers at cluster  $i$ . The total cost  $TC$  (in \$) of the CDN is the sum of its power cost and the amortized cost of procuring the battery.

We theoretically characterize the benefits that batteries provide and how these benefits vary with power demand, battery characteristics, power prices, battery prices, battery lifetimes, and server power proportionality. We also develop algorithms to solve the power supply minimization ( $TPM$ ) and the power cost minimization ( $PCM$ ) problems. Our algorithm and LP-formulations provide a basis for provisioning tools that a CDN can employ to derive how much power and batteries are needed at each cluster to achieve the largest cost reductions. Note that our optimizations are *offline computations*. We envision that in a real deployment, these computations would be performed periodically (say, once a month or once a quarter) on *predicted* power demands at a time-scale where contracts and deployments can be changed. In particular, they are not online computations that attempt to respond to real-time variations in power demand.

We use our theoretical formulation and algorithms on extensive realistic load traces from Akamai’s CDN and model the power contracts prevalent in the CDN industry to make an assessment of the power and cost reduction achievable by using batteries. Our aim is not to provide a detailed system-level implementation of a CDN employing batteries in this fashion. Rather, our primary contribution is to assess the viability of batteries as a key element of next-generation CDN architecture. Here, we make several specific contributions that we list below. By convention, we express battery capacity in the unit of minutes per server where a  $B$ -minute battery can power a server for  $B$  minutes at its peak load.

<sup>4</sup>For notational ease, we express load, power demand, power supplied, and battery parameters on a per-server basis.

- Even a small 5-minute battery (comparable to those in UPS systems today) can provide a 7% power saving, while a large 40-minute battery provides 14%.
- As servers become more power-proportional by decreasing their idle power, batteries provide even more power savings. For a next-generation server whose idle power is only 20% of its peak power, batteries can achieve up to 22% power savings. For a perfectly power-proportional server with zero idle power, batteries can achieve 35.3% savings.
- Most of the power savings are achievable with a small cycle rate of one full discharge/charge cycle every three days. This would allow batteries to last for about 5 years, which is at the higher-end of server refresh durations.
- Heterogenous battery deployments and power-aware global load balancing does not provide sufficient extra power savings to warrant the extra complexity.
- The typical *cost* savings due to batteries is 13.26%, but increases to 21.4% for more power-proportional next-generation servers, and increases even more to 33.8% for perfectly power-proportional servers.

As an aside, note that in the supplied power tariff model, some techniques for energy usage reduction, such as turning off idle servers in a CDN [18], do not necessarily result in a significant reduction in the power cost. Batteries provide CDNs the complimentary option of reducing the power cost, without a significant change in the energy usage. Further, note that while we use the canonical example of a CDN throughout our study, much of our techniques and results also apply to other Internet-scale distributed networks that deploy in colocation facilities, procuring energy using the commonly-used supplied power tariff model.

## 2. BACKGROUND AND DATA SETS

### 2.1 CDN Architecture

A modern CDN is a large distributed system of servers where the servers are deployed in clusters throughout the world. Content providers such as Web portals, SaaS application providers, e-commerce sites, news outlets, media companies, social networks, and movie distribution services use CDNs to host and deliver their content and applications. When a user accesses content or an application hosted by the CDN, the user is assigned to a specific server within a specific cluster by the CDN’s load balancing system. The user then proceeds to download the requested content from the chosen server within the chosen cluster. The user accesses induce *load* on the servers that in turn induces a *power demand* that must be satisfied from the provisioned power supply at each cluster. Note that, unlike in certain other IT domains, the load within a CDN must usually be satisfied immediately and cannot be “deferred,” since these are real-time requests from users. In fact, a primary goal of a CDN is to minimize the response time to the user.

The load balancing system of a CDN has two components: the *global* load balancing component that chooses the optimal cluster for each user, and the *local* load balancing component that chooses the optimal server within the chosen cluster. The load balancing system is optimized for maximizing availability and performance, while also minimizing bandwidth costs. For much of our work, we assume that the load balancing system is unaltered and hence the load and power demand of each cluster are inputs to our problem and cannot be changed by moving load between clusters. However, in Section 6, we study the impact of altering global load balancing itself to make it “power-aware” and we ask how much more savings are possible with this additional capability. Furthermore, we make the simplifying assumption <sup>5</sup> that the CDN’s servers are homogenous and that the local load balancing system distributes the incoming load into a cluster evenly among servers so that each server receives the average cluster load. For a comprehensive treatment of CDN architecture, please refer to [20].

### 2.2 Power

*Power Prices.* While in reality the unit power price varies from one colocation facility to another, the variation within a geographical region (such as a country) is smaller than across regions. For instance, while most major US colocation providers would currently provide a cost in the range \$125 to \$175 per KW per month, the cost in Europe is larger and tends to be in the range \$275 to \$325 per KW per month. Note that the power price is a *blended price* that also implicitly pays for the rack space, cooling power, and other overheads sustained by the colocation provider and so cannot be directly compared with raw per KWH electricity pricing from a power utility.

It is important also to contrast pricing in the supplied power model that is associated with *long-term contracts* between the CDN and the colocation provider that specify both the supplied power and unit power price with the real-time power prices

<sup>5</sup>Neither of these assumptions is strictly true in a real-life CDN but these are reasonable simplifications for our work.

in wholesale electricity markets considered in recent work [21, 23]. The wholesale electricity prices show fine-grained variation across time (e.g., on an hourly-basis) and space (from one location to another) caused by the real-time supply and demand imbalances and transmission inefficiencies. Exposure to such fine-grained variations are, of course, absent in supplied power pricing contracts and is hence not exploitable by a CDN. In our experiments, we use a single blended unit power price for all clusters but evaluate different values for  $c_p$  starting with the value of \$150/KW/month that is typical of current US prices [2] - a “low” value that will likely rise over the next decade. We also consider a “moderate” value of \$300/KW/month that is similar to prices Europe today, and a “high” value of \$500/KW/month to model a situation where there is a sharp increase in power prices over the next decade.

*Power Consumption and Power Proportionality.* We use the standard linear model [10] where the power (in Watts) consumed by a server serving load  $l$  is

$$\text{power}(\lambda) \triangleq P_{idle} + (P_{peak} - P_{idle})l, \quad (1)$$

where the load  $0 \leq l \leq 1$  is the server’s resource utilization <sup>6</sup>,  $P_{idle}$  is the power consumed by an idle server, and  $P_{peak}$  is the power consumed by the server under peak load. Most servers today consume significant amounts of power even when idle. A holy grail for server design is to make them “power proportional” by making  $P_{idle}$  zero. We define a factor that we call the power proportionality factor (PPF) that equals  $\frac{P_{peak} - P_{idle}}{P_{peak}}$ . PPF measures the degree to which a server is power proportional. Several power management techniques have been proposed to make servers power proportional [12]: some minimize energy at the circuit level when idle, some deactivate components not in use (dynamic component deactivation or DCD), others reduce energy consumption by “slowing” server components (dynamic voltage/frequency scaling for CPU/Memory, or DRPM for disks), while still others power off entire servers during non-peak hours, either within a data center [26] or at the level of a complete distributed service spanning several data centers [18]. While we are agnostic to what technique is used, we model three different values for PPF. Our baseline is the current generation of servers engineered for energy efficiency that is used in deployed CDNs with a PPF of 0.6 ( $P_{peak} = 250W$ ,  $P_{idle} = 100W$ ). In anticipation of next-generation servers, we model a PPF of 0.8 ( $P_{peak} = 250W$  and  $P_{idle} = 50W$ ). Finally, we also consider the perfectly power-proportional server with a PPF of 1.0 ( $P_{peak} = 250W$  and  $P_{idle} = 0W$ ). While whether or not the ideal is reachable is debatable, there has been exciting recent research that shows drastic decreases in idle power, e.g., the PowerNap mechanism where a reduction of the effective idle power of a blade server from 270W to 10.4W is reported [19].

Trends in power proportionality are particularly important in CDNs as the typical server utilizations are kept low (generally under 40%, see Figure 3) to maintain high performance and adequate “headroom” for flash crowds, resulting in even smaller power efficiencies (in terms of used power vs. idle power) than what PPF implies.

### 2.3 Batteries

In current colocation facilities, energy storage capability exists in the form of centralized UPS units, typically using lead-acid batteries, that supply power to the facility during a potential outage of the utility. Upon a utility failure, the charge stored in the UPS batteries is used to supply power for the duration it takes for backup sources (e.g., diesel generators) to start up and take over the role of supplying power. Whereas this transition is typically only 15-20 seconds long, for a variety of reasons (both technical and economic) the UPS units end up being significantly over-provisioned and can actually keep the data center powered for up to a few minutes [17]. Recent research has shown how colocation facilities can tap into this energy within UPS batteries for optimizing costs [15, 23], while still providing the reliability guarantees that a traditional way of using the UPS offers [16]. In fact, a colocation facility may find it cost-effective to procure even more energy storage capacity than offered by typically provisioned UPS units [17].

Motivated by this, we explore cost saving opportunities for CDNs by using batteries. To realize such cost savings, the CDN would need to provide its clusters with access to batteries that they can control/operate according to their needs. In the current hosting model, where the CDN “rents” rack space and power from colocation facilities for its clusters, a CDN does not have control over the facilities centralized UPS. However, the advent of distributed UPS [22] makes it feasible for a CDN to provision additional rack-level or server-level batteries for power and cost savings that are entirely under the control of the CDN. While our work does not assume where the batteries are located (rack- versus server-level), we do assume that the battery is owned and operated by the CDN. Furthermore, we assume that even when it does not have per-server batteries (e.g., it may have per-rack UPS), it employs suitable sharing/virtualization techniques that allow each server to operate as if

---

<sup>6</sup>In practice, one may employ a more sophisticated model that considers separately the utilization of various server resources such as CPU, memory, I/O devices, etc., and expresses the server’s power consumption in terms of these. We choose to employ a simpler model which captures well servers where the CPU is the dominant power consuming resource, allowing us to ignore the power consumed by other resources.

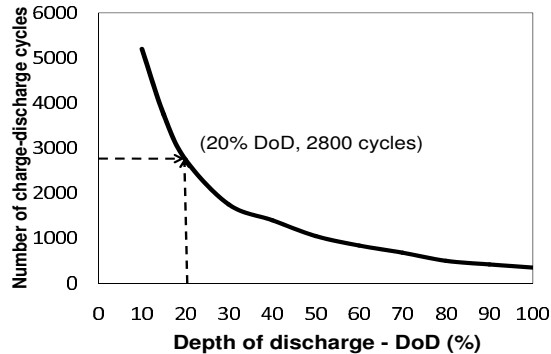


Figure 2: Lifetime chart for a typical lead-acid battery.

it possessed its own dedicated battery. Correspondingly, most of our subsequent discussion will be in terms of a single server employing its battery for savings in supplied power and costs. We model the following key characteristics of batteries that are relevant to our optimizations.

*Battery Capacity  $B$ .* Capacity  $B$  represents the amount of energy a battery can store in KWH/server or Joules/server. We use the term *size* in lieu of capacity as well. Following convention, capacity is also expressed in the units of time by computing the amount of time the battery can sustain a server at peak load, i.e., 1 KWH is equivalent to  $60/P_{peak}$  minutes of capacity (where  $P_{peak}$  is expressed in units of KW).

*Maximum Discharge and Charge Rates.* A battery has limits on the maximum rate  $P_d^{max}$  KW (resp.,  $P_c^{max}$  KW) at which it can discharge (resp., charge). We assume that a battery is able to discharge at a rate sufficient to power a server at maximum load, i.e.,  $P_d^{max} \geq P_{peak}$ . Note that this is also a requirement for UPS batteries, as they need to be able to power the server during a utility outage. We set  $P_d^{max} = P_{peak}$  and allow  $P_c^{max}$  to be smaller. The discharge/charge ratio  $r = \frac{P_d^{max}}{P_c^{max}}$  and is in the range 5-10 for lead-acid batteries.

*Energy Loss Factor  $\alpha$ .* The use of batteries results in energy loss in two key ways: (i) loss due to AC-DC conversion (one conversion for distributed UPS and two conversions for centralized UPS), and (ii) leakage (whereby the battery loses energy merely with the passage of time). Whereas (ii) can be important for certain technologies (e.g., flywheels [25]), it is negligible for lead-acid batteries (over the charge/discharge durations we determine appropriate in Section 4.2) that is most apt for our purposes. Therefore, we ignore (ii) and employ a “loss factor”  $\alpha$  to denote the fractional energy lost in the process of charging. We assume  $\alpha = 0.15$  that is typical for lead-acid batteries.

*Battery Lifetime.* It is crucial that we model the lifetime of a battery as cost must be incurred in replacing it. The lifetime  $L$  of a battery is dependent on complex factors that include environmental factors (e.g., temperature), the technology itself, and the manner in which it is used. A typical lifetime desired from batteries in our CDN would be 5 years (equal to the typical server refresh/upgrade cycle). Additionally, we also consider a more conservative lifetime of 3 years, and a worst-case lifetime of 1 year. A well-known relationship between a battery’s usage and its expected lifetime is given by the “lifetime chart” that presents the number of charge/discharge cycles at a certain depth-of-discharge (DoD) that the battery can undergo (on average) reliably [3]. Consequently, our algorithms limit the *cycle rate* that equals the number of full discharge/charge cycles performed per day, so that a battery may last for its desired lifetime. Figure 2 depicts the lifetime chart for typical lead-acid battery.

*Battery Prices.* The battery costs involve a unit price  $c_b$  (in \$ per KWH) that is technology dependent, e.g.,  $c_b$  is lower for lead-acid batteries than for lithium-ion or ultracapacitors. We consider multiple values for  $c_b$  and lifetime  $L$ , starting with the value of  $c_b = \$100/\text{KWH}$  and a lifetime  $L = 5$  years that is typical for lead acid batteries that are most commonly used and are internally similar to the omnipresent car batteries [5, 25]. We also consider a conservative value of  $c_b = \$300/\text{KWH}$  and a lifetime  $L = 3$  years, and a worst case scenario of  $c_b = \$500/\text{KWH}$  and a lifetime  $L = 1$  year.

*Battery Deployment Model.* Our battery deployment model considers two options: the homogenous model where the same battery capacity  $B$  (per server) is deployed across every cluster of the CDN and the heterogenous model where each cluster  $i$  has a (potentially) different amount of battery capacity  $B_i$  deployed per server. We primarily investigate the homogenous model, but also analyze the extra power and cost savings heterogeneity can provide. It must be noted that homogenous deployment has the advantage of a uniform battery architecture across all clusters, which has potential operational and cost advantages.

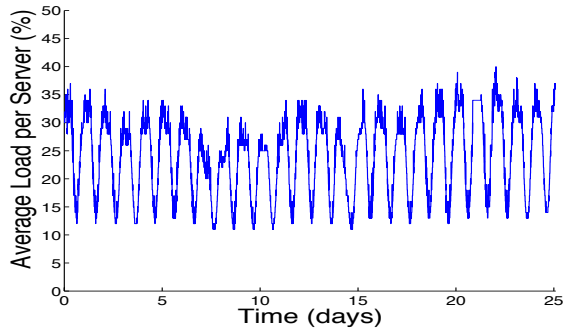


Figure 3: Average normalized load per server measured every 5 minutes from 22 Akamai clusters in the US over 25 days showing day/night variations.

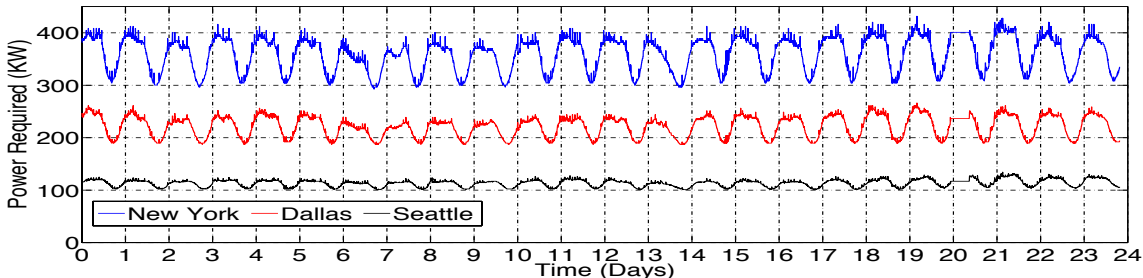


Figure 4: Power demands at a representative large-, medium-, and small-sized cluster.

## 2.4 Akamai Data Sets

For our empirical evaluation, we use a repository of extensive traces from Akamai’s CDN collected from 22 major clusters deployed across the US over a period of 25 days. Our traces are a representative slice of the Web traffic that Akamai’s CDN served in the US during the measurement period. In aggregate, our traces represent data from 15,349 servers that served 950 million requests during the course of our measurements, including a peak traffic of 800,000 requests per second.

We use two types of traces collected every five minutes during the measurement period. The first type of trace has cluster load information that consists of each cluster reporting its average server load every five minutes. The cluster load information is averaged system-wide across all the measured clusters and presented in Figure 3. In addition, we also collect the geographic location of each cluster, including city, state, latitude, and longitude, and the number of servers in that cluster. We could not measure the power demands of each cluster directly. However, we use the power function of Equation (1) with typical values of  $P_{idle}$  and  $P_{peak}$  to map the cluster load sequences to cluster power demand sequences. Thus, we obtain a power demand sequence that provides the average power demand per server for each cluster for each 5-minute time window over our measurement period. Figure 4 shows the resulting power demand sequence at 3 representative clusters with  $P_{idle} = 100W$  and  $P_{peak} = 250W$ . The power demand sequence thus derived serves as the input for all our optimization algorithms. Our methodology of synthetically deriving power demand from actual load traces enables our study of servers with different values for  $P_{idle}$ ,  $P_{peak}$ , and PPF.

The second type of trace is user load information that includes, for each cluster and each block of user IPs, the number of requests made and the total bytes downloaded by users in that IP block from that cluster in the each 5-minute time window. Furthermore, we collected the geographic location of each block of user IPs. Combining the two traces above enables us to derive how much user load originated at each block of IPs and which cluster served them, for each 5-minute time window. The cluster load information is used only in Section 6 where we consider reassigning users to clusters with an “energy-aware” global load balancing algorithm.

## 3. POWER SUPPLY OPTIMIZATION

We formulate two problems and formally derive algorithms and properties that are critical to understanding power supply and cost minimization for CDNs. The *Power Supply Minimization (PSM)* problem minimizes the total power  $TP$  supplied to the clusters of a CDN by the colocation providers, given a set of  $n$  clusters and a power demand sequence  $\langle p_{i,t} \rangle$ , for  $i \in \mathcal{C}$

where  $\mathcal{C}$  is the set of all clusters and  $1 \leq t \leq T$ , where  $P_{idle} \leq p_{i,t} \leq P_{peak}$  is the average power demand of a server (in KW) in cluster  $i$  at time slot  $t$ . The length of each time slot in our data sets is 5 minutes and  $T$  is the length of our trace of roughly 25 days. The total power supply to be minimized is  $TP = \sum_i S_i P_i$ , where  $P_i$  is the power supply per server and  $S_i$  is the number of servers in cluster  $i$ . For each cluster  $i$ ,  $P_i$  must be sufficient to satisfy the power demand sequence  $\langle p_{i,t} \rangle$ , for  $1 \leq t \leq T$ , with the help of a provisioned battery with capacity of  $B_i$  per server with loss rate  $\alpha$ , maximum charge rate  $P_c^{max}$  per server and the maximum discharge rate  $P_d^{max}$  per server. Note that the maximum discharge rate seldom plays a role in our optimization as we assume that  $P_d^{max} = P_{peak}$ .

The *Power Cost Minimization (PCM)* problem minimizes the total *cost TC* of supplying power to the CDN for a month that is inclusive of the cost of procuring the power supply from each colocation provider plus the amortized cost of procuring batteries that are deployed with the servers. As with the PSM problem, we need to ensure that there is sufficient power either from the power supply or the batteries to satisfy the power demand  $\langle p_{i,t} \rangle$  at each server in cluster  $i$ , for  $i \in \mathcal{C}$  where  $\mathcal{C}$  is the set of all clusters, and  $1 \leq t \leq T$ . The total cost to be minimized is  $TC = c_p \sum_i S_i P_i + \frac{c_b}{L} \sum_i B_i S_i$ , where  $P_i$  is the power supply (in KWs) provisioned per server in cluster  $i$ ,  $B_i$  is the battery capacity provisioned per server and  $S_i$  is the number of servers in cluster  $i$ ,  $c_p$  is the unit cost of power (in \$/ KW/month),  $c_b$  is the amortized unit cost of the battery capacity (\$/KWH) and  $L$  is the expected lifetime of the battery (in months). Note that the PSM and PCM as defined above disallow load movement between clusters, i.e., the power demand sequence for each cluster are fixed inputs.

*Role in CDN Power Provisioning.* As mentioned earlier, both PSM and PCM are formulated as *provisioning* problems in the sense that they can guide the CDN operator on how to provision the power supplies and batteries for each of the clusters of the CDN. PCM and PSM are solved in an *offline* fashion using future predictions of power demand. We expect such power demand predictions will be derived from past trace data, much like the trace data that we use in our paper. Besides helping us answer fundamental questions on the efficacy of batteries in CDN deployments, we expect that our work will provide techniques for the CDN operator to analyze potential savings for different deployment scenarios, server energy usage trends, future power cost regimes, and different battery technologies. The CDN operator can use our techniques periodically to determine what power resources need to be contracted from the colocation facility and what battery resources need to be provisioned along with the servers.

### 3.1 A Characterization of the Optimum Power

We study a simple case of the PSM problem and fully characterize the minimum power supply  $P_{opt}$  needed per server to satisfy the power demand of a single cluster, given a battery with capacity  $B$  per server. The characterization in this section builds intuition that is critical for understanding the properties and tradeoffs in optimizing the power supply. In addition, it provides faster algorithms for the simpler cases of PSM, while the LP approach outlined in Section 3.2, albeit slower, is better suited for the more complex forms of PSM.

**THEOREM 3.1.** *Consider a power demand sequence  $\langle p_i \rangle$ ,  $1 \leq i \leq T$ , for a cluster with  $S$  servers, battery capacity  $B$  per server, loss factor  $\alpha$  and maximum charge rate  $P_c^{max}$  per server. Suppose that the battery is full at the start of first time slot. The minimum required power supply to accommodate the power demand sequence is  $P_{opt} = \max_{1 \leq i \leq j \leq T} P(i, j)$ , where*

$$P(i, j) = \min\{P : (1 - \alpha) \sum_{k \in U(i, j, P)} \min\{(P - p_k), P_c^{max}\} + B \geq \sum_{k \in L(i, j, P)} (p_k - P)\},$$

where  $U(i, j, P) = \{k : i \leq k \leq j \text{ and } P \geq p_k\}$  and  $L(i, j, P) = \{k : i \leq k \leq j \text{ and } P < p_k\}$ .

**PROOF.** First, we show that  $P_{opt} \geq P(i, j)$ , for any time interval  $(i, j)$ ,  $1 \leq i \leq j \leq T$ . Let  $(i, j)$  be any time interval and  $P$  be the provisioned power. The battery (potentially) gets charged at step  $k \in U(i, j, P)$  and must get discharged at each step  $k \in L(i, j, P)$  so as to not exceed the supplied power  $P$  (See Figure 5). At the beginning of time  $i$ , the maximum energy available in the battery is clearly  $B$ . The maximum energy entering the battery during time step  $k \in U(i, j, P)$  is at most  $(1 - \alpha) \min\{(P - p_k), P_c^{max}\}$ , since  $P_c^{max}$  is the maximum rate at which the battery can be charged and  $\alpha$  is the loss factor. Likewise, for any  $k \in L(i, j, P)$ , the rate at which the battery must be discharged is  $\min\{(p_k - P), P_d^{max}\} = p_k - P$ , since  $P_d^{max} = P_{peak} \geq p_k$ . Now, if a power supply of  $P$  and a battery capacity of  $B$  satisfies the power demand sequence from time  $i$  to  $j$ , the total energy already present or flowing into the battery in that time interval is at least the total energy flowing out (see Figure 5). That is,

$$(1 - \alpha) \sum_{k \in U(i, j, P)} \min\{(P - p_k), P_c^{max}\} + B \geq \sum_{k \in L(i, j, P)} (p_k - P). \quad (2)$$

Since  $P(i, j)$  is smallest value of  $P$  that satisfies Equation 2,  $P_{opt} \geq P(i, j)$ . Now, since the above is true for any  $i, j$ ,



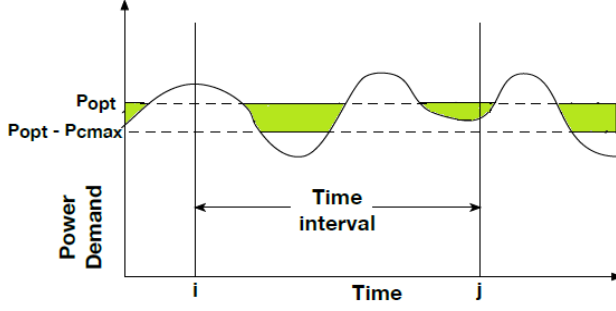


Figure 5: The optimum supplied power  $P_{opt}$  is the smallest value such that for any time interval the sum of the battery capacity  $B$  and the total possible charging in that interval ( $\alpha$  times the shaded area between the two dotted lines) is at least the total discharging required for that interval (area above the dotted line), where  $P_c^{max}$  is the maximum charge rate of the battery.

$$1 \leq i < j \leq T,$$

$$P_{opt} \geq \max_{1 \leq i < j \leq T} P(i, j). \quad (3)$$

Now, we show that provisioning a power supply  $P$  equal to the RHS of Equation 3 is *sufficient* to satisfy the power demand of the cluster with battery of  $B$  per server. This shows that  $P_{opt}$  equals the RHS of Equation 3, since the RHS is a lower bound on required power supply. Let  $P$  equal the RHS of Equation 3 and let charging and discharging decisions be made in a simple manner, i.e., if  $P > p_k$  then the battery is charged until capacity is reached and when  $P < p_k$  the battery is discharged at a rate of  $p_k - P$ . It suffices to show that the power demand sequence  $\langle p_i \rangle$ ,  $1 \leq i \leq T$ , can be satisfied with the power supply  $P$  and battery capacity  $B$ . For contradiction, suppose that this were not true. Then, there must be a time  $j$  where too much power needs to drawn from the battery causing it to underflow. Now, let  $i$  be the largest time  $i < j$  such that the battery is full at the start of that time step. Such an  $i$  always exists, since battery is full at the start of the first time slot. We are guaranteed that there is no battery overflow in the time interval  $(i, j)$  since the battery charge level is always less than  $B$  at any time  $i < k \leq j$ . Thus, we know that any charging opportunity is fully utilized during this time interval without a battery overflow. Therefore, the total charge entering the battery in time interval  $(i, j)$  is  $(1 - \alpha) \sum_{k \in U(i, j, P)} \min\{(P - p_k), P_c^{max}\}$ . If there is a battery underflow at time  $j$ , then we can conclude that amount of charge in the battery at time  $i$  plus the amount of energy that flowed into the battery in time interval  $(i, j)$  is smaller than the amount of energy that flowed out of the battery in same time interval, i.e.,

$$(1 - \alpha) \sum_{k \in U(i, j, P)} \min\{(P - p_k), P_c^{max}\} + B < \sum_{k \in L(i, j, P)} (p_k - P).$$

Since we chose  $P$  to equal the RHS of Equation 3, we know that  $P$  obeys Equation 2. Contradiction.  $\square$

The above characterization enables a fast algorithm to compute the optimum power supply.

**COROLLARY 3.2.** *Given the power demand sequence  $\langle p_i \rangle$ , for  $1 \leq i \leq T$  and a battery with capacity  $B$  per server, the optimum power supply required  $P_{opt}$  can be computed in time  $O(T^2 \log(P_{peak}))$ .*

**PROOF.** We perform binary search to find the smallest value of  $P$ ,  $0 \leq P \leq P_{peak}$ , that satisfies Equation 2 for all time intervals  $(i, j)$ ,  $1 \leq i < j \leq T$ . For a given  $P$ , checking Equation 2 for all time intervals  $(i, j)$  can be performed in  $O(T^2)$  time using dynamic programming. We define  $DIFF_P(i, j)$  to be the difference between the LHS and RHS of Equation 2. It is easy to see that  $DIFF_P(i, j)$  can be computed from  $DIFF_P(i, j - 1)$  in constant time by adding in the terms that correspond to time  $j$ . Thus, for a given  $P$ , we can compute  $DIFF_P(i, j)$  for all values  $1 \leq i \leq j \leq T$  in  $O(n^2)$  time. We can ascertain whether or not  $P$  satisfies the power demand by examining whether all values of  $DIFF_P(i, j)$  are non-negative in  $O(T^2)$  time. We perform binary search for the smallest value  $P$  using the above process to decide if a value of  $P$  is a feasible one. Since we try  $O(\log P_{peak})$  values for  $P$  each taking  $O(T^2)$  time, the total time is  $O(T^2 \log(P_{peak}))$ .  $\square$

### 3.2 An LP approach to Power Optimization

We provide a more general linear programming (LP) framework that is capable of solving more complex variants of the PSM and PCM problem. While our prior characterization solves for the optimum power supply for each cluster individually,

one can equivalently solve for all clusters simultaneously in the LP formulation as follows. Once stated as an LP, we use the CPLEX package [4] to solve the LP. For our traces, our formulation involves roughly 1.7 million variables and takes between 30 to 60 minutes solve, which is acceptable since we envision our optimizations to be performed in an offline fashion on historical or predicted trace data.

The objective function for the PSM problem is simply the total supplied power that needs be minimized, i.e.,

$$\text{Minimize } TC = \sum_{i \in \mathcal{C}} S_i P_i, \quad (4)$$

where  $\mathcal{C}$  is the set of all clusters,  $P_i$  is the power supply required per server in cluster  $i$ , and  $S_i$  is the number of servers in cluster  $i$ . The power demand  $p_{i,t}$  of a server in cluster  $i$  at time  $t$  is satisfied partly from the power supply and partly by discharging the battery: For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$p_{i,t} = \text{SupplyToServer}_{i,t} + \text{BattToServer}_{i,t}. \quad (5)$$

The power provided from the power supply to the server is non-negative: For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$\text{SupplyToServer}_{i,t} \geq 0. \quad (6)$$

Power from the battery to the server is non-negative and at most the maximum discharge rate: For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$0 \leq \text{BattToServer}_{i,t} \leq P_d^{\max}. \quad (7)$$

The provisioned power supply  $P_i$  at cluster  $i$  is partly used to power the server and partly used to charge the battery: For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$\text{SuppToServer}_{i,t} + \text{SuppToBatt}_{i,t} \leq P_i. \quad (8)$$

The power supplied to the battery for charging is non-negative and at most the maximum charge rate: For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$0 \leq \text{SuppToBatt}_{i,t} \leq P_c^{\max}. \quad (9)$$

Energy stored in the battery  $\text{Energy}_{i,t}$  at cluster  $i$  at the end of time  $t$  obeys the following: For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$\begin{aligned} \text{Energy}_{i,t} = & \text{Energy}_{i,t-1} + \text{SuppToBatt}_{i,t}(1 - \alpha) \Delta t \\ & - \text{BattToServer}_{i,t} \Delta t, \end{aligned} \quad (10)$$

where the second term above represents the charging of the battery in time step  $t$ , the third term represents the discharging of the battery,  $\alpha$  is the loss rate, and  $\Delta t$  is the length of a time slot (which is 5 minutes in our experiments). Energy stored in the battery at cluster  $i$  is no more than the battery capacity  $B_i$  and no less than zero (no battery underflow): For all  $i \in \mathcal{C}, 1 \leq t \leq T$ ,

$$0 \leq \text{Energy}_{i,t} \leq B_i. \quad (11)$$

The initial and final condition of the battery can be specified in several ways. One could posit that the battery is full at time zero, i.e.,  $\text{Energy}_{i,0} = B_i$ , for all  $i \in \mathcal{C}$ . Alternately, we can posit that energy in the start of the time period equals the energy at the end of the time period so that there is no net charge or discharge, i.e.,  $\text{Energy}_{i,0} = \text{Energy}_{i,T}$ , for all  $i \in \mathcal{C}$ .

Given a power demand sequence  $\langle p_{i,t} \rangle, i \in \mathcal{C}$  and  $1 \leq t \leq T$ , a set of  $S_i$  servers in cluster  $i, i \in \mathcal{C}$ , and battery deployments  $B_i, i \in \mathcal{C}$ , the above constraints from Equations 5 to 11 and the objective function of Equation 4 can be input into an LP solver to produce the optimum value of the total provisioned power  $TC$ . While the algorithm in Section 3.1 can solve the basic TSM problem above, the LP framework is particularly useful for the following variants that we study.

**Bounding the Battery Cycle Rate.** When the lifetime of the battery is a concern, we need to bound the number of times the battery can discharge per unit time. We can introduce an additional constraint that bounds the cycle rate to be below a prescribed value of  $\text{MaxCycleRate}$ : For all  $i \in \mathcal{C}$ ,

$$\frac{\sum_{1 \leq t \leq T} \text{BattToServer}_{i,t}}{B_i T} \leq \text{MaxCycleRate}. \quad (12)$$

**Power Cost Minimization (PCM).** We simply replace the objective function of Equation 4 by

$$\text{Minimize } TC = c_p \sum_i S_i P_i + \frac{c_b}{L} \sum_i B_i S_i, \quad (13)$$

where  $P_i$  is the power supply (in KWs) provisioned per server in cluster  $i$ ,  $B_i$  is the battery capacity provisioned per server and  $S_i$  is the number of servers in cluster  $i$ ,  $c_p$  is the unit power price (in \$/KW/month),  $c_b$  is the unit battery price

(\$/KWH/month), and  $L$  is the expected lifetime of the battery (in months). The inputs to PCM is the power demand sequence  $\langle p_{i,t} \rangle$ ,  $i \in \mathcal{C}$  and  $1 \leq t \leq T$ , a set of server deployments  $S_i$  servers,  $i \in \mathcal{C}$ , and costs  $c_p$ ,  $c_b$ , and  $L$ . The amount of power  $P_i$  and battery  $B_i$  are variables that are computed by the LP so as to minimize the new cost objective function.

**Heterogenous versus Homogenous Batteries.** The above formulation models the more general case of heterogenous batteries. However, unless specified otherwise, we assume that the battery per server is uniform throughout the CDN, i.e., for all  $i \in \mathcal{C}$ , we replace  $B_i$  with a single variable  $B$  throughout the LP.

**Load Balancing.** To model global load balancing, we compute the total load across all clusters that enter the CDN denoted by  $\langle L_t \rangle$ ,  $1 \leq t \leq T$ , where  $L_t$  is the incoming total load at time  $t$  and can be derived from our traces. The global load balancer portions the incoming load across clusters to such that cluster  $i$  receives an average load of  $l_{i,t}$  per server. This can be captured by the following constraint: for all  $1 \leq t \leq T$ ,  $\sum_{i \in \mathcal{C}} S_i l_{i,t} = L_t$ . The corresponding power demand sequence  $p_{i,t} = \text{power}(l_{i,t})$ , for all  $i \in \mathcal{C}, 1 \leq t \leq T$ . We then solve the LP with these additional constraints added to the existing constraints of Equations 5 to 11.

## 4. POWER SAVINGS ANALYSIS

The power savings of a CDN is defined to be the percentage reduction in the total supplied power  $TP$  due to the use of batteries. Thus, power savings equal

$$100 \times \frac{\text{Min TP without batteries} - \text{Min TP with batteries}}{\text{Min TP without batteries}}. \quad (14)$$

Note that in our definition of power savings above, we compare the minimum total power supply ( $TP$ ) achievable with and without batteries. In reality, a CDN will likely be over-provisioned by the ‘‘safety factor’’ above the minimum power supply required to satisfy its expected maximum power demand. The safety factor provides some ‘‘headroom’’ for unexpected spikes in load or power demand; recall Figure 1. It is reasonable to assume that the safety factor is the same with or without batteries and hence does not affect the power savings. We now ascertain the power savings achievable by a CDN as a function of the (i) battery capacity, (ii) battery cycle rate, (iii) discharge/charge ratio, (iv) power proportionality factor, and (v) battery deployment model.

### 4.1 Battery Capacity

First, we prove that the power savings is a concave function of the deployed battery capacity for which law of diminishing returns apply, i.e., an initial increase in battery capacity can significantly increase the power savings, while a further marginal increase in battery size may lead only to a smaller marginal increase.

**THEOREM 4.1.** *For any power demand sequence  $\langle p_t \rangle$ ,  $1 \leq t \leq T$  for a specific cluster, the optimum power supply  $P_{opt}$  for that cluster is a non-increasing convex function of the battery capacity  $B$ . Thus, total power savings across the entire CDN is a non-decreasing concave function of capacity  $B$ .*

**PROOF.** For two battery capacities  $B$  and  $B'$  provisioned in a given cluster, let  $P$  and  $P'$  be the optimum power supply values for that battery capacity respectively. It suffices to show that a supplied power value of  $P'' = (P + P')/2$  satisfies power demand sequence with a battery of capacity  $B'' = (B + B')/2$ . One can view a battery with capacity  $B''$  and a charge rate of  $P_c^{max}$  as two batteries with capacity  $B/2$  and  $B'/2$ , each with a charge rate of  $P_c^{max}/2$ . Since  $P$  (resp.  $P'$ ) is feasible for serving the power sequence  $\langle p_t \rangle$  with a battery of capacity  $B$  (resp.  $B'$ ),  $P/2$  (resp.  $P'/2$ ) is feasible for serving the power sequence  $\langle p_t/2 \rangle$  with a battery of capacity  $B/2$  (resp.,  $B'/2$ ). That is, one can take a feasible schedule for  $P$  (resp.,  $P'$ ) with battery capacity  $B$  (resp.,  $B'$ ) and obtain another feasible schedule by simply dividing the power supply, battery capacity, battery charge rate, and power demand by a factor of two. Putting together the two feasible schedules constructed in this fashion, we obtain a feasible schedule with power supply  $P'' = (P + P')/2$ , and battery capacity  $B'' = (B + B')/2$  for the original power demand sequence  $\langle p_t \rangle$ , implying that  $P''$  is at least the optimum power supply value for battery capacity  $B''$ . Thus,  $P_{opt}$  for any cluster is a non-increasing convex function of battery capacity, implying from Equation 14 that total power savings is a non-decreasing concave function.  $\square$

*Experimental Results.* We simulate a typical deployment of homogenous batteries with discharge/charge ratio  $r = 5$ , and servers with PPF of 0.6. Figure 6 shows the optimal power savings for batteries of various capacities for power demand derived from our Akamai’s CDN traces described in Section 2.4 computed using the algorithms described in Section 3. As suggested by Theorem 4.1, the power savings vs. battery capacity curve is concave with diminishing gains as battery capacity increases. We visually discern three distinct regions in this curve: (i) ‘‘I’’ where relatively high improvements result from additional battery capacity - this spans till the battery capacity of about 5 minutes where the power savings are about 7%, (ii)

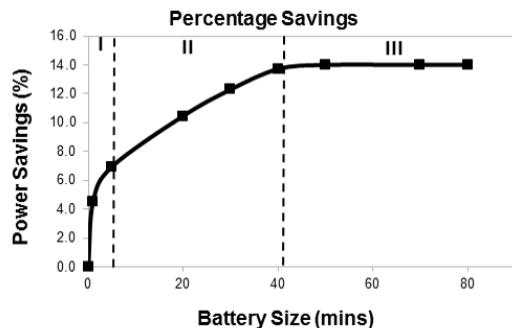


Figure 6: Power savings increase rapidly till about a capacity of 5 minutes, grow more slowly till about 40 minutes before reaching a point of diminishing return.

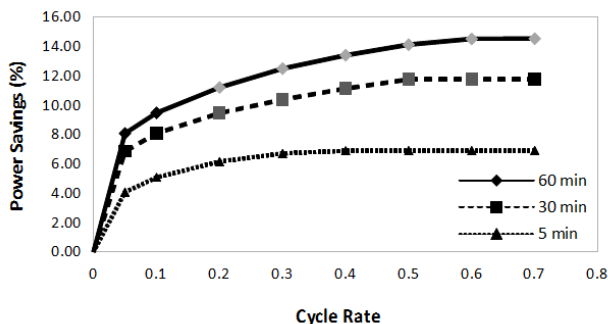


Figure 7: Power savings when the cycle rate is bounded. Most savings can be realized with a small cycle rate.

“II” where the improvement in power savings slows down - this spans till the battery capacity of about 40 minutes where the power savings are about 14%, and (iii) “III,” which spans beyond battery capacity of about 40 minutes and where additional capacity yields negligible or no additional savings.

OBSERVATION 4.2. *Even small battery capacities of 1-5 minutes can offer appreciable power savings of about 4-7%.*

This bodes well for their immediate use for power savings in CDNs since these sizes are comparable to those in currently deployed systems - not only in the form of traditionally popular centralized UPS units [15] but also in rack/server-level batteries in emerging designs [22].

OBSERVATION 4.3. *The higher power savings of up to 14% are yielded by a battery capacity of 40 minutes.*

Volume constraints might prohibit packing several tens of minutes of battery capacity at the server-level UPS units [25]. This suggests that, in order to realize the full extent of power savings possible via batteries, our CDN may want to choose a design that provides its clusters with access to rack-level batteries that can pack adequate capacity.

## 4.2 Battery Lifetime

Since a battery can only last a bounded number of cycles in its lifetime (Figure 2), a high cycle rate degrades its the lifetime. We ask how much of the power savings is obtainable with a bound on the cycle rate (see Equation 12) in comparison with the power saving achievable with no bound on the cycle rate at all. Surprisingly, as shown in Figure 7, a small cycle rate of 0.33 is sufficient to obtain most of the power savings.

OBSERVATION 4.4. *Most of the power savings obtainable with an unbounded cycle rate are achievable by charging the battery to full capacity once every three days on average (cycle rate of 0.33 per day), i.e., 86%, 88.3%, and 97% of the power savings are obtainable for a 60-minute, 30-minute, and 5-minute battery, respectively with the small cycle rate. This is compatible with the battery lasting about 5 years per the lifetime chart.*

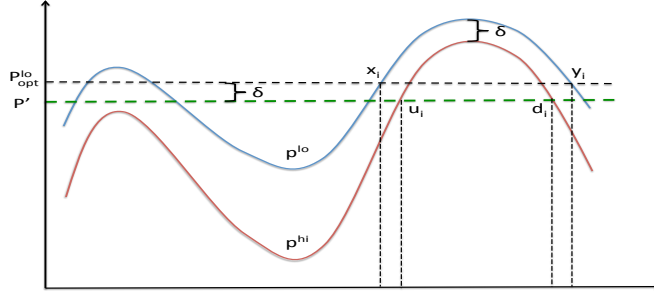


Figure 8: Illustration of the power demand sequence and threshold policies for servers with low and high PPF.

### 4.3 Discharge/Charge Ratio

While we use a value of discharge/charge ratio ( $r$ ) of 5 that is typical for lead-acid batteries, in this section we explore if smaller values of  $r$  can impact savings. Notice that, given that  $P_d^{max} = P_{peak}$ , this amounts to exploring different values of  $P_c^{max}$  allowed by the battery technology being used. Certain battery technologies offer better  $r$  than lead-acid. Two prominent examples of such technologies include ultra-capacitors and flywheels, both of which offer  $r$  close to 1 [25]. Theoretically, for a given battery size, a battery technology that allows faster charging (than another technology) can allow more energy to be accumulated during a given charging period, thus allowing a possibly higher reduction in supplied power later. However, repeating our experiments from Section 4.1 with  $P_c^{max}$  higher than 50W, we do not find significant additional benefit than what we obtained for  $r = 5$ . The reason is that, as we saw in Section 3.1, the power available to charge the battery at step  $k$  is  $\min\{(P - p_k), P_c^{max}\}$ , where  $P$  is supplied power at a cluster and  $p_k$  is power demand at step  $k$ . As noted earlier, the CDN's servers had load of at most 40% most of the time resulting in a typical power demand  $p_k$  of at most 160W, a typical value of  $P - p_k$  was at least  $P_{idle} - p_k = 60W$ . Thus, any  $P_c^{max}$  larger than 60W made no difference since first term of the minimization would be smaller.

**OBSERVATION 4.5.** *The discharge/charge ratio  $r$  offered by lead-acid batteries is adequate for achieving the all of the realizable power savings in our CDN.*

### 4.4 Power Proportionality

We first provide a theoretical basis for why we can expect larger savings due to batteries as the servers become more power proportional. Next, we explore this issue empirically.

**THEOREM 4.6.** *Given a server-level load sequence  $\langle l_t \rangle$ ,  $1 \leq t \leq T$ . Suppose that the battery is full at the start of the first time slot. Then the optimal power savings realized for a server with peak power  $P_{peak}$ , battery size  $B$ , and PPF of  $f^{hi}$  is greater than or equal to that realized for a server with the same peak power and battery size but a lower PPF  $f^{lo}$  (i.e.,  $f^{hi} > f^{lo}$ ).*

**PROOF.** Let us denote by  $\langle p_t^{hi} \rangle$  and  $\langle p_t^{lo} \rangle$  the power demand sequences induced on servers with PPFs  $f^{hi}$  and  $f^{lo}$ , respectively, by the given load demand  $\langle l_t \rangle$ . Since  $f^{hi} > f^{lo}$ , using (1) we have  $p_t^{lo} \geq p_t^{hi}$ ,  $\forall t$ , with equality occurring when  $l_t = 1$ . Let us denote by  $\Delta_t$  the difference  $p_t^{lo} - p_t^{hi}$ . We first observe that  $l_i > l_j \Rightarrow \Delta_i < \Delta_j$ . Consequently, the minimum value of the difference between power demands (call it  $\delta$ ) occurs during a time slot (call it  $t_m$ ) where the load takes its maximum value. That is,  $\delta = \min_t \Delta_t = p_{t_m}^{lo} - p_{t_m}^{hi}$ , where  $t_m = \arg \max_t l_t$ .

Let  $P_{opt}^{lo}$  denote the minimum required power supply for the sequence  $\langle p_t^{lo} \rangle$  for the server with PPF  $f^{lo}$ . Note that this is fully characterized by Theorem 3.1. Furthermore, as shown in Theorem 3.1, the threshold based policy that discharges the battery when  $P_{opt}^{lo} < p_t^{lo}$  at a rate of  $p_t^{lo} - P_{opt}^{lo}$  and charges until capacity is reached whenever  $P_{opt}^{lo} \geq p_t^{lo}$  is optimal. Note that the peak power reduction achieved by this policy is equal to  $p_{t_m}^{lo} - P_{opt}^{lo}$ .

We now show that a similar threshold based policy that uses the threshold  $P' = P_{opt}^{lo} - \delta$  on the sequence  $\langle p_t^{hi} \rangle$  for the server with PPF  $f^{hi}$  is feasible. This means that a peak power reduction of at least  $p_{t_m}^{hi} - P' = (p_{t_m}^{lo} - \delta) - (P_{opt}^{lo} - \delta) = p_{t_m}^{lo} - P_{opt}^{lo}$  can be achieved for the load sequence  $\langle l_t \rangle$  under the server with higher PPF. Since the minimum total power supply (TP) achievable without batteries under the server with higher PPF (i.e.,  $p_{t_m}^{hi}$ ) is smaller than that under the server with lower PPF (i.e.,  $p_{t_m}^{lo}$ ), it follows from (14) that the optimal power savings under the server with higher PPF exceed those under the server with lower PPF.

Let  $E_t^{hi}$  denote the total energy stored in the battery at time slot  $t$  when the threshold policy described above is used on the power demand sequence  $\langle p_t^{hi} \rangle$  for the server with PPF  $f^{hi}$ . Define  $E_t^{lo}$  in a similar way for the server with PPF  $f^{lo}$  and power demand sequence  $\langle p_t^{lo} \rangle$  that uses the threshold  $P_{opt}^{lo}$ . Let  $u_i$  denote the sequence of time slots at which the power

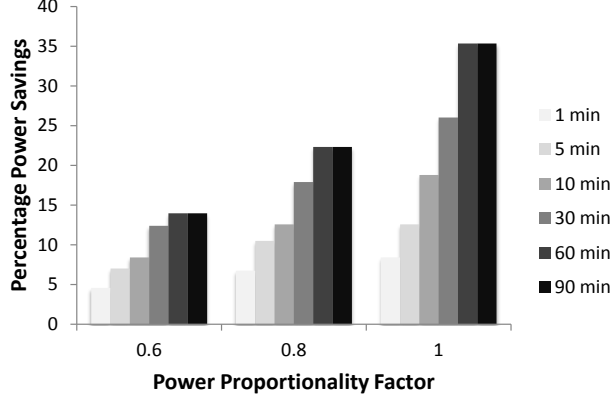


Figure 9: Power savings improve as PPF increases.

demand sequence  $\langle p_t^{hi} \rangle$  intersects the threshold  $P'$  such that  $p_t^{hi}$  is increasing at  $t = u_i$ . Likewise, let  $d_i$  denote the sequence of time slots at which the sequence  $\langle p_t^{hi} \rangle$  intersects the threshold  $P'$  such that  $p_t^{hi}$  is decreasing at  $t = d_i$ . For simplicity, we assume that  $p_0^{lo} \leq P_{opt}^{lo}$  so that  $u_i \leq d_i$  for all  $i$ . The opposite case where  $d_i \geq u_i$  for all  $i$  can be treated similarly.

For every interval  $[u_i, d_i]$  of the sequence  $\langle p_t^{hi} \rangle$  in which  $p_t^{hi} \geq P'$ , there is a corresponding interval  $[x_i, y_i]$  of the sequence  $\langle p_t^{lo} \rangle$  in which  $p_t^{lo} \geq P_{opt}^{lo}$  such that  $x_i \leq u_i$  and  $y_i \geq d_i$ . This is shown in Lemma 4.7.

We now show that  $E_{u_i}^{hi} \geq E_{x_i}^{lo}$  and  $E_{d_i}^{hi} \geq E_{y_i}^{lo}$  for all time slots  $u_i, d_i, x_i, y_i$ . This, combined with the fact that the total energy required from the battery in the interval  $[u_i, d_i]$  for the sequence  $\langle p_t^{hi} \rangle$  (equal to the area under the red curve and green dotted line in the interval  $[u_i, d_i]$  in Fig. 8) is smaller than the total energy required from the battery in the interval  $[x_i, y_i]$  for the sequence  $\langle p_t^{lo} \rangle$  (equal to the area under the blue curve and black dotted line in the interval  $[x_i, y_i]$  in Fig. 8) shows that the threshold policy is feasible.

To show that  $E_{u_i}^{hi} \geq E_{x_i}^{lo}$  for all time slots  $u_i, x_i$ , we use induction. Note that this trivially holds for  $u_1, x_1$ . This is because  $E_{u_1}^{hi} = B \geq E_{x_1}^{lo}$  since there has been no discharge from the battery yet. Next,  $E_{d_1}^{hi} \geq E_{y_1}^{lo}$  since the total amount discharged over the interval  $[u_1, d_1]$  under the sequence  $\langle p_t^{hi} \rangle$  is smaller than the total amount discharged over the interval  $[x_1, y_1]$  under the sequence  $\langle p_t^{lo} \rangle$ .

Now suppose these relations hold for some  $i > 1$ . We show that these holds for  $i + 1$  as well. Consider the interval  $[d_i, u_{i+1}]$  under the sequence  $\langle p_t^{hi} \rangle$ . Using Lemma 4.7, we have that the length of this interval exceeds that of the interval  $[y_i, x_{i+1}]$  under the sequence  $\langle p_t^{lo} \rangle$ . This means that the total amount by which the battery gets recharged is *at least* as large for the higher PPF case. This means that  $E_{u_{i+1}} \geq E_{x_{i+1}}$ . Finally,  $E_{d_{i+1}} \geq E_{y_{i+1}}$  using the same argument as before.  $\square$

LEMMA 4.7. *For every interval  $[u_i, d_i]$  of the sequence  $\langle p_t^{hi} \rangle$  in which  $p_t^{hi} \geq P'$ , there is a corresponding interval  $[x_i, y_i]$  of the sequence  $\langle p_t^{lo} \rangle$  in which  $p_t^{lo} \geq P_{opt}^{lo}$  such that  $x_i \leq u_i$  and  $y_i \geq d_i$ .*

PROOF. This can be seen as follows. Suppose  $p_{u_i}^{hi} = P' = P_{opt}^{lo} - \delta$  and  $p_t^{hi}$  is increasing at  $t = u_i$ . Since  $p_t^{lo} \geq p_t^{hi} + \delta \forall t$ , we get  $p_{u_i}^{lo} \geq p_{u_i}^{hi} + \delta = P_{opt}^{lo}$  and  $p_t^{lo}$  is increasing at  $t = u_i$  (by linearity of (1)). Thus, the point  $x_i$  at which  $p_t^{lo}$  crossed  $P_{opt}^{lo}$  cannot exceed  $u_i$ . Using similar arguments, it can shown that  $y_i \geq d_i$ .  $\square$

Figure 9 shows how power savings vary with PPF for different battery sizes. One can see that for any battery capacity, power savings increase with an increase in PPF. The intuitive reason is that as PPF increases, the constant part of the power demand as represented by  $P_{idle}$  shrinks, increasing the relative variability of the power demands. The increased variability increases the opportunities for a battery to decrease the peaks and hence decrease the supplied power. For a small battery of capacity of 5 minutes (in range I), the power savings across the entire CDN ranged from 7% for a PPF of 0.6, to 12.57% for a PPF of 1.0, with a value of 10.5% for a PPF 0.8. Likewise, for a medium-sized battery of capacity 30 minutes (in range II), the power savings ranged from 12.4% to 26%, with a value of 17.9% for PPF of 0.8. And, for a large battery of capacity 60 minutes, the power savings ranged from 13.9% to 35.3%, with a value of 22.3% for PPF of 0.8.

A perhaps less intuitive point to note is that as server PPFs increase, there are greater incentives to making large batteries even larger. As an example, consider how the power savings improvement going from the battery capacity of 30 minutes to that of 60 minutes itself improves with PPF: power savings increase by 1.5% for PPF=0.6, 4.4% for PPF=0.8, and 9.3% for PPF=1.

OBSERVATION 4.8. *Improving the power proportionality of servers improves the power savings from 7-14% with current servers with PPF=0.6, to a range of 10.5%-22% with next-generation servers with PPF=0.8, to the range 13.9-35.3% in the*

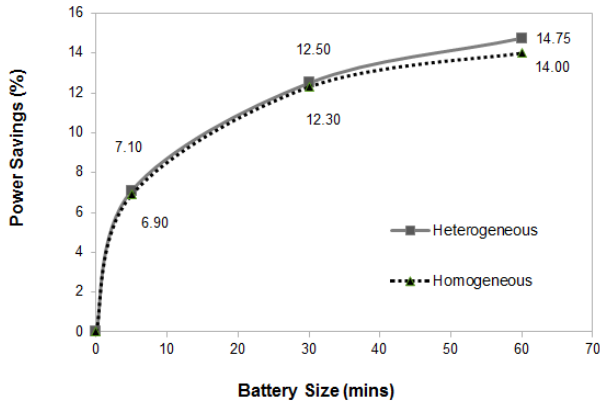


Figure 10: Heterogenous batteries provide little additional power savings compared to homogenous ones.

hypothetical best-case of perfect power proportionality  $PPF=1$ . In addition to increased power savings, a greater  $PPF$  also increases the range of battery sizes that can offer such savings.

## 4.5 Heterogeneous Battery Provisioning

As Figure 10 shows, for the same amount of total battery capacity, distributing the batteries to clusters in a heterogenous manner with some clusters getting (potentially) more battery capacity than others produces little additional savings over providing each server in each cluster the same amount of battery. The intuitive reason is that power demands across all clusters (see Figure 4) had similar enough statistical characteristics in terms of day-night and weekly variations that the battery requirements across clusters were similar.

**OBSERVATION 4.9.** *Heterogenous batteries do not provide much additional savings over homogenous ones. The additional savings increase with battery capacity with an additional savings of 0.2% for a 5-minute battery, 0.2% for a 30-minute battery, and 0.75% for a 60-minute battery. Given the operational advantages, a homogenous deployment of batteries suffices.*

## 5. COST SAVINGS ANALYSIS

We analyze the cost savings (in \$) our CDN can obtain by employing batteries. This involves characterizing the trade-off between (i) the OpEx reduction offered by an investment in batteries in the form of power savings, and (ii) the CapEx increase for procuring the batteries. The cost saving for a CDN equals the reduction in total cost (TC) due to batteries, i.e.,

$$100 \times \frac{\text{Min TC without batteries} - \text{Min TC with batteries}}{\text{Min TC without batteries}},$$

where the “Min TC with batteries” is derived by minimizing the expression in Equation 13. Likewise, “Min TC without batteries” is the least power cost achievable without batteries, i.e., smallest cost with battery capacity of zero. Note that battery capacity can either be fixed or be a variable in the minimization where the best battery size that minimizes  $TC$  is picked by our optimization. It is easy to see that cost saving depends on the unit battery price ( $c_b$ ), the battery lifetime ( $L$ ), server power proportionality ( $PPF$ ), and the unit power price ( $c_p$ ). As outlined in Sections 2.1 and 2.3, we now analyze cost savings for a discrete set of values for each parameter above that we believe characterizes the possible current and future scenarios.

### 5.1 Battery Characteristics

In Figure 11, we derive the cost savings assuming a low power price of \$150 per KW per month for three different battery scenarios outlined in Section 2.1: typical ( $c_b=\$100/\text{kWh}$ ,  $L=5$  yr), conservative ( $c_b=\$300/\text{kWh}$ ,  $L=3$  yr), and worst case ( $c_b=\$500/\text{kWh}$ ,  $L=1$  yr). Except with the overly pessimistic “worst-case”, our cost savings results are encouraging: with the more realistic “typical” and “conservative” cases, we find that batteries are capable of turning a positive return-on-investment (ROI). We find cost savings of 10% with a 40-minute battery in the “conservative” case and 13.9% with a 60-minute battery in the “typical” case. Even for our “worst-case,” we find that up to 5.5% cost savings can be achieved with battery size of 5 minutes. Beyond the size of 60 minutes, battery investment actually becomes financially unsound for all cases, causing negative ROIs in the “worst-case.”

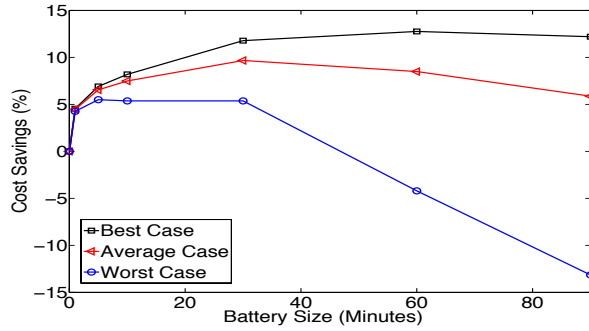


Figure 11: The cost savings offered by different battery sizes for our three battery cost/lifetime models, namely “typical,” “conservative,” and “worst-case.”

OBSERVATION 5.1. *Even for low power prices currently seen in the US and a low PPF of 0.6, an investment in lead-acid batteries can allow cost savings for our CDN of about 13.9% under typical assumptions for battery cost and 10% under more conservative assumptions. Furthermore, the optimal cost savings are achieved with larger battery capacities when the amortized battery cost ( $c_b/L$ ) is smaller.*

## 5.2 Power Proportionality

We now present the optimal cost savings for different PPFs. The cost savings improve with PPF for all three battery

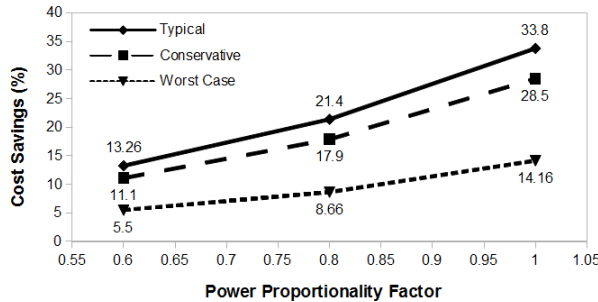


Figure 12: The cost savings improve as servers become more power proportional. We show the results for a 30 minute battery for our three different battery scenarios: typical, conservative and worst-case and typical power price of  $c_p = \$150$  per KW per month.

scenarios: typical, conservative and worst-case. Note that the improvement in cost savings is greater in absolute terms for the typical battery scenario as opposed to the worst-case battery scenario, since the amortized battery cost is cheaper in the first scenario than in the second.

OBSERVATION 5.2. *Batteries provide significantly higher cost benefits as servers become more power proportional. For a next-generation system with PPF=0.8, the cost savings for a typical battery are 21.4% and can rise even further to 33% as the PPF nears 1.0.*

## 5.3 Power Supply Prices

In Figure 13, we plot the cost savings for a typical battery as function of the supplied power costs ( $c_p$ ) in the range \$100-\$500/KWH/month. The cost savings are 13.26% for the “low” power prices currently in the US, which rises to 13.57% for “moderate” prices typical of Europe, and becomes even larger at 13.69% when the power prices are in the “high” range. A reason for the increased cost savings that you observe in the figure is that the amortized cost of the battery becomes less significant in relation to higher power costs, as  $c_p$  increases. Further, the amount of additional power savings needed to pay for more battery capacity is lower for larger  $c_p$ .

OBSERVATION 5.3. *As the unit power price increases from \$100/KW/month to \$500/KW/month, the cost savings increased*



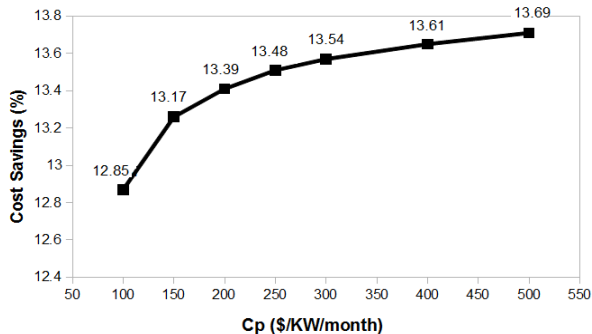


Figure 13: Cost savings increase with power price.

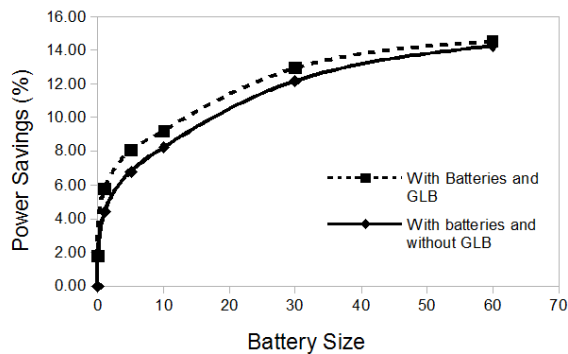


Figure 14: Power-aware GLB with batteries provides little additional benefit over using batteries alone.

from 12.85% to 13.69%. While the increase in cost savings is modest and less than 1%, the cost savings is much greater in absolute terms as the cost itself is about five times higher.

## 6. GLOBAL LOAD BALANCING

We are interested in understanding if it helps to make global load balancing (GLB) “power aware,” so that it can move traffic between clusters to “shave off peaks” and reduce the total power supply  $TC$ . One can view GLB as a mechanism complementary to batteries in that one could study the benefit of either mechanism alone, or both together. In an actual CDN, GLB must optimize two key objectives other than power. First, GLB must ensure that each user receives good performance by being served by a “nearby” cluster in the “network sense” [20]. Second, GLB must minimize bandwidth costs<sup>7</sup> paid by the CDN to each network provider. The GLB of a modern CDN is already optimized for performance [20] and bandwidth costs [7]. Thus far, by assuming that the load and power demands cannot be moved across clusters we ensured that the optimized performance and bandwidth costs reflected in the Akamai traces remain unaltered while we minimize power supply and power costs with batteries. In order to understand the potential impact of GLB, we study the *best case scenario* where we ignore performance and bandwidth costs and assume that users and their load can be moved to *any* cluster as long as the capacity constraints of the cluster are met.

As described in Section 3.2, the best case scenario for GLB can be incorporated into the power supply minimization problem by adding additional variables and constraints. In Figure 14, we show the power savings obtained with and without GLB. Using GLB alone without any batteries produced only a small power savings of 1.78%. The additional power savings due to GLB narrowed as we increased the battery capacity: a 5-minute battery produced an additional savings of 1.6% due to GLB, a 30-minute battery an additional savings of 0.8%, and a 60-minute battery an additional savings of only 0.3%. The small additional savings due to GLB leads us to conclude that a “power aware” GLB is not beneficial enough to implement from a power savings standpoint, especially in comparison with batteries that produce much larger gains. Incorporating performance

<sup>7</sup>The bandwidth cost of a cluster is computed by averaging the bytes sent by the CDN’s servers over 5-minute time intervals for each month. The cost is proportional to either the 95<sup>th</sup> percentile or the average of these 5-minute averages.

and bandwidth cost that is a critical component of any real-world global load balancer would restrict the inter-cluster load movements resulting in even less savings than the best case evaluated above.

The intuitive reason for the low additional power savings due to GLB is because the power demands across clusters of the CDN peak and ebb at around the same time following Web access patterns of the users across the country (see Figure 4). The power demands would be even more similar across clusters if GLB were to incorporate performance constraints since the available choices of clusters for a user would have to be “nearby” in the same geographical area with even more similarity in usage patterns. Thus, GLB is unable to find *significant* gains by “shaving peaks” in the power demand of one cluster and moving it into a “valley” in the power demand of another cluster. Note also that such opportunities are made rare by the requirement that the CDN’s load must be served in real-time, and cannot be “deferred” to a later time when the power demands are low.

*OBSERVATION 6.1. The maximum additional benefit of a power-aware GLB is small and at most 1.78%. Since the additional power savings are likely even smaller if performance and bandwidth costs are incorporated, we conclude that GLB is not an important mechanism for power supply or power cost savings.*

## 7. RELATED WORK

Batteries have a long history of use in disaster recovery (i.e., for handling failures of the power utility) and in situations such as battery-powered sensor or mobile environments [13] where power availability is intermittent. Recently, batteries have been studied for various forms of demand response in data centers, and we consider this body of work the closest to ours. These include using batteries to reduce the OpEx costs when the utility prices are time-varying [23] or when the utility bill depends on the peak power usage [15]. These also include using batteries to reduce the CapEx costs for provisioning data center power infrastructure [17, 25]. While benefiting directly from the characterization of battery properties in these papers, our work differs significantly from them. First, our work is in the context of a CDN - a highly distributed system with multiple deployments across the globe - as opposed to a single data center. This difference becomes especially stark when we consider GLB, which is unique to a CDN and does not apply to prior work on single data centers. Second, we work with a very different tariff model - the supplied power tariff model with long-term contracts that is standard in the CDN industry - which presents us with a very different optimization criterion than those arising in prior work. Third, a CDN’s load cannot be deferred and must be served in real-time. This results in different kinds of constraints in our optimization than for data centers where some of the workloads are delay-tolerant. Finally, to the best of our knowledge, our work is the first to evaluate the efficacy of batteries in futuristic settings where servers might improve in their power proportionality and electricity prices might be higher than today.

In a seminal work [21], the authors consider an Internet-scale distributed system with multiple deployments like a CDN. They show that if costs are based on electricity usage and if the power prices vary in real-time, global load balancing decisions can be made such that users are routed to locations with the cheapest power without significantly impacting user performance or bandwidth costs. Our work has a different focus as we use batteries, a power pricing model that is based not on usage but on how much power is provisioned, and power prices are fixed for the length of contract that is several months to a year. The distributed battery architecture that places batteries at the rack- or server-level that we consider in our work is deployed and used by Facebook, Google [22] and Microsoft [6]. While their primary motivation is to implement a distributed UPS system that costs less and wastes less energy than a centralized UPS, we use batteries for the different goal of power supply and power cost minimization.

Finally, there is relevant algorithmic work on minimizing the peak draw from an utility using batteries [8]. In fact, Theorem 3.1 can be viewed as a generalization of their work to a richer set of battery parameters. However, our work is distinguished from theirs in that we focus on power and cost savings in a CDN context, use a supplied power instead of the peak usage model, and focus on a multi-cluster distributed service. It is also worth noting that recently batteries have been considered for minimizing the peak power draw in residential settings [9, 24].

## 8. CONCLUSIONS

Our work proposed and provided strong evidence that a CDN could utilize batteries to significantly reduce both the total supplied power and the total power costs. This work is important as CDNs are growing rapidly and are already the platform for a significant fraction of Web, application, and media traffic world-wide. We showed that batteries can provide significant savings with current server and battery technologies, and that these savings increased rapidly as servers became more power-proportional. Furthermore, when the savings were viewed in absolute terms, batteries yielded significantly larger cost savings as power prices increased. Since we are likely to see increases in both power prices and more power-proportional

servers, and given the increasing focus on OpEx reduction in CDNs, we believe that our work will have a significant impact in establishing batteries as a key element in future CDN architecture. Besides our experimental work, our theoretical and algorithmic framework for formulating and solving power supply minimization and power cost minimization is an important contribution. In fact, we believe that our algorithms will form the basis for tools that CDNs could use to optimally provision power supplies and batteries, given a prediction of future load on the network.

It is important to note that we view our work as the first step in exploring the feasibility of batteries in a CDN by providing a high-level analysis of what might be possible. The favorable results of our analysis provide motivation for a more detailed exploration that considers the hardware design, placement, spatial requirements, and any additional operational costs associated with deploying batteries. Finally, we observe that our results extend beyond CDNs to other Internet-scale distributed networks that similarly deploy in co-location facilities, procuring power using the typical supplied power tariff model.

## 9. ACKNOWLEDGEMENTS

This work was supported, in part, by NSF CAREER award 0953541.

## 10. REFERENCES

- [1] Akamai Technologies. <http://www.akamai.com>.
- [2] Analysis: Colocation pricing trends. <http://www.datacenterknowledge.com/archives/2011/02/11/analysis-colocation-pricing-trends/>.
- [3] Battery FAQ. [http://www.windsun.com/Batteries/Battery\\_FAQ.htm](http://www.windsun.com/Batteries/Battery_FAQ.htm).
- [4] Cplex - ibm. <http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>.
- [5] Lead-acid battery costs. [http://photovoltaics.sandia.gov/Pubs\\_2010/PV%20Website%20Publications%20Folder\\_09/Hanley\\_PVSC09%5B1%5D.pdf](http://photovoltaics.sandia.gov/Pubs_2010/PV%20Website%20Publications%20Folder_09/Hanley_PVSC09%5B1%5D.pdf).
- [6] Microsoft reveals its specialty servers racks. <http://www.datacenterknowledge.com/archives/2011/04/25/microsoft-reveals-its-specialty-servers-racks/>.
- [7] M. Adler, R. K. Sitaraman, and H. Venkataramani. Algorithms for optimizing the bandwidth cost of content delivery. *Computer Networks*, 55(18):4007–4020, 2011.
- [8] A. Bar-Noy, M. Johnson, and O. Liu. Peak shaving through resource buffering. *Approximation and Online Algorithms*, pages 147–159, 2009.
- [9] S. Barker, A. Mishra, D. Irwin, P. Shenoy, and J. Albrecht. Smartcap: Flattening peak electricity demand in smart homes. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 67–75. IEEE, 2012.
- [10] L. Barroso and U. Holzle. The case for energy-proportional computing. *Computer*, 40(12):33–37, 2007.
- [11] C. Belady. In the data center, power and cooling costs more than the IT equipment it supports. *Electronics cooling*, 13(1):24, 2007.
- [12] A. Beloglazov, R. Buyya, Y. Lee, and A. Zomaya. A taxonomy and survey of energy-efficient data centers and cloud computing systems. *Arxiv preprint arXiv:1007.0066*, 2010.
- [13] Q. Cao, D. Fesehayee, N. Pham, Y. Sarwar, and T. Abdelzaher. Virtual battery: An energy reserve abstraction for embedded sensor networks. In *Real-Time Systems Symposium, 2008*, pages 123–133. IEEE, 2008.
- [14] J. Dille, B. M. Maggs, J. Parikh, H. Prokop, R. K. Sitaraman, and W. E. Weihl. Globally distributed content delivery. *IEEE Internet Computing*, 6(5):50–58, 2002.
- [15] S. Govindan, A. Sivasubramaniam, and B. Urgaonkar. Benefits and limitations of tapping into stored energy for datacenters. In *ISCA*, pages 341–352, Jun 2011.
- [16] S. Govindan, D. Wang, L. Chen, A. Sivasubramaniam, and B. Urgaonkar. Towards realizing a low cost and highly available datacenter power infrastructure. In *Hotpower*, Oct 2011.
- [17] S. Govindan, D. Wang, A. Sivasubramaniam, and B. Urgaonkar. Leveraging stored energy for handling power emergencies in aggressively provisioned datacenters. In *ACM ASPLOS*, pages 75–86, Mar 2012.
- [18] V. Mathew, R. K. Sitaraman, and P. J. Shenoy. Energy-aware load balancing in content delivery networks. In *INFOCOM*, pages 954–962, 2012.
- [19] D. Meisner, B. Gold, and T. Wenisch. Powernap: eliminating server idle power. *ACM Sigplan Notices*, 44(3):205–216, 2009.
- [20] E. Nygren, R. Sitaraman, and J. Sun. The Akamai Network: A platform for high-performance Internet applications. *ACM SIGOPS Operating Systems Review*, 44(3):2–19, 2010.
- [21] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs. Cutting the electric bill for internet-scale systems. In *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, pages 123–134. ACM, 2009.
- [22] D. Schneider and Q. Hardy. Under the hood at google and facebook. *Spectrum, IEEE*, 48(6):63–67, 2011.
- [23] R. Urgaonkar, B. Urgaonkar, M. J. Neely, and A. Sivasubramaniam. Optimal power cost management using stored energy in data centers. In *ACM SIGMETRICS*, pages 221–232, Jun 2011.
- [24] P. Ven, N. Hegde, L. Massoulié, and T. Salonidis. Optimal control of residential energy storage under price fluctuations. In *International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies*, 2011.
- [25] D. Wang, C. Ren, A. Sivasubramaniam, B. Urgaonkar, and H. K. Fathy. Energy storage in datacenters: What, where and how much? In *ACM SIGMETRICS*, Jun 2012.
- [26] A. Wierman, L. Andrew, and A. Tang. Power-aware speed scaling in processor sharing systems. In *INFOCOM 2009, IEEE*, pages 2007–2015. IEEE, 2009.