Thermal Energy Storage for Datacenters with Phase Change Materials

by

Matthew Allen Skach

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in The University of Michigan
2018

Doctoral Committee:

Assistant Professor Jason Mars, Co-Chair
Assistant Professor Lingjia Tang, Co-Chair
Professor Kevin Pipe
Professor Dean Tullsen, University of California San Diego

Matthew Skach

skachm@umich.edu

ORCID iD: 0000-0002-7198-7215

© Matthew Skach 2018

ACKNOWLEDGEMENTS

I would like to thank my advisors, Jason and Lingjia, for their guidance during my years as a graduate student. I would also like to extend a special thanks to Dean, who even though I am not one of his students has been like a third advisor providing invaluable guidance to me more times than I can count.

Thank you to also to Yunqi, Chang-Hong, Shih-Chieh, and all other students of Clarity Lab who went through (or are still going through) the same long, sometimes grueling, but always educational journey. Manish Arora, with whom I collaborated on this and other projects, and everyone else who I worked with or along side over the years.

Finally I thank my parents, Bill and Milly Skach, and my fiance, Yixing Shi, for their endless love and support.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Datacenters, vast warehouses containing millions of servers that run the internet and the cloud, have experienced double digit growth for almost two decades. Datacenters cost hundreds of millions of dollars, with the largest now exceeding over a billion dollars each, and consume enormous amounts of power–over 2% of all electricity in the US and projected to increase up to 10% by 2030.

The impact of such high compute density, with thousands of individual compute nodes packed together in a small space, is heat: every watt of power used by servers must be removed from the datacenter. This requires active cooling: air cooling is by far the most common with an air conditioner or other form of heat exchanger cooling air in the datacenter room then transporting heat outside the facility to heat exchanger or similar fixture. Such a system is simple, common, and functional, but inherently inefficient due to the nature of datacenter workloads.

Datacenters primarily server user facing workloads, that is: the user requests a search or sends and email and their query prompts load in the datacenter. The query is handled locally, on a relative geographic scale, to provide a low response time and positive user experience. This necessitates globally distributed datacenter capacity, but also creates a diurnal load pattern whereby datacenters are most heavily loaded during the peak hours when users in their region of service are awake and active online versus the off hours when users are offline or asleep and query requests are low. Because datacenter infrastructure must be provisioned for peak load, servers, power distribution, and cooling infrastructure is significantly underutilized most of the time.

This dissertation investigates the cooling needs of datacenters, and proposes to decouple the work and cooling needs. Specifically, we hypothesize that by storing thermal energy we can reshape the thermal profile of a datacenter to better balance cooling load throughout the day. We call this technique Thermal Time Shifting (TTS). First, we discuss how phase change materials (PCMs) enable TTS and evaluate the potential use scenarios of placing a small amount of PCM inside of servers for thermal energy storage. Next we dive deeper into the potential of thermal energy storage and propose Virtual Melting Temperatures (VMT), a technique that uses active job placement to control the melting and cooling of PCM to enable a much greater degree of control over the behavior of the thermal profile. Finally we propose and evaluate Thermal Gradient Transfer (TGT), a technique that uses direct water cooling to move heat straight from CPUs and GPUs to the wax for wider applicability and greater peak cooling load reduction.

CHAPTER I

Introduction

Modern datacenters are enormous, and growing. The largest currently consume over 200 MW of power [121, 40], and this class of "hyperscale" datacenters is expected to double to nearly 500 by 2020 [54] and cost over a billion dollars each [84]. Datacenters globally are notoriously inefficient [120] and expected to produce 3% of all greenhouse gas emissions by 2030 [8].

Within these datacenters, work is primarily driven by user load resulting in a diurnal cycle [14]. During the day, when users are awake and active, load is high and during the night when users are asleep the opposite is true. This creates to periods of operation–the "peak hours" and the "off hours"–and due to the nature of these user-facing workloads, most computation cannot be moved from one period to the other nor from one geographic region to another due to latency constraints (the time from when a user requests content over the internet to when that content arrives back ready for use).

This creates an inherent inefficiency: although all infrastructure (servers, power distribution cooling, etc.) must be sized for the peak load, it spends over half the significantly under utilized. This dual mode complicates the efficiency and cost, as a system designer must attempt to optimize both across both modes of operation. Cost and efficiency becomes even more complicated as the designer must also consider total cost of ownership, including initial equipment costs (capital expenditure) as well as power and ongoing maintainence (operating expenditure) throughout the lifetime of every component of the datacenter, which naturally all have different lifespans: servers are typically expected to last only 3-4 years while the cooling system and building infrastructure may be expected to operate for up to 20 [14, 65]. This is a vastly complex topic that cannot possibly be fully addressed in a single work.

In this dissertation, we focus on the problem of cooling large scale datacenters. Specifically, we break the assumption that cooling must be done in the same time frame as the work that produced it. The laws of thermodynamic are fixed, as far as we know, and so while more efficient processors produce less the heat the drive tocompute more data has created an increasing total power and power density in datacenters that must be removed to prevent overheating and/or thermal downclocking. However, by storing this energy we are able to delay when the heat must be removed.

In this work we propose, investigate, and project the the use of phase change materials to store thermal energy over multi-hour time periods to enable datacenters to move cooling work from the peak hours to off hours, enabling significant benefits from cooling undersubscription and cost savings.

1.1 Motivation

1.1.1 Thermal Time Shifting

Increasingly, a significant portion of the world's computation and storage is concentrated in the cloud, where it takes place in large datacenters, also referred to as ”warehouse-scale computers” (WSCs) [14]. One implication of this centralization of the world's computing infrastructure is that these datacenters consume massive amounts of power and incur high capital and operating costs. Even small improvements in the architecture of these systems can result in huge cost savings and/or reductions in energy usage that are visible on a national level [95, 65, 14, 83, 33, 82].

Due to the increasing computing density of these systems, a significant portion of the initial capital expenditures and recurring operating expenditures are devoted to cooling. To prevent high server failure, the cooling infrastructure must be provisioned to handle the peak demand placed on the datacenter. The scale of cooling infrastructure can cost over 8 million dollars [65], even if the datacenter only reaches peak utilization for a fraction of a load cycle. The cooling system also may become inadequate as servers are upgraded or replaced and the thermal characteristics of the datacenter change.

To mitigate these challenges, we propose the use of phase change materials (PCMs) to temporarily store the heat generated by the servers and other equipment during peak load, and release the heat when we have excess cooling capacity. The advantages of this approach may not be immediately obvious, because heat is not being eliminated, it is only stored temporarily then released at a later time. However, the key insight of this work is that the ability to store heat allows us to shape the thermal behavior of the datacenter, releasing the heat only when it is advantageous to do so.

1.1.2 Virtual Melting Temperature

The unprecedented growth of web and cloud services over the last decade spurred an enormous investment in datacenters, also called ”warehouse-scale computers” (WSCs) [14]. With the largest datacenter facilities consuming over 200 MW of power each [121, 40] and costing over a billion US dollars to build [84], datacenters represent a huge investment not only in server equipment but also in power, connectivity, facilities, and cooling infrastructure.

In the United States alone, datacenters consumed over 2% of all electrical power generated in 2014 [120, 77]. Extensive prior work investigates how to build more energy efficient processors and remove heat from the processors and servers better [13, 5, 79, 69, 131, 61, 132], but relatively little research has been published on how to maximize utilization and efficiency while minimizing cost to remove this heat from a datacenter facility.

In even a modestly sized datacenter the cooling system cost can exceed hundreds of thousands of dollars per MW of critical power, with large datacenters spending tens of millions in capital costs plus millions more per year in operating expenses to power and maintain the cooling system [65]. Datacenter cooling capital expenses in 2015 alone totaled more than $2.58 billion and are expected to exceed $6 billion by 2023 [1]. Prior work investigating server and datacenter cooling techniques demonstrate efficiency improvements [24, 58, 91], but cannot address the growing cost problem due to a critical assumption: that work and heat are coupled so that all of the heat must be removed at the same time the work is done.

In this chapter of the dissertation, we investigate ways to break this coupling and temporally separate heat produced from work done in a datacenter. We present two keys ideas: Thermal Time Shifting, the use of phase change materials to store heat until cooling capacity is available, and Virtual Melting Temperature, the active management of workload placement to create a more desireable thermal footprint for thermal energy storeage.

1.1.3 Thermal Gradient Transfer

Datacenters have been growing in size and number at an exponential rate over the last two decades, driven by the explosive growth of internet and cloud services [14]. As new workloads such as intelligent personal assistants (IPAs), eg. Amazon's Alexa and Apple's Siri, and other algorithms driven by machine learning become more common, the need for compute resources is only expected to grow, further increasing the rate of datacenter growth [49].

By 2030, datacenters globally are expected to consume just shy of 3,000 TWh of electricity annually [8] while global electricity demand is forecasted to reach 30,000 TWh [3]. That is, by 2030 as other energy sectors push for energy efficiency and the IT sector continues to grow, datacenters are expected to consume nearly 10% of the total electricity generated globally.

As the first order effects of this growth became apparent in recent years, much work has been dedicated to accelerators [88, 4, 22] and other techniques to reduce total energy consumption [13, 79, 16, 18, 15]. However, energy consumption continues to grow and this has significant second order effects on an area that has received significantly less study: datacenter cooling.

In addition to servers that perform work and an electrical distribution network to power the servers, every datacenter needs a cooling system to remove heat generated. These cooling systems are very large and costing, costing millions of dollars for large facilities [65] and although high efficiency cooling systems usually save money in the long term, the initial cost of more efficient systems is typically higher [11]. As a result, and despite recent improvements, most datacenter cooling systems are still not very efficient, consuming on average around 40% and sometimes up to 60% or more of total datacenter power [10].

Thermal time shifting (TTS) [116] proposes to reduce the cost of datacenter cooling systems by temporally decoupling work done in a datacenter from the load on the cooling system in that datacenter. TTS leverages the diurnal cycle created by user facing traffic (such as web search queries, social media interaction, etc.): during the day, when users in a datacenter's service region are awake and active, load is high and load is much lower during the late night and early morning when users are not active online. During the peak hours of the day, TTS stores excess heat in a phase change material (PCM) such as paraffin wax and releases it during the off hours to better balance the cooling load, thus enabling a smaller cooling system by overprovisioning.

However, TTS faces a severe limitation: in order to enable the phase change, the PCM must be placed to maximize the swing from high temperatures during the peak hours to low temperatures during the off hours. In the datacenter this has meant placing the PCM inside of the servers directly behind the CPU where the temperature gradient is highest, but as a result the quantity of wax (and thus quantity of heat that can be shifted) is severely limited [116].

In this chapter of the dissertation, we investigate how direct water cooling can be used to transport heat straight from the CPU and other high power components to wax at a distance, removing the need for wax to be physically located inside of servers. This not only lowers the upfront costs of direct liquid cooling, which is much more energy efficient than air cooling, but also enables thermal energy storage in servers with GPU accelerators and other designs without extra air space leveraged in the earlier chapters.

1.2 Decoupling Work and Heat

1.2.1 Thermal Time Shifting: Leveraging Phase Change Materials in Warehouse-Scale Computers

Thermal time shifting is illustrated in Figure 1.1. This figure presents a diurnal pattern with a peak utilization and heat output during the middle of the day (7 AM to 7 PM). If we were able to cap heat output during the peak hours and time shift the energy until we have excess thermal capacity in the off hours, we can maintain the same level of server utilization using a cheaper cooling system with a much smaller cooling capacity.

This PCM-enabled thermal time shifting allows us to significantly reduce capital expenses, as we can now provision the cooling infrastructure for a significantly lower peak demand. Prior work on power shifting using batteries [65, 43] demonstrates the ability to produce a flat power demand in the face of uneven diurnal power peaks.

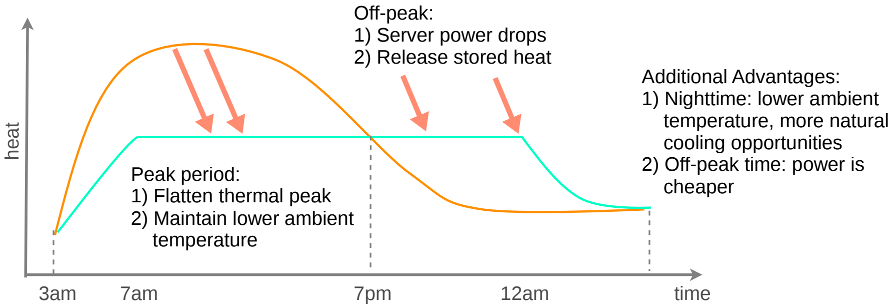


Figure 1.1: Thermal time shifting using PCM

However, the power for the cooling still peaks with the workload. This work allows the cooling power to also be flattened, placing a tighter cap on total datacenter power.

Alternatively, we can use PCM to pack more computational capacity into the warehouse of an existing datacenter with a given cooling infrastructure without adding cooling capacity– this better amortizes the fixed infrastructure costs of the entire datacenter. Furthermore, given a load pattern such as the one in Figure 1.1, the ability to shift cooling demands from peak hours to the night time would allow us to take advantage of lower electricity rates during the night, or even leverage free cooling in regions with low ambient temperatures [41, 32, 126, 43, 73].

Despite the numerous advantages of PCM-enabled thermal time shifting, a number of important research challenges needed to be addressed to fully exploit its advantages:

1. We need an adequate simulation methodology and infrastructure to study the PCM design space. To directly deploy PCM at a datacenter scale for design space exploration would be cost-prohibitive.

2. We need to investigate the trade-offs of various PCMs and identify the material that fits best in the datacenter environment. No prior work has studied PCM-enabled computation on this scale before, and selecting the correct PCM is critical to maximize impact while minimizing total cost of ownership (TCO).

3. We need to investigate suitable design strategies for integrating PCM in thou-

sands of servers. Modern commodity servers are designed with excess cooling and interior space to allow for many applications, but there are ways to leverage this reconfigurability to enhance PCM performance.

4. We need to quantify the potential cost savings of using PCM. Datacenter cooling systems are very expensive, and even a small reduction can save hundreds of thousands or millions of dollars.

1.2.2 Virtual Melting Temperature: Managing Server Load with Phase Change Materials

Thermal time shifting (TTS) [116] decouples work and heat by storing excess heat in a phase changing material (PCM) and removing that heat at a later time. TTS with PCM works by placing a quantity of PCM downwind from the CPU sockets in a rack mounted server. During the peak hours (mid-day through the evening) when users are online and load is high, the PCM melts absorbing heat to reduce the thermal output of the datacenter. Then during the off hours (late night and early morning) when most users are asleep, load is low, and extra cooling capacity is available, the PCM refreezes and the stored thermal energy is released.

A reduced peak cooling load has two major advantages: the datacenter can employ a smaller cooling system while still meeting the computational demands of peak load, or the same datacenter can run more and/or hotter servers under the same cooling budget. Both benefits can save hundreds of thousands of dollars per year or millions of dollars in capital expenses [116], however it is not a universal solution.

While TTS implemented with commercial grade paraffin wax can be both thermally effective and cost effective (approximately $1,000 per ton [116]), it has key limitations. Most of these limitations stem from the fact that the optimal melting temperature for a datacenter depends on many factors, from ambient temperature, to workload, to power and delivery limits. All of these can change from installation to installation, from season to season, or even from day to day. This is problematic because:

1. Commercial-grade paraffin can only be purchased within a limited range of melting temperatures, typically 40-60 °C, however if a melting temperature outside of this range is needed molecularly pure n-paraffin options cost in excess of $75,000 per ton.

2. Once installed, the wax melting temperature cannot be adjusted. On days when the load does not cause wax to melt, there is no flattening of the diurnal cooling load while on days when all the wax melts too soon, there is no reduction in peak temperature and cooling load.

3. The power and temperature profile of a workload often changes over the multi-year lifetime of a server. As the power profile changes, the ideal (or required) melting temperature can also change to necessitate new wax or leave the range of commercial wax melting temperatures entirely.

In all three cases, deploying wax in the servers provides little to no benefit and TTS is a passive system that cannot adapt.

In this dissertation, we propose a new adaptable technique called Virtual Melting Temperature (VMT) to handle workload power mixtures that TTS alone is unable to cool. VMT does so in such a way that it induces melting of the PCM (and thus heat redistribution) at load and average temperature levels that are (configurably) different than would happen with TTS, thus mimicking the operation of wax with a melting point that is different than the physical melting point of the deployed wax.

This is accomplished by rebalancing the load to raise temperatures in some of the servers above the PCM's melting temperature and storing energy in select servers, with the benefits of reduced cooling load and reduced power. Strategically employing

VMT enables fine-grained control of wax melting and cooling, allowing VMT to reduce the peak cooling load when TTS cannot.

1.2.3 Thermal Gradient Transportation: Liquid Enhanced Heat Transportation for Energy Storage with Phase Change Materials

In this dissertation, we propose Thermal Gradient Transportation (TGT), a method to move the large temperature difference desired by TTS outside of the server to enable a much greater quantity of wax, and thus much greater thermal energy storage. TGT works by leveraging server-level direct water cooling, a technique long proposed to cool high power density systems [29, 142]. Server-level water cooling offers significant power and energy efficiency benefits: total cooling power consumption has been demonstrated as low as 3.5% of total power [58], 11x less power than typical air cooled datacenters [10]. However despite the savings from reduced power costs, water cooled servers have not been widely adopted in no small part due to the expense of installing them [14, 94].

1.3 Summary of Contributions

1.3.1 Leveraging Phase Change Materials in Warehouse-Scale Computers

In this dissertation, we present the advantages of PCM on a datacenter scale.

1. We consider several PCMs for deployment in a datacenter, and select one for further investigation.

2. We then perform a set of experiments with PCM on a real server, and validate a simulator with these tests.

3. Using our validated simulator, we perform a scale out study of PCM on three

different server configurations to predict the impact of PCM deployed in a datacenter.

In an unconstrained datacenter, we find PCM enables a 12% reduction in peak cooling utilization or the deployment of 14.6% more servers under the same thermal budget. In a thermally constrained datacenter (e.g., more servers than the cooling system can cool), we find PCM can increase peak throughput by up to 69% while delaying the datacenter from reaching a thermal limit by over three hours.

1.3.2 Managing Server Load with Phase Change Materials

In this dissertation, we make the following contributions:

1. We introduce VMT, a method to manage the thermal properties of a PCM-enabled datacenter by controlling workload placement. We introduce and discuss three workload placement algorithms to enable VMT, and select the most promising two for further study.

2. We perform a scale out study of VMT with two algorithms, using a previously verified simulation methodology to execute a design space exploration of VMT on a cluster of 1,000 PCM-enabled servers. We examine two VMT algorithms in a cluster running five different workloads, each with unique thermal characteristics.

3. We quantify the impact of VMT at the cluster and datacenter levels, providing useful discussion of how best to use VMT in a datacenter and quantifying the potential benefits of VMT-enabled cooling oversubscription policies.

At the cluster level, we find that VMT can reduce the peak cooling load by 12.8% even when the average thermal output of the the cluster is too low for TTS. At the datacenter level VMT reduces the peak cooling load by up to 3.2 MW, allowing for up to 7,339 more servers under the same cooling budget or for the datacenter to operate at full capacity with a smaller cooling system saving $2.6 million in scenarios where TTS provides no measurable benefit.

1.3.3 Liquid Enhanced Heat Transportation for Energy Storage with Phase Change Materials

In this dissertation, we show how energy storage can enable a datacenter to deploy a significantly smaller cooling system that still meets the datacenter's peak cooling needs. Specifically, we make the following contributions:

1. Thermal Gradient Transportation (TGT): We introduce a new technique, TGT, to store thermal energy in a PCM at a distance, thus greatly increasing the quantity of thermal energy that can be stored.

2. Real world experiments: We conduct experiments on a real server, connecting a water cooled server with a reservoir of paraffin wax to gain insights into single-node performance and build a model for our scale out study.

3. Scale out simulation study: Using the knowledge from our real hardware tests, we construct a scale out simulation model to evaluate TGT techniques at the cluster scale for traditional datacenter workloads as well as new workloads such as neural networks and IPAs running on a GPU.

In our scale out study, we show that TGT can reduce the peak cooling load of a datacenter up to 1.97x more than prior work in this area. We also show that TGT can reduce the peak cooling load up to 20% for datacenters with GPUs, an increasingly common datacenter topology that TTS cannot service.

CHAPTER II

Background and Related Work

In this Chapter, we consider related works to the subject matter of this dissertation. Prior work has investigated a wide range of chip and datacenter cooling methods, including PCM for computational sprinting, batteries for power overprovisioning, chilled water tanks for emergency cooling, and a variety of load balancing schedules for workload allocation.

2.1 Thermal Time Shifting Related Work

The thermal energy storage potential of paraffin has previously been examined on a small, single-chip scale for computational sprinting in [106, 105, 104] with promising results. While that work uses PCM in small quantities to reshape the load without impacting thermals, we take the opposite approach, using PCM to reshape the thermal profile with minimal change to the load. Additionally, we study PCM deployment on a datacenter scale to consider thermal time shifting over periods lasting several hours, compared to seconds or fractions of seconds in the computational sprinting approach.

When considering PCM deployment across thousands of servers, we find that some of the techniques used in computational sprinting, such as the application of expensive n-paraffin wax, are cost prohibitive on our scale. We also observe that while

Raghavan, et al. [105] studied a metal mesh embedded in paraffin to improve thermal conductivity, this potentially expensive measure is not necessary when melting paraffin over the course of several hours and the melting speed can be sufficiently improved by placing the paraffin in multiple containers to maximize surface area.

To reduce power infrastructure capital expenses in a datacenter, many authors have investigated UPS batteries to make up the difference when load exceeds the power distribution system power [65, 43, 44, 126, 45]. Our implementation of PCM is complementary to UPS power oversubscription, and also inherently more efficient than a UPS-only solution to oversubscribe cooling system power because while a UPS-only system must first stores electricity in batteries then uses the batteries to power the cooling system, PCM can store thermal energy directly and with negligible loss.

Chilled water tanks for thermal energy storage is an active cooling solution considered by several authors [141, 140, 108, 39] to leverage the sensible heat of water during peak demand or emergencies. Our PCM approach is a completely passive thermal solution that is complementary to any active cooling solution (whether it be forced air HVAC, chilled water, etc.), because our passive technique will always reduce the peak demand placed on the active solution.

Comparing, in particular, to the chilled-water, active cooling solution of Zheng, et al. [141], PCM-enabled thermal time shifting also has the advantage of no software, power or infrastructure overhead to control and contain water that TE-Shave requires. PCM requires no additional floor space or infrastructure because it is deployed inside of the server and draws no additional power, unlike chilled water tanks that must be deployed outdoors and cooled regularly, whether used or not, to compensate for environmental losses.

2.2 Virtual Melting Temperature Related Work

A number of works have proposed to leverage the thermal energy storage capacity of PCMs in the computing domain. Computational sprinting [106, 105, 104, 107, 112] proposes to place a small amount of PCM in contact with the CPU to enable brief ”sprints” of fast operation that exceed the safe sustained power levels, but is less useful for datacenters where increased activity lasts for multiple hours at a time. TTS proposes to use wax [116, 117] to passively reshape the thermal profile, but cannot be widely deployed or adapted for many workload mixtures. Other work related to TTS has proposed to use a PCM for emergency overprovisioning [57], and to use an adversarial approach to mitigate conflict for shared resources in datacenters with limited power and cooling utilities [31].

VMT uses a similar approach to TTS, but propose to accomplish the thermal reshaping using both latent energy storage in wax as well as thermal aware job placement to maximize stored energy. In contrast to TTS, which places wax in servers and passively waits for conditions to be amenable to melt wax, TTS actively places jobs to maximize thermal storage and peak cooling load reduction.

Prior work on load balancing [68, 6, 71, 98, 26, 19] used workload placement to improve performance, energy consumption, and/or cooling efficiency. VMT implements workload placement to unbalance power consumption and thus temperatures at the cluster level, however for many workloads these load balancing techniques may still be useful to coordinate jobs within the hot and cold groups and to distribute hot and cold servers spatially to balance load across multiple cooling systems.

Similarly, job consolidation has been considered in prior work to reduce power consumption [127, 125, 81, 89, 9], however this approach requires extra server capacity that may not be available during the peak hours. Job consolidation can be used alongside VMT during the off hours, as long as jobs are not consolidated to a level where they melt wax before the peak hours.

Prior work in thermal aware job placement leverages spacial aware of hot and cold spots in the datacenter to increase efficiency [87]. Tang et al. manage the inlet temperature distribution and place jobs accordingly for maximum power efficiency [122], and Xu et al. propose to relocate jobs between geographically dispersed datacenters to maximize cooling efficiency [134, 135, 42, 74]. These are parallel or compatible work with potential benefits when used alongside VMT.

Power over subscription is another area where prior work proposed to use batteries to manage peak hours and/or power emergencies [65, 43, 44, 126, 45]. Most of these leverage uninterruptible power supply (UPS) batteries already present in datacenters, and their techniques complement VMT well as hot jobs both draw the most power and release the most heat.

Prior work for thermal overprovisioning proposed to use a variety of sensible heat storage mechanisms. Several works propose to use water tanks for thermal energy storage as the thermal density of water is much greater than air, and the water may be chilled during off hours to prepare for peak hour load [141, 140, 108, 39]. VMT is not strictly applicable to techniques that rely on sensible energy storage, rather than latent energy storage, but these techniques are compatible with VMT.

2.3 Thermal Gradient Transfer Related Work

The benefits of PCMs such as paraffin wax for thermal energy storage have been considered for a number of applications related to electronics. Computational sprinting [106, 105, 104, 112] proposed to use a very limited quantity of PCM to temporarily exceed a chip's cooling capabilities.

More recent work on TTS [116, 117, 118] has demonstrated the ability for a whole datacenter to temporarily exceed its cooling ability through the use of commercial paraffin wax deployed inside of servers. This technique is particularly limited, as the space inside of servers is both small and becoming smaller as accelerators such as GPUs are added to datacenters. In this work, comparing the same servers and workloads using TGT, we demonstrate a nearly 2x better reduction in the peak cooling load compared to TTS [116] and the ability to handle GPU-equipped servers too.

Extensive prior work has shown the energy efficiency benefits water cooling at the chip level in datacenters [25, 30, 29, 37, 46, 91, 142]. We build upon this body of work, using their recommendations when designing our test system and scale out study.

TGT is particularly amenable to warm water cooling proposed by Zimmermann et al. [142], as largely temperature differences can melt wax more efficiently, however as most of the energy is released at a lower temperature than it was initially captured due to the phase change, the energy reuse scenarios proposed by Zimmermann et al. are less efficient.

Perhaps the most direct comparison to TGT are chilled water and water ice tanks for thermal energy storage [39, 108, 140, 141]. Zheng et al. [141] estimate the cost of chilled water tanks to be \$140 per kWh of storage, while ice takes cost around \$72 per kWh with a coefficient of performance (COP) of 5 decreasing to \$40 and \$20, respectively, with a COP of 1.4. Commercial paraffin wax on the other hand, if available in the right range of melting temperatures, costs approximately \$18 per kWh of energy storage for the wax alone [116] making it very competitive. Furthermore, requires no additional electricity to pre-cool or pre-freeze the reservoir as water and ice do, estimated at 1-5% of total energy storage capacity per day [140], even if the energy storage capacity is not needed that particular day.

CHAPTER III

Thermal Time Shifting: Leveraging Phase Change Materials to Reduce Cooling Costs in Warehouse-Scale Computers

Datacenters, or warehouse scale computers, are rapidly increasing in size and power consumption. However, this growth comes at the cost of an increasing thermal load that must be removed to prevent overheating and server failure.

In this chapter, we propose to use phase changing materials (PCM) to shape the thermal load of a datacenter, absorbing and releasing heat when it is advantageous to do so. We present and validate a methodology to study the impact of PCM on a datacenter, and evaluate two important opportunities for cost savings. We find that in a datacenter with full cooling system subscription, PCM can reduce the necessary cooling system size by up to 12% without impacting peak throughput, or increase the number of servers by up to 14.6% without increasing the cooling load. In a thermally constrained setting, PCM can increase peak throughput up to 69% while delaying the onset of thermal limits by over 3 hours.

3.1 Integrating PCM in WSCs

To enable thermal time shifting, this work proposes to place a quantity of PCM inside of each server, as shown in Figure 3.1. When the temperature rises above the PCM's "melting threshold," the PCM will melt and absorb energy until all of the PCM is liquefied. Later, when the temperature drops below the threshold, the PCM will re-solidify and release energy until the PCM is solid again.

Placing PCM directly in contact with a the heat spreader of a single processor is beneficial for computational sprinting and other short-term cooling applications [106, 105, 104, 128], but we require a much greater quantity of PCM in a datacenter-sized cooling system with a 24 hour thermal cycle [64, 82]. Placing PCM in the server downwind of the processor sockets enables more PCM and still leverages the large temperature difference between idle and loaded levels. Alternatives such as placing PCM outside of the datacenter or adding a layer insulation in the walls and ceiling (reducing the ability of heat to escape when ambient conditions are favorable) require a infrastructure to move heat to the PCM and suffer a lower temperature differential due to heat loss and mixing over the travel distance.

Thus, the advantages of our PCM-enabled system are simple: the PCM is entirely passive. There is no power, software or floor space overhead to add PCM to a datacenter, and minimum labor is needed after installation to achieve the potential benefits.

3.1.1 Investigation of PCM Characteristics

A variety of PCM materials are available, but not all are suitable for the scale or operating conditions of a datacenter. To evaluate the available PCMs, several key properties need to be taken into account including the *melting temperature*, *energy density*, *stability*, and *cost*.

Melting temperature is critical as it determines when our PCM absorbs and re-

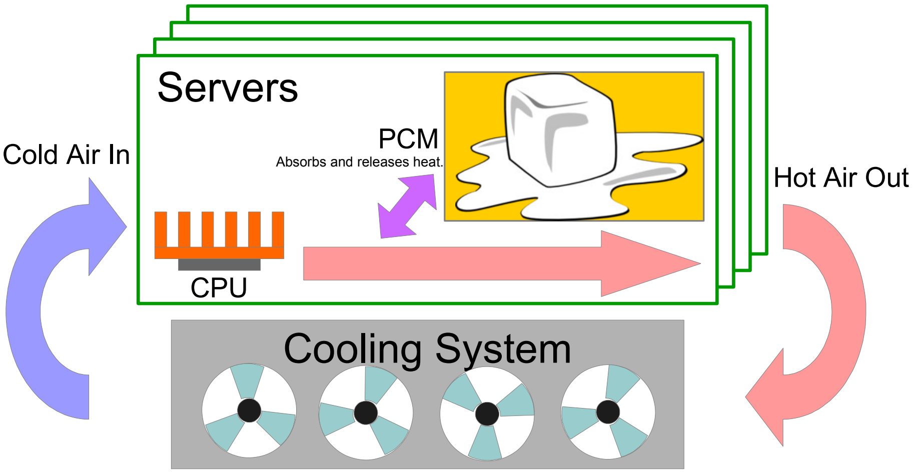


Figure 3.1: Integrating PCM in a WSC

Table 3.1: Properties of common solid-liquid PCMs.

PCM	**Melting Temp. (°C)**	**Heat of Fusion (J/g)**	**Density (g/ml)**	**Material Stability**	**Electrical Conductivity**	**Corrosivity**
Salt Hydates	25-70	240-250	1.5-2	Poor	High	Yes
Metal Alloys	>300	High	High	Poor	High	No
Fatty Acids	16-75	150-220	0.8-1	Unknown	Unknown	Yes
n-Paraffins	6-65	230-250	0.7-0.8	Excellent	Very Low	No
Commercial Paraffins	40-60	200	0.7-0.8	Very Good	Very Low	No

leases significant amounts of heat. In a datacenter, we want the melting temperature to fall between the peak and minimum load temperatures. Although the best melting temperature must be determined based upon ambient temperatures where the PCM is located, among other factors, the appropriate range is usually between 30 to 60 °C.

The energy density of the PCM defines how much energy it can store and is proportional to the heat of fusion (melting energy) and density of the PCM in both solid and liquid phases. A high energy density is desirable to maximize energy storage using the small amount of space available inside of the server. We also need to consider the corrosivity and electrical conductivity to contain a PCM and minimize damage in case it leaks out of the enclosure.

PCM Comparison - Of the phase transformations presented by Pielichowska, et al. [96], we find solid-liquid transformations to be promising for datacenter deployment right now. Liquid-gas and solid-gas have a much lower density in the gaseous state that reduces the energy storage density, and make PCM containment much more difficult. Solid-solid PCMs are attractive with a potentially high heat of fusion, low thermal expansion, and low risk of spillage; however, the solid-solid PCMs considered for energy storage by Pielichowska, et al.[96] undergo the phase change outside of acceptable datacenter temperatures, exhibit poor material stability in as few as 100 cycles of melting and resolidifying, possess a low energy density, or would be cost prohibitive in a datacenter at this time.

In Table 3.1 we compare five types of solid-liquid PCMs. Of the five, salt hydrates and metal alloys both have a high energy density but poor stability over repeated phase changes. The typical melting temperature of the metal alloys is much too high for datacenter use, and salt hydrates and fatty acids are both corrosive [96, 113, 47, 53].

We find that paraffin waxes are the most promising of the PCMs available right now. Paraffins typically have a low density but a good heat of fusion, are non-corrosive and don't conduct electricity. Paraffin is also highly stable, with negligible deviation from the initial heat of fusion after more than 1,000 melting cycles [96]. Paraffin wax is typically available in two forms: molecular pure n-paraffin (eicosane, tridecane, tetradecane, etc.) and commercial grade paraffin. Eicosane, previously studied for computational sprinting [105], has promising material properties including a high heat of fusion (247 J/g) and an appropriate melting temperature of 36.6 °C. However, we conclude that it is cost prohibitive to deploy at large volume in a datacenter. Sigma-Aldritch® quoted the mass production price of eicosane n-paraffin at $75,000 per ton. Even in a relatively small datacenter the cost of equipping every server with eicosane would be over a million dollars in wax costs alone.

Commercial grade paraffin is a less refined wax consisting of a mixture of paraffin molecules. It has a slightly lower heat of fusion (200 J/g), but is much less expensive than eicosane. As of August 2014, quotes for bulk commercial grade paraffin with melting temperatures ranging between 40 and 60 °C were typically \$1,000 to \$2,000 per ton on Alibaba.com® [90]: 50x cheaper for 20% lower energy per gram compared to eicosane, which we deem as a reasonable trade-off.

3.2 Modeling and Model Validation

The lack of experimental infrastructure and simulation methodology is a major challenge for conducting an investigation on PCM-enabled thermal time shifting. In this section, we introduce our infrastructure to simulate paraffin wax inside of a server. We integrate PCM modeling within a computational fluid dynamics (CFD) simulation for server layout using ANSYS® Icepak. To validate our PCM modeling, we rely on a series of measurements taken using a small quantity of paraffin inside of a real server and compare our model against those real server results. Modeling heat and airflow at this level is critical for two reasons. First, we need to accurately model heat exchange between the components, the air, and the wax. Second, our wax enclosures disrupt the airflow of the server and can have negative effect on heat removal if placed incorrectly.

3.2.1 Test System Configuration

We perform extensive benchmarking of a Lenovo® RD330 server to accurately model the server in Icepak and validate the model of PCM in Icepak. Our RD330 (Figure 3.2) is a 1U server with two sockets, each populated by a 6-core Intel® Sandy Bridge Xeon® CPU clocked at 2.4 GHz with Intel TurboBoost turned off. The server has 144 GB of RAM in 10 DDR3 DIMM sticks, a 1 TB 2.5" hard drive, and a single power supply unit rated at 80% efficiency idle and 90% efficiency under load. The server has six 17W fans, and runs Ubuntu® 12.04 LTS server edition. For the PCM, we purchased commercial grade Paraffin wax from Amazon.com® and measured the

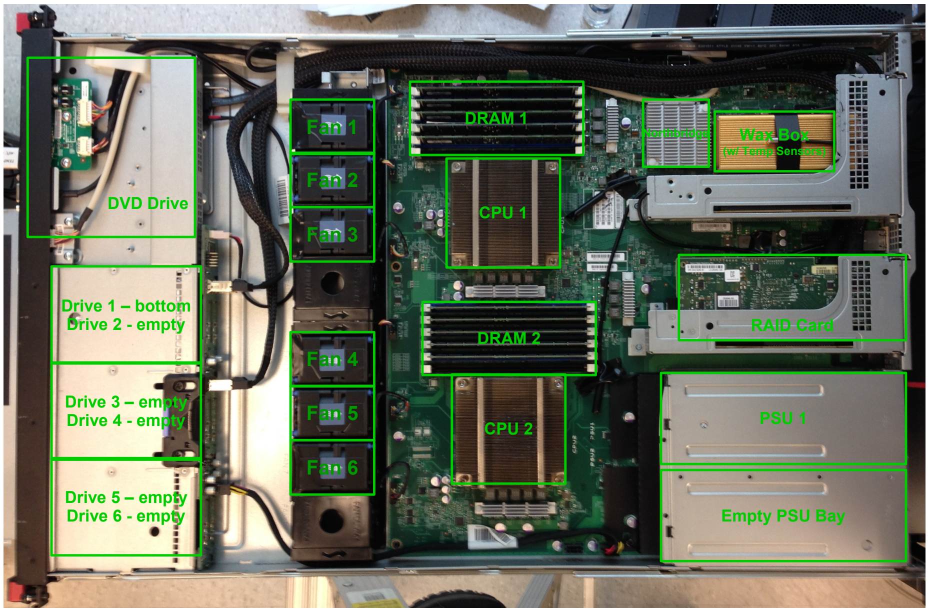


Figure 3.2: RD330 Server with major components labeled

melting temperature at 39 °C.

3.2.2 Experimental Methodology

Accurate measurement is critical for creating an accurate model. To acquire accurate ground-truth measurement, we design several experiments and use a number of tools to measure server power, temperatures at various points, and PCM's impact on temperatures. To measure total system power at the wall, we use a Watts Up Pro® USB® power meter. We measure internal temperatures in the server with a set of TEMPer1 USB temperature sensors. We also use the Intel Power Governor tool to measure the socket, core, and DRAM power in real time.

To measure the effect of a small amount of PCM in the system, we fill a sealed aluminum container with 90 ml (70 grams) of paraffin wax and leave an extra 10 ml of airspace to account for paraffin expansion and contraction. The aluminum box was placed in the rear of the server, downwind of CPU 1 and three TEMPer1 sensors were inserted to record temperatures near the box and server outlet. We also conducted a series of trials with the same aluminum box empty of wax (filled only with air) in the same location in the server as a placebo to further validate our model as well as separate the thermal effects of the PCM and airflow impact of the box on the server.

We perform multiple trials with and without wax where we subject the server to 60 minutes of idle time, followed by 12 hours under heavy load (one instance of SPEC® h264 per logical thread) to heat the server up until temperatures stabilize, and then 12 hours at idle again to measure the server cooling down.

We observe that the total system power doubles from 90 W idle to 185 W fully loaded. CPU power increased by 7.7x from 6 W idle to 46 W per socket under load. Package temperature, as reported by the chip's internal sensors, rose from 42 °C idle to 76 °C under load.

3.2.3 Modeling Server and PCM in Icepak

To simulate the effects of wax in our server, we construct a model of our server in the computational fluid dynamic simulator ANSYS Icepak. From front to rear, we model the hard drive, DVD drive and front panel as a pair of block heat sources. The fans are modeled as a time-based step function between the idle and loaded speeds. Each DRAM module is modeled independently, but memory accesses are approximated as uniform to evenly distribute power across all of the modules. The PSU is modeled in the rear of the server enclosure, and all other heat sources (motherboard, LEDs, I/O, etc.) are lumped together with the CPU sockets.

3.2.4 Model Validation

In Figure 3.3 (a) and (b), we highlight the heating up and cooling down traces of average temperatures near the server outlet. We see a strong correlation between the real measurements and Icepak simulation measurements for the trace, and observe the wax reduces temperatures for two hours while the wax melts (absorbing heat), and afterwards increases temperatures for two hours while the wax freezes again (releasing

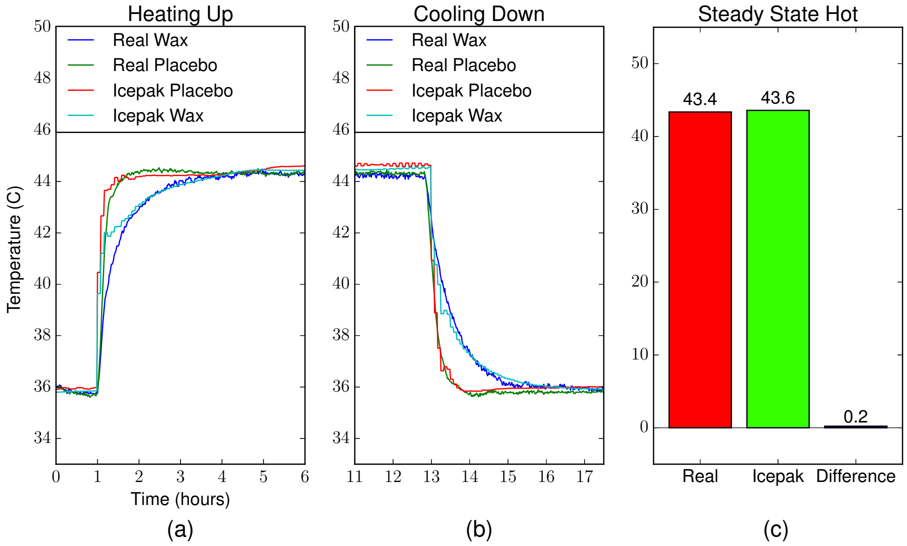


Figure 3.3: Model validation. Transient traces while heating up (a) and cooling off (b), and steady state while hot (c) comparison of temperatures around the wax in the real server and our Icepak model.

heat).

In Figure 3.3 (c), we compare steady state temperatures measured from USB sensors on the real server to temperatures measured from the same locations on the Icepak model while both were fully loaded (between hours 6 and 12). We observe a mean difference of 0.22 °C between the real measurements and Icepak simulation measurements on the loaded server.

3.3 Methodology

In this section, we introduce our methodology and candidate machines for a scale out study on PCM datacenters. We examine three homogeneous datacenters each provisioned with a different type of machine, shown in Figure 3.4. First, we consider a deployment of low power servers using the same 1U commodity server validated in Section 3.2. Second, we consider a high-throughput deployment consisting of 2U commodity servers similar to the Sun® Server X4470 with four 8-core Intel Xeon CPUs, and last we consider a high-density deployment of Microsoft® Open Compute®

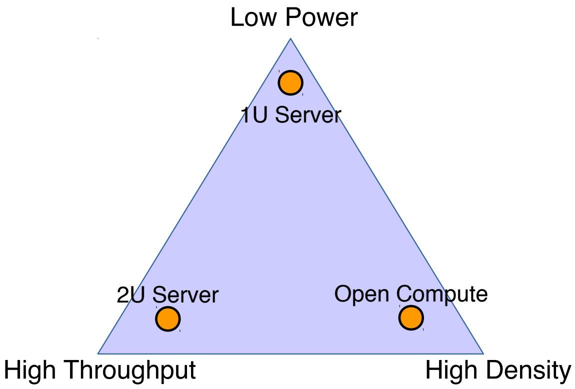


Figure 3.4: Three servers considered in the scale out study, each targeting a different end of the spectrum.

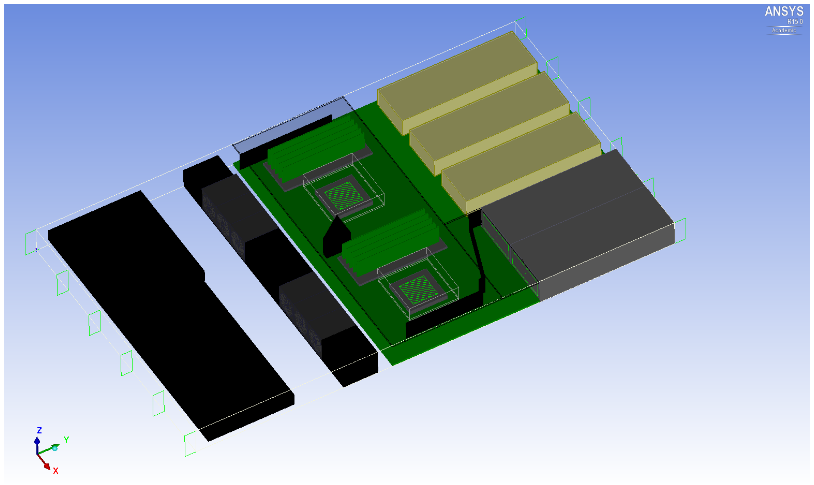
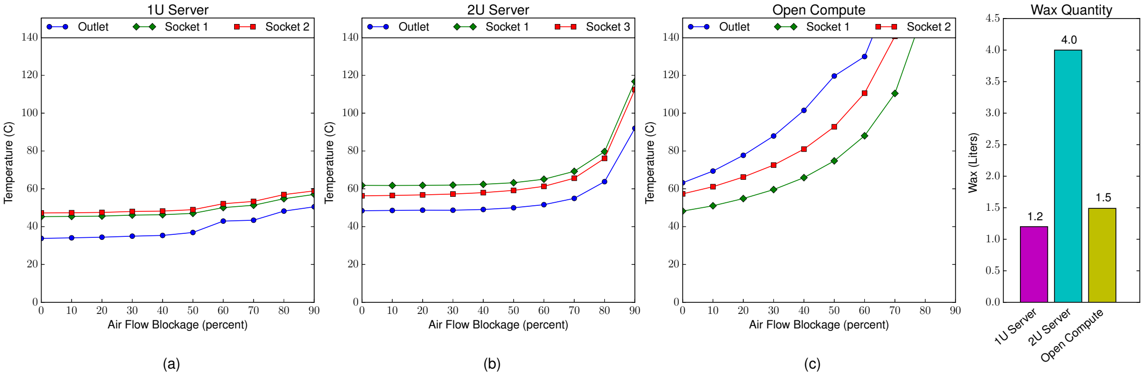


Figure 3.5: 1U low power server modeled in Icepak, with 1.2 liters of wax (gold).

blades with two 6-core Xeon CPUs each. We evaluate each datacenter using real workload traces from Google®, and present the results in Section 5.4.

3.3.1 Servers

1U Commodity Server - The Lenovo RD330 we validated is a low power, 1U commodity server with an estimated cost of $2,000 for our configuration. To increase available space inside of the server, we replace the PCIe® risers and unnecessary RAID card (there is only one HDD in the server) with PCM. We conduct a series of experiments in Icepak blocking airflow with a uniform grille downwind of the CPU



(a)

(b)

(c)

Figure 3.6: Server temperatures as airflow through each server is blocked. CPU temperatures in the 1U server (a) rise less than 2 °C below 50 %, and begin to rise quicker thereafter. Temperatures in the 2U server (b) are stable below 60 % quickly rise to unsafe rise to unsafe levels above 70 % obstructed airflow. Temperatures in the Open Compute server (c) rise to unsafe levels as soon as almost any airflow is obstructed.

heat sinks, shown in Figure 3.6 (a). In these experiments, we maintain a constant frequency and power consumption to maintain parity across configurations. From 0% (no air blocked) up to 90% of air flow blocked, we observe a 14 °C increase in air temperatures at the outlet, and at no time do the CPU temperatures reach unsafe levels.

We model the addition of 1.2 liters of wax inside of aluminum boxes as shown in Figure 3.5 blocking 70% of airflow downwind of the CPUs. We could have increased the amount of wax (blocking further airflow), but found it was better to leave sufficient space between the boxes and edges of the server, thus maximizing surface area in contact with moving air in order to speed melting.

2U Commodity Server - The Sun X4470 is a high-throughput commodity server with up to four Intel E7-4800 processors. We model the server with four 8-core processors and 32 GB of RAM in two DDR3 DIMM packages per socket. In a 2U form factor we can fit up to 20 servers per rack and we estimate peak server power at 500 W per server after the PSU. Based on suggested retail prices, we estimate total cost to be $7,000 per server.

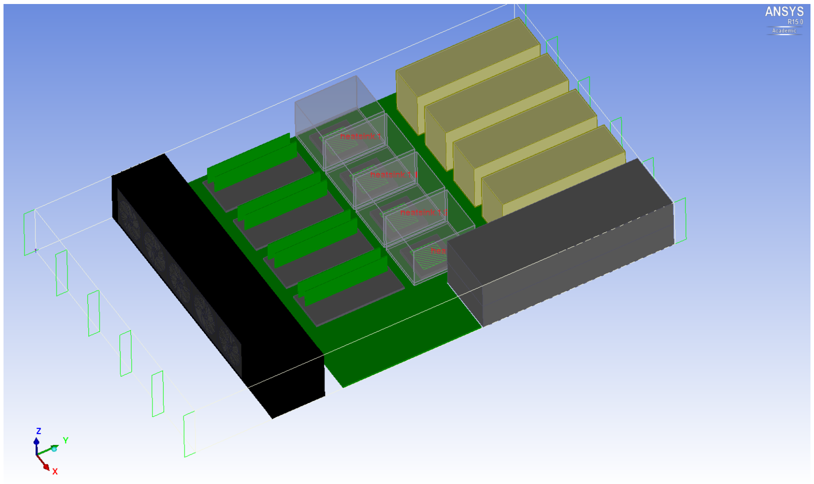


Figure 3.7: 2U high-throughput server with four CPU sockets, modeled in Icepak with 4 liters of wax (gold).

We model the 2U commodity server in Icepak in Figure 3.7. From front (left) to rear (right), air is pulled in through a series of fans, passes over the RAM, through the CPU heat sinks, past vacant PCIe card slots and out the rear of the server. The PCIe slots are present in the commodity server, but in our configuration they are not utilized so we leverage the free airspace to add wax into the server.

In Figure 3.6 (b) we plot temperature in the server as air is blocked by a uniform grille. When less than 50% of the air flow through our 2U commodity server is blocked we observe an almost negligible impact on outlet and CPU temperatures while at above 50% the temperature increases exponentially.

To add wax to our server without dangerously raising temperatures, we choose to add 4 one liter aluminum boxes filled with wax (colored gold in Figure 3.7) and maintain sufficient unfilled space to account for thermal expansion. These boxes block 69% of airflow through the server, increasing the outlet and CPU temperatures (with empty boxes) by less than 6 °C.

Open Compute Blade Server - The published production Microsoft Open Compute server is a 1U, sub-half-width blade with two sockets each containing a 6-core Intel Xeon processor and 64 GB of RAM in two DDR3 DIMM packages per

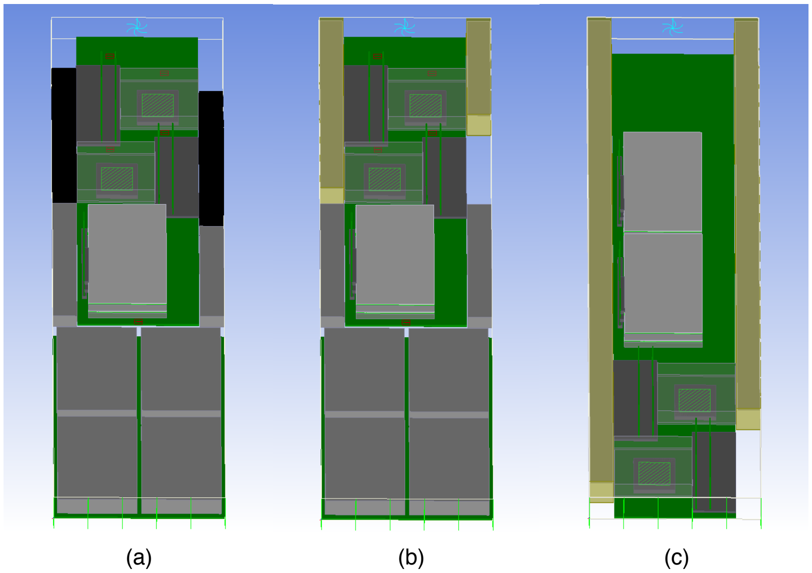


Figure 3.8: Microsoft Open Compute server, modeled in Icepak from prior work [102] (a), with air flow inhibitors replaced with wax containers (b), and Open Compute reconfigured with 1.5 liters of wax. (c).

socket. Two solid state drives (SSDs) connected via PCIe provide primary data storage, while four 3.5” 2 TB hard drives are present for redundancy. Each quarter-height Open Compute chassis fits 24 blades and has a total of six fans that draw air out the rear of the servers at less than 200 linear feet per minute at the rear of the blade. The peak power consumption for any single blade is limited to 300 W before the PSU, and the air temperature behind Socket 2 was measured at 68 °C. We model the idle power at be 100 W and active power at no more than 300 W. Based on current (August 2014) market trends we estimate cost per blade to be $4,000 [102].

We model the Open Compute server in Icepak based upon published dimensions and specifications for the form factor, CPUs, hard drives, and motherboard [102, 114, 115, 70], and estimate dimensions and power ratings for the SSDs based on the Fusion-io enterprise product line [35]. As with the commodity servers, additional heat sources in the Open Compute blade are lumped together with the CPUs. We do not model the volume or power requirements of the Catapult FPGA board [102].

In Figure 3.8, we present three Icepak models of the Open Compute configurations.

Figure 3.8 (a) shows the production Open Compute configuration. We observe that even in a densely populated server like Open Compute, there is still useful space available where we can add wax without impacting airflow: along the sides of either CPU, plastic inserts (black) block air from traveling around the CPU heat sinks. In Figure 3.8 (b), we replace these blocks with 0.5 liters of wax in sealed aluminum containers.

The temperature gradient necessary to melt and cool wax in the server is created primarily by the CPUs, so wax is only useful if placed behind the CPUs. To increase the wax capacity, we consider an alternate configuration where we switch the CPU location with that of the SSDs to increase the downwind volume. We then consider a possible future Open Compute design where the redundant HDDs have been replaced with a second set of SSDs to achieve 1.5 liters of wax as shown in Figure 3.8 (c) without increasing the air flow blockage versus the production blade.

In Figure 3.6 (c), we study blocking additional airflow to add more than 1.5 liters of wax. (The outlet temperature is measured higher than CPU temperature due to the thermal output of the four enterprise class PCIe SSDs, which can exceed 85 °C even with proper cooling [137].) We observe that the already high outlet temperature and CPU temperatures increase exponentially as soon as any blockage is placed in the Open Compute blade, outweighing the benefits that any more wax would add.

3.3.2 Google Workload

We use a two day workload trace from Google [65, 123] to evaluate the effects of wax on our three datacenter server configurations. The workload we consider has three different job types: Web Search, Social Networking (Orkut$^{®}$) and MapReduce from November 17th through November 18th, 2010. This data was acquired as described by Kontorinis, et al. [65], and normalized for a 50% average load and 95% peak load for a cluster of 1008 servers of each configuration. After 2011, Google changed the

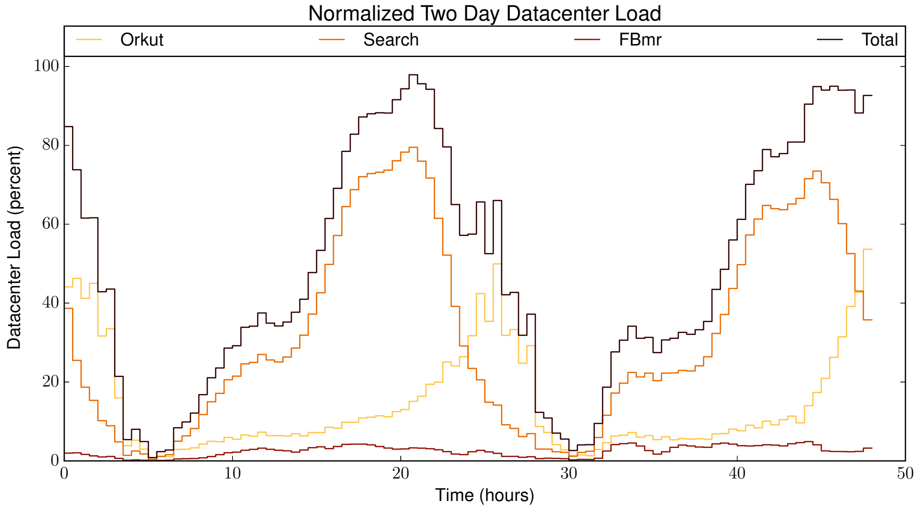


Figure 3.9: Two day datacenter workload trace from Google, normalized to peak throughput.

$$\begin{aligned} TCO \; = \; & (FacilitySpaceCapEx + UPSCapEx + PowerInfraCapEx \\ & + CoolingInfraCapEx + RestCapEx) + DCInterest + \\ & (ServerCapEx + WaxCapEx) + ServerInterest + (DatacenterOpEx \\ & + ServerEnergyOpEx + ServerPowerOpEx + CoolingEnergyOpEx \\ & + RestOpex) \end{aligned} \tag{3.1}$$

format of its transparency report so newer data is unavailable.

To model traffic and datacenter throughput, we use DCSim, a traffic-based simulator from prior work [65]. DCSim is an event-based simulator that models job arrival, load balancing, and work completion for the input job distribution traces at the server, rack, and cluster levels, then extrapolates the cluster model out for the whole datacenter. We use a round robin load balancing scheme, and extend DCSim to model thermal time shifting with PCM using wax melting characteristics derived from extensive Icepak simulations of each server.

Table 3.2: Parameters used to model TCO. (Dollars per watt refers to dollars per watt of datacenter critical power.)

Description	**TCO/month**	**Unit**
FacilitySpaceCapEx	1.29	$/sq. ft.
UPSCapEx	0.13	$/server
PowerInfraCapEx	15.9–16.2	$/kWatt
CoolingInfraCapEx	7.0	$/kWatt
RestCapEx	19.4–21.0	$/kWatt
DCInterest	31.8–36.3	$/kWatt
ServerCapEx	42–146	$/server
WaxCapEx	0.06–0.10	$/server
ServerInterest	11.00–38.50	$/server
DatacenterOpEx	20.7–20.9	$/kWatt
ServerEnergyOpEx	19.2–24.9	$/kWatt
ServerPowerOpEx	12.0	$/KWatt
CoolingEnergyOpEx	18.4	$/kWatt
RestOpEx	5.7–6.6	$/kWatt

3.3.3 TCO Modeling

We base our total cost of ownership (TCO) after Kontorinis, et al., modifying the model for our datacenter and server configurations, and add the interest calculation from Barroso, et al. (Table 3.2 and Equation 3.1) [65, 14]. To calculate the total savings from PCM, we consider the TCO without wax and subtract the TCO with wax for a single cluster of 1008 servers and extrapolate out to the size of the datacenter.

To best evaluate the TCO savings enabled by PCM, we consider the cooling infrastructure and the electricity cost of the cooling system separately from the datacenter operating expenditure (DatacenterOpEx). These two terms are important to our evaluation because they isolate the overall efficiency of the thermal-control system (including CRAC, cooling tower, and the PCM addition). We assume a linear relationship between the cost of cooling infrastructure and the peak cooling load the cooling system can handle. The electricity cost OpEx of the cooling system represents the average efficiency of removing heat. In addition, we also include the cost of adding the wax and the wax containers into the server capital expenditure (ServerCapEx), although the WaxCapEx is almost negligible representing less than 0.1% of the ServerCapEx.

To calculate the TCO for each server configuration, we consider three datacenters each with a critical power of 10 MW, the first filled with 55 clusters of 1U low power servers, the second with 19 clusters of 2U high throughput servers and the third with 29 clusters of Open Compute blades. We assume a peak electricity cost of $0.13 per kWh and an off-peak electricity cost of $0.08 per kWh [41].

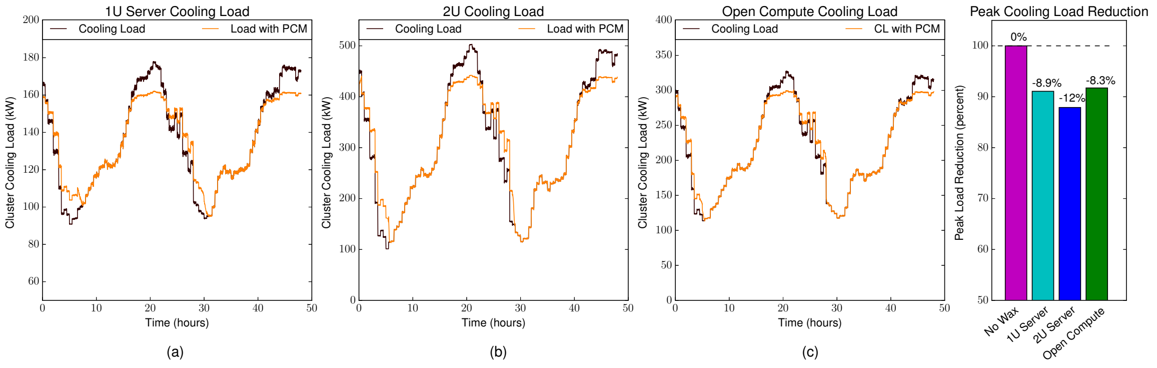
3.4 Evaluation

In Section 3.2, we validated Icepak to simulate PCM in a server, and in Section 5.3, we described our servers and workload for a scale out study of PCM. In this section, we consider two potential use cases for PCM to reduce cooling load and increase throughput.

First, in Section 3.4.1 we consider a datacenter with a fully subscribed cooling system and evaluate how PCM can reduce the peak cooling load. This translates to a smaller, less costly cooling system or alternatively providing cooling support for more servers with the same cooling system. Next, in Section 3.4.2 we consider an oversubscribed datacenter and show how PCM can increase the datacenter throughput without surpassing the datacenter thermal threshold.

3.4.1 PCM to Reduce Cooling Load

We first consider a datacenter with a fully subscribed cooling system that can remove the peak cooling load indefinitely. The cooling load of a datacenter is the power that must be removed to maintain a constant temperature [17, 92], and allows a direct comparison between different server, temperature, and datacenter configurations. In Figure 3.10 (a-c), we plot the peak cluster cooling load for a cluster of 1008



(a)

(b)

(c)

Figure 3.10: Cooling load per cluster over a two day Google trace in a datacenter with a fully subscribed cooling system. PCM reduces peak cooling load by 8.9 % in a cluster of low power 1U servers (a), 12 % in a cluster of 2U high throughput commodity servers (b), and by 8.3 % in a cluster of high density Open Compute servers (c).

of each test server without and with wax.

In this model, we assume all of the wax has a conservative heat of fusion of 200 J/g, and selected the melting temperature to minimize cooling load. The range of melting temperature available in commercial grade paraffin allows us to select one with an optimal melting threshold to reduce the peak cooling load of each cluster, and the best melting temperature is determined on the shape and length of the load trace: for the Google trace, we find that the best wax typically begins to melt when a server exceeds 75% load and melts quickly thereafter.

As shown, we achieve an 8.3% reduction in peak cooling in the Open Compute cluster, up to an 8.9% reduction in the cluster of 1U servers and 12% in the cluster of 2U servers as the wax absorbs heat and melts.

When the server utilization and temperatures fall below the melting threshold, we observe a period of time with increased cooling load higher than the placebo server while the wax cools off, lasting between six and nine hours. As the cooling system is operating below peak capacity during these times, there is sufficient cooling capacity to completely resolidify before the end of a 24 hour cycle.

With the peak cooling load safely reduced, we can then either decrease the size of the cooling system without sacrificing throughput, or add servers and increase critical power of the datacenter without increasing the size of the cooling system.

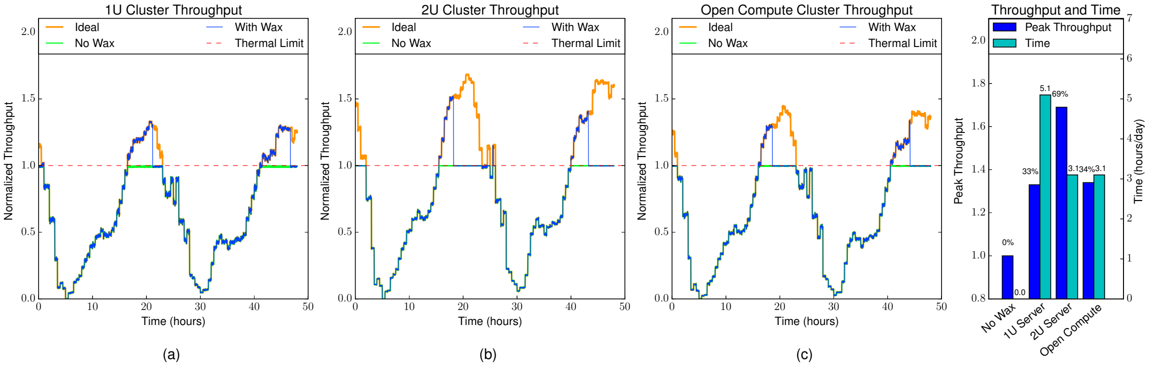
In a 10 MW datacenter, PCM allows us to install an 8.3% smaller cooling system in a high density Open Compute datacenter, an 8.9% smaller system with 1U low power servers, and a 12% smaller system with 2U high throughput servers. This translates to estimated cost savings of $174,000, $187,000, and $254,000 per year, respectively, on the cooling system and cooling power infrastructure. Here we observe that peak load reduction and savings correlate to the quantity of wax: the more wax that is added to a server, the greater the potential savings.

Alternatively, if instead of installing a smaller cooling system we use the excess cooling capacity enabled by PCM to install more servers, we can add 2,770 (8.9%) Open Compute blades, 4,940 (9.8%) more 1U low power servers or 2,920 (14.6%) more 2U high throughput servers to a 10 MW datacenter without exceeding the peak cooling load of the existing cooling system.

We evaluate the TCO savings created by oversubscribing the cooling system in a retrofit scenario: the old servers in a 10 MW datacenter have reached the end of their 4 year lifespan but the cooling system still has 6 years of useful lifespan remaining [65]. By adding PCM to a new deployment 1U, Open Compute, or 2U servers with an oversubscribed cooling system, we save an estimated $3.0 million, $3.1 million, and $3.2 million per year, respectively, over the cost of a new cooling system to achieve the same throughput.

3.4.2 PCM to Increase Throughput

In this section, we consider an oversubscribed datacenter where the cooling system is significantly smaller than the thermal output of the datacenter with all servers active. Such circumstances can arise as old servers are replaced with new denser



(a)

(b)

(c)

Figure 3.11: Google workload throughput normalized to peak throughput in a thermally constrained datacenter. PCM increases peak throughput by 33 % over 5.1 hours in the 1U server (a), 69 % over 3.1 hours in the 2U server (b) and 34 % over 3.1 hours in the Open Compute server (c).

servers, or in a datacenter constructed with an oversubscribed cooling system to run under peak power due to thread and cache contention issues, contention reducing techniques [80, 67, 138, 136] that enable increased utilization through collocation increase the cooling load unsustainable.

In this oversubscribed datacenter, thermal management techniques such as downclocking/DVFS or relocating work to other datacenters [75, 74, 76] must be applied to prevent the datacenter from overheating.

In Figure 3.11, we plot the cluster throughput if the thermal limit did not exist and downclocking is not imposed, the throughput without wax, and the throughput with wax. In the trace without wax, downclocking to 1.6 GHz is imposed to prevent the cluster from overheating and throughput is normalized to the peak throughput while downclocked. Below the thermal limit, all three have the same throughput.

By adding PCM into the servers, we are able to maintain clock speeds and/or utilization as the wax absorbs thermal energy and until the thermal capacity of the wax is full. Once the wax is melted and can absorb no more energy downclocking or job relocation must be applied to prevent the datacenter from overhearing, but wax delays this by three to five hours.

In the Open Compute cluster, PCM delays the onset of thermal constraints by 3.1 hours and we observe a 34% increase in peak throughput during that time. In the 1U low power cluster, PCM delays thermal constraints by 5.1 hours with a 33% increase in peak throughput, and in the 2U high throughput cluster PCM delays thermal constraints by 3.1 hours and increases peak throughput by 69%.

To evaluate the impact of the increased throughput, we consider TCO efficiency: the ratio of TCO with increased peak throughput from PCM to the TCO required to achieve the same peak throughput without PCM. When thermal constraints lead to a decrease in throughput, we would need additional machines at significant additional cost to make up the difference. Thus an improvement without increasing the number of machines can lead to significant TCO efficiency savings.

We model TCO using Equation 3.1 with the assumption that most CapEx–including the facility space, power and the cooling infrastructure without PCM–are linear to the critical capacity of a datacenter [14]. OpEx terms related to the servers in Equation 3.1, such as server energy and cooling energy are proportional to the increase in the throughput and thus increase with or without wax.

In the 10 MW datacenter consisting of 1U low power servers, PCM achieves a TCO efficiency improvement of 23%, 39% in the 2U datacenter, and 24% in the high density Open Compute datacenter.

3.5 Summary

In this chapter, we introduce thermal time shifting, the ability to reshape a thermal load by storing and releasing energy when beneficial. We study paraffin wax, a phase change material that we place inside a real server to demonstrate thermal time shifting in a single server and validate a suite of software simulations we develop to study thermal time shifting on the cluster and datacenter scales. We show that thermal time shifting with a PCM can be used to reduce peak cooling load by up to 12% or increase the number of servers by up to 14.6% (5,300 additional servers) without increasing the cooling load. In a thermally constrained datacenter, we demonstrate that PCM can increase peak throughput by up to 69% while simultaneously postponing the onset of thermally mandated throughput reduction by over three hours.

CHAPTER IV

Virtual Melting Temperature: Managing Server Load to Minimize Cooling Overhead with Phase Change Materials

As the power density and power consumption of large scale datacenters continue to grow, the challenges of removing heat from these datacenters and keeping them cool is an increasingly urgent and costly. With the largest datacenters now exceeding over 200 MW of power, the cooling systems that prevent overheating cost on the order of tens of millions of dollars. Prior work proposed to deploy phase change materials (PCM) and use Thermal Time Shifting (TTS) to reshape the thermal load of a datacenter by storing heat during peak hours of high utilization and releasing it during off hours when utilization is low, enabling a smaller cooling system to handle the same peak load. The peak cooling load reduction enabled by TTS is greatly beneficial, however TTS is a passive system that cannot handle many workload mixtures or adapt to changing load or environmental characteristics.

In this work we propose VMT, a thermal aware job placement technique that adds an active, tunable component to enable greater control over datacenter thermal output. We propose two different job placement algorithms for VMT and perform a scale out study of VMT in a simulated server cluster. We provide analysis of the use cases and trade-offs of each algorithm, and show that VMT reduces peak cooling load by up to 12.8% to provide over two million dollars in cost savings when a smaller cooling system is installed, or allows for over 7,000 additional servers to be added in scenarios where TTS is ineffective.

4.1 Background– TTS

In user-facing workloads such as Web Search, work must be done in response to requests placed by the user at the time the request is placed [85] unlike HPC workloads or batch workloads that may be scheduled then executed at a later time [14, 139]. This quality of user facing workloads is responsible for the diurnal cycle: load is high when users are awake and active, and low when they are not. Although the cumulative service load may be distributed through time zones around the world in such a way that flattens global peaks and troughs, latency considerations require a certain amount of spacial locality that prevents completely arbitrary workload placement with respect to geography [74].

This means that user facing work cannot be reassigned to a different time spot or redistributed geographically to amortize the inefficiencies of a diurnal cycle. Given the correlation between work, power, and heat that must be removed by the cooling system (the cooling load), datacenters typically provisioned the cooling system for peak load, resulting in a system that is fully utilized for only a small portion each day and significantly underutilized the rest of the time (a problem that has only become more pronounced as servers become more energy proportional [13]).

Prior work [116] showed that TTS can greatly reduce the peak cooling load of a datacenter, providing significant cost savings by reducing the size of the cooling system needed or providing cooling for thousands more servers under the same cooling budget. This is particularly important as datacenters continue to grow because, while the cooling system is an integral and critical part in the design of every datacenter,

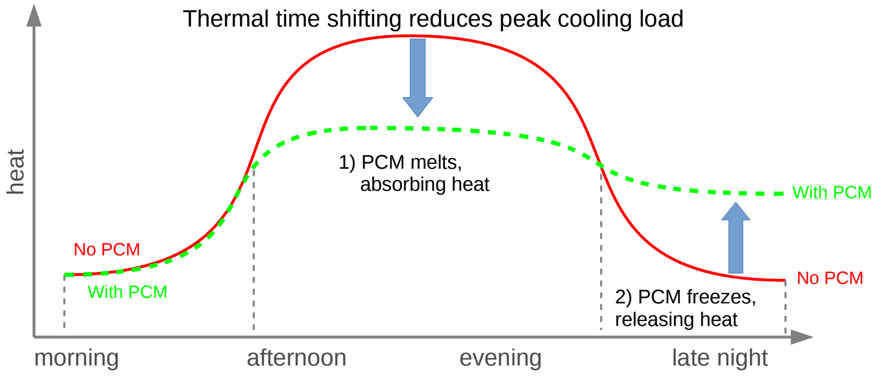


Figure 4.1: Thermal time shifting with PCM

the cooling system itself does not directly contribute towards revenue generation. With cooling infrastructure costing millions of dollars for even modestly sized datacenters [65] and consuming millions of MWh annually [120], working towards more efficient and affordable cooling systems is of critical importance.

TTS proposes to place a small amount of PCM in each server in a datacenter running primarily user-facing workloads. These types of workloads typically see a diurnal load pattern with a high peak during the afternoon/evening and a large trough during the late night [14, 139], however a diurnal cycle is particularly problematic for the cooling system because the system size must be provisioned for peak load even though it spends most of the day running at levels considerably below the peak (Figure 4.1). TTS addresses this by raising the minimum load and lowering the maximum load, increasing average utilization in an appropriately resized cooling system. In the right configuration, TTS can accommodate the same load without overheating or thermal downclocking.

To enable TTS, the PCM must have a melting temperature that lies between the peak and trough such that during the peak hours wax melts and stores thermal energy, and then during the off hours when the load is low the PCM solidifies and releases the stored energy. The total amount of energy stored is proportional to the latent energy of the PCM (the amount of energy absorbed during the phase transition), and how much PCM melts. The sensible heat (energy required to raise the temperature of the PCM without a phase transition) also stores energy, but typically stores several times less energy than the phase transition [116, 96, 113]. TTS does not inherently remove heat or reduce the amount of heat that must removed from the datacenter.

TTS proposes to alter this paradigm, storing thermal energy at the peak and releasing it during the off hours to flatten the cooling load. This enables two major opportunities for cost savings. First, the cooling system in a datacenter may now be sized for a reduced peak load, saving hundreds of thousands of dollars per year in amortized TCO, or alternatively second: more servers with a higher peak power load may be added to the same datacenter without increasing the peak cooling load and saving millions of dollars over a new cooling system in a retrofit scenario [116].

However, a passive management system for TTS that only melts or cools wax at a specific and set threshold [116] (the physical melting temperature of the wax) cannot handle many mixtures of different workloads, especially as the types, prevalence and power characteristics of these workloads change over the lifetime of the datacenter and may change as frequently as day to day or hour to hour.

PCM Selection - Thermal energy storage can be accomplished with any PCM, however not all PCMs are appropriate for deployment in a datacenter. Commercial paraffin wax is particularly advantageous, not only because it is non-corrosive and non-conductive in case of a leak, but also because it is cheap and available with a range of melting temperatures typically between 40 and 60 °C. Molecular n-paraffins can have lower melting temperatures, but are cost prohibitive to deploy in a datacenter [116, 96, 113, 47].

Wax Placement - TTS proposes to place the wax directly inside of each server, behind the CPU heat sinks occupying empty air space left available for expansion card slots and other configuration options. Prior work demonstrated the benefits of TTS in a variety of servers including low power and high throughput commodity servers as well as high density Microsoft Open Compute servers for workloads with heterogeneous thermal profiles [116].

4.2 Virtual Melting Temperature

VMT actively manages workload placement to control the distribution of temperatures within the datacenter, raising the temperature in a subset of servers to melt wax (and thus store heat) while lowering the temperature in other servers to reduce the peak cooling load for the whole datacenter. This creates a ”virtual” melting temperature where, although the average temperature is unable to melt wax, we initiate melting in a subset of servers to benefit from heat storage. VMT gives the system or operator active control over the melting and cooling cycles of wax in the datacenter.

Without VMT, the datacenter's ability to target the best physical melting temperature (the point at which temperature of the server is held stable while the material melts) is relatively limited and, most importantly, remains constant for the life of the server unless the wax is swapped out and replaced (labor intensive). VMT is a technique that allows a datacenter to vary the apparent melting temperature in the datacenter to melt wax even if it would not normally melt. With a diverse workload, we can create thermal imbalance via job placement. With a homogeneous workload we can do the same through load imbalance; for this work we assume the former.

VMT can also raise the melting temperature by locating hot jobs in a subset of servers with already melted wax, preserving wax in anticipation of a very hot peak still to come. However, the focus of this work is on reducing the melting point rather than increasing it.

In this section we present two scheduling algorithms to enable virtual melting temperature: a thermal aware algorithm that sorts and places jobs based upon their thermal properties, and a wax aware algorithm that additionally reallocates jobs away from fully melted servers.

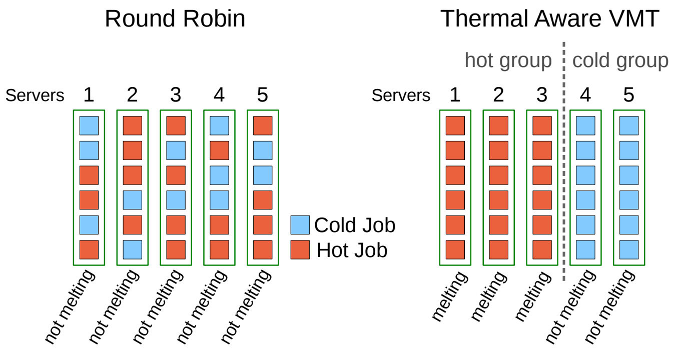


Figure 4.2: Thermal Aware VMT scheduling

4.2.1 VMT with Thermal Aware Job Placement

VMT with thermal aware job placement (VMT-TA) proposes to divide the cluster into a hot group of servers and cold group, then schedule jobs with a hot thermal profile in the hot group while jobs with a cold thermal profile are placed in the cold group (Figure 4.2). (Note that hot group and cold group servers do not need to physically clustered: they can be distributed throughout datacenter to maintain the same cluster or DC-level temperature distributions.)

Jobs are placed into either the hot group or the cold group based on the thermal profile of the workload they belong to: if a server filled with only a single workload can melt significant wax over a peak load cycle, regardless of whether the jobs can be colocated with itself enough times to do so as long as they could be collocated with other hot jobs, the workload is considered hot and VMT will attempt to located these jobs together in the hot group. Otherwise, the workload is labeled cold and VMT attempts to place jobs in the cold group.

In such a configuration, the hot group can melt wax even if the mean temperature within all of the servers or mean temperature with round robin/non-thermal-aware scheduling is not high enough to melt wax.

To calculate the number of servers placed in the hot group, VMT-TA uses a ratio of the user-set Grouping Value (GV) divided by the Physical Melting Temperature

(PMT) of the wax in the following formula:

$$hot\_group\_size = \frac{GV}{PMT} \times num\_servers \tag{4.1}$$

Where num_servers is the number of servers in the cluster and hot_group_size is the number of servers in the hot group.

There is not a general solution that maps the GV to a Virtual Melting Temperature (VMT) because such a mapping depends on the PMT as well as the workload power profile and workload mixture, however a mapping can be experimentally derived for a given combination. In Section 4.4.1 we show a GV to VMT mapping for our test datacenter.

After calculating the hot group size, the cold group is simply composed of the remaining servers:

$$cold\_group\_size = num\_servers - hot\_group \tag{4.2}$$

To implement VMT-TA, workload types are first classified as hot jobs or cold jobs based upon thermal characteristics. This can be done using on-package thermal sensors and/or power sensors or models (e.g. Intel RAPL). Once deployed, hot jobs are placed in the hot group of servers while cold jobs are placed in the cold group .

Within each group, jobs are distributed evenly among the servers. Here care must be taken to ensure each group is large enough to support the peak load for its respective subset of workloads else individual queries must be dropped or queued causing QoS degredation. This can be handled by dynamically adjusting the VMT to modify the group sizes or by allowing jobs to be scheduled to the other group if one group fills up.

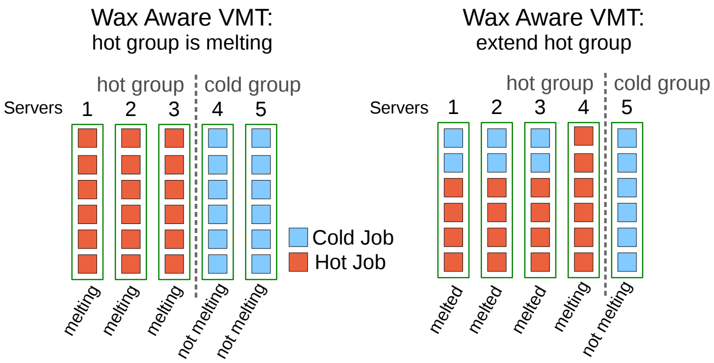


Figure 4.3: Wax Aware VMT scheduling

4.2.2 VMT with Wax Aware Job Placement

Last, we propose VMT with wax aware job placement (VMT-WA). Where VMT-TA has no mechanism to handle all of the wax in the hot group melting early, VMT-WA monitors the melted state of the wax and automatically increases the size of the hot group if all of the wax melts before the end of the load peak.

At its simplest, VMT-WA schedules just like VMT-TA until wax on a server in the hot group is fully melted. Unlike VMT-TA, once the wax is fully melted in a server VMT-WA moves a server from the cold group to the hot group, maintains just enough load on the melted servers to keep the wax melted, and moves the additional load to the newly added server to continue melting wax (Figure 4.3). A detailed description of the algorithm follows.

VMT-WA begins by calculating the size of the hot and cold groups using Equation 4.1, the same as VMT-TA, however the group sizes are dynamically updated as wax melts and cools.

Periodically, the cluster scheduler updates the size of the hot and cold groups by scanning the amount of wax melted on each server. The scheduler compares each server against the Wax Threshold, the fraction of the wax melted above which the server is considered completely melted, and adds each server above the threshold to a list of fully melted servers. After counting the number of servers in this list, servers are removed from the cold group and added to the hot group based upon current load trends. During each update, the scheduler restarts from the minimum hot group size (Equation 4.1) and adds servers in order. To the extent possible, we do not transition servers from the hot group to the cold group during the peak because cooling a melted server releases heat.

When placing individual jobs, the scheduler considers the job's thermal classification (the same as VMT-TA) but does not strictly place the job in the corresponding server group. For hot jobs, the scheduler first attempts to schedule the job in the hot group by considering a subset of servers in the hot group that are currently below a certain amount of wax melted (the wax threshold) or are below the wax melting temperature. Placing a hot job on either such server will attempt to melt more wax or keep already molten wax melted (both advantageous for reducing cooling load).

If there are no hot group servers meeting these characteristics (possible with sudden spikes in load), then servers are added to the hot group from the cold group sequentially until the hot group includes a server that is below the wax threshold or melting temperature. In the event that no such servers exist (a corner case where all servers are added to the hot group) then the job is scheduled on any server below the melted threshold or, barring that, any remaining servers.

To place a cold job, the scheduler first attempts to place the job in the cold group. If the job cannot be placed in the cold group (as may occur when the hot group grows), the scheduler attempts to place the job on a server in the hot group that is already above the melted threshold and melting temperature to minimize thermal impact. If the job cannot be placed in these servers either, then the job is placed into one of the remaining hot group servers.

This ordering of scheduling policies will only fail to schedule a job in the case where a thermally unconstrained datacenter would also run out of computational space, so we do not model that case.

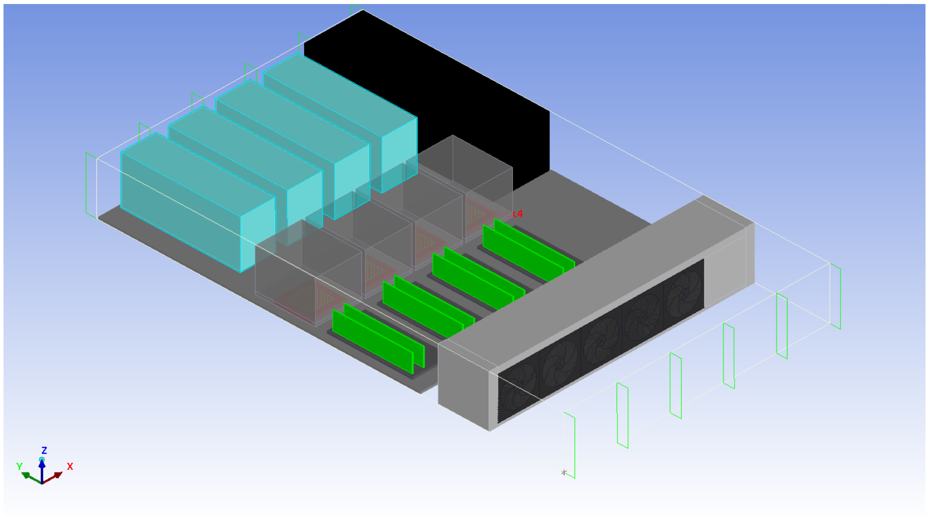


Figure 4.4: Test server with 4.0 liters of wax (light blue) behind the CPUs

VMT-WA requires knowledge of the current melted state of wax in servers in the cluster to adjust the size of the hot group properly. A single temperature sensor on the exterior of the wax container can tell us when the wax starts melting or freezing, then we add a light weight model of current wax state running on each server. The model uses the CPU power consumption and temperature sensors already in the server to estimate the current state of the wax based upon a lookup table [117].

4.3 Methodology

4.3.1 Datacenter Architecture

We consider a datacenter running a Google-style suite of workloads: all are user facing with different latency requirements that, with modern contention reduction techniques [80, 67, 136], allow for collocation on the same servers. Within the datacenter, servers are divided into homogeneous clusters and job scheduling is performed at the cluster level. In Section 5.4 we perform a number of cluster-level scale out studies on a cluster of 1,000 servers (with some parameter sweeps performed with 100 servers to reduce total compute time). To perform a datacenter-level TCO analysis, we consider many clusters summing to a critical power of 25 MW, just shy of the 27.25 MW median critical power for large scale datacenter reported by Ghiasi et al [40].

We provision the datacenter with 2U high throughput servers (Figure 4.4), based upon the internal layout of a Sun Fire X4470 server but populated with 4x Xeon E7-4809 v4 CPUs. In this form factor, this corresponds to approximately 20 servers per rack and 50 racks per cluster. Each server has a peak power consumption of 500 W, and an idle power consumption of 100 W. Per core power consumption is approximated using a linear model [65].

Based upon computational fluid dynamics (CFD) design space exploration, this server can hold 4.0 liters of wax without exceeding CPU thermal limits [116]. The wax in each server is commercial paraffin wax with a melting temperature of 35.7 °C, the lowest commercially available temperature [90].

The paraffin wax is divided between four aluminum containers that contain the wax when molten and provide surface area contact for heat transfer from the air to the wax. Even though the paraffin wax has a melting temperature of 35.7 °C, the lowest of a commercially available paraffin wax, for many workload compositions this is not low enough to melt wax even at peak load due to the thermal characteristics of the datacenter and the workloads.

Each server in the cluster maintains its own model of the state of the wax inside of it [117]. We update the model once per minute based upon load and temperatures in the last minute, and report the wax state to the cluster level scheduler when it is updated. Running only once per minute, the update process provides a negligible impact on server and network performance.

4.3.2 Workloads

We consider a cluster of servers inside of datacenter running 5 different workloads (Table 4.1). All of the workloads can be co-located within the same server, however they are assigned separate physical cores and never share simultaneous multithreading (SMT) contexts to reduce the complexity of contention mitigation techniques.

Of the five workloads two are user-facing, latency critical workloads that demand immediate responses back from the server: Web Search and Data Caching. These workloads have strict QoS requirements on the order of milliseconds or microseconds.

The other three workloads perform user-facing functions that demands a degree of QoS (that is, they are not batch jobs that can be scheduled hours later) but are not as strict as web search and data caching. Video Encoding, e.g. for Youtube, Virus Scanning files, e.g. for uploading files to Google Drive, and Clustering, e.g. for web advertisements, demand computation be near when the action is initiated to ensure benefit (responsive file downloads, relevant ads, etc.) but a runtime difference on the order of seconds will not greatly reduce the user experience. On these workloads, we consider a datacenter running contention mitigation techniques [80, 67, 136] that allow a small performance penalty to ensure that the latency critical workloads meet their QoS requirements.

To enable sorting and placement using VMT, jobs are classified as either a 'hot' or 'cold' based upon whether their power and temperature profile would enable them to melt significant wax if run in isolation.

Web Search - We consider the CloudSuite 2.0 Web Search benchmark [34]. Web Search shards queries to multiple servers, each holding a portion of the index, and returns the results based upon the users query. Using power profiling of Web Search [129], we classify it as a hot job for VMT.

Data Caching - For Data Caching, we consider the CloudSuite 2.0 implementation using the Memcached server framework to meet the demands of a social media

Table 4.1: Workloads considered for scaleout study (power is normalized to a single 8 core Xeon E7-4809 v4 CPU; each server contains four CPUs).

Workload	CPU Power	VMT Class
WebSearch	37.2 W	hot
DataCaching	13.5 W	cold
VideoEncoding	60.9 W	hot
VirusScan	3.4 W	cold
Clustering	59.5 W	hot

service [34]. The Memcached server must respond in real-time to user requests, performing a number of memory operations on large sets of data. With relatively low CPU power consumption [129], VMT classifies data caching as a cold job.

Video Encoding (h264) - We consider the SPEC 2006 implementation of h264 video encoding [50]. Video media uploaded to video sharing sites such as Youtube are re-encoded to several different file sizes [21] before users can share or view the video. As such, although this is not a batch job where the host provider can leave the user waiting potentially several hours until the encoding can be scheduled during off hours [139], a small delay of seconds or even minutes for longer videos is tolerable. Based upon power measurements of h264 Video Encoding [116], we classify video encoding as a hot job for VMT.

Virus Scanning - Files uploaded to a file host like Google Drive are scanned for viruses before they are shared, converted or downloaded [2]. We consider a virus scanner [129] running on freshly uploaded files. Similarly, these are not latency critical but cannot be delayed for batch job scheduling. Based upon power profiling of VirusScan [129], virus scanning is classified as a cold job for VMT.

Clustering - Clustering is commonly used to deliver ads targeted ads based upon user actions on the web [23]. This is a computationally intensive task with some leeway for contention mitigation, but the sooner it can finish then the sooner relevant ads can be delivered to the end user [23]. This makes batch job scheduling possible but not ideal in many cases. Based upon power profiling [78], we classify clustering as a hot job for VMT.

4.3.2.1 Workload Migration

If a job cannot be migrated at all and load cannot be redirected to a different host then VMT cannot be used, however this is a relatively rare case.

Of the diverse workloads we consider, all can be migrated or reallocated but some are more portable than others. Virus Scanning and Video Encoding, for example, are very portable with data requirements dominated by the incoming files to be scanned or encoded. Web Search on the other hand requires a large amount of data that is not very portable, however multiple copies of the data are already distributed throughout the datacenter to enhance query speed and redundancy [12]. This allows a degree of flexibility in job placement without requiring data migration that VMT exploits.

4.3.3 Workload Colocation and Interference

In Figure 4.5, we consider latency scaling with load and cores for mixtures of Web Search and Data Caching servers from the Cloudsuite 3.0 benchmark suite [34] running on a 6 core E5-2420 CPU with Turbo Boost turned off. All jobs are scheduled on separate cores and have sufficient main memory to prevent swapping to disk but still may interfere in the last level cache and memory bandwidth. No contention reduction techniques are applied during this test. Data Caching RPS per server core was fixed at was 45k when collocated with Web Search, while Web Search clients per server core was fixed at 37.5 when collocated with Data Caching.

For Data Caching, we observe that that at very low loads, when QoS targets are most often met, 6 cores running together provides the best latency. Similarly at high latencies when QoS targets are violated 6 cores once again provides slightly better average QoS, however in the middle range for Data Caching a mixture provides

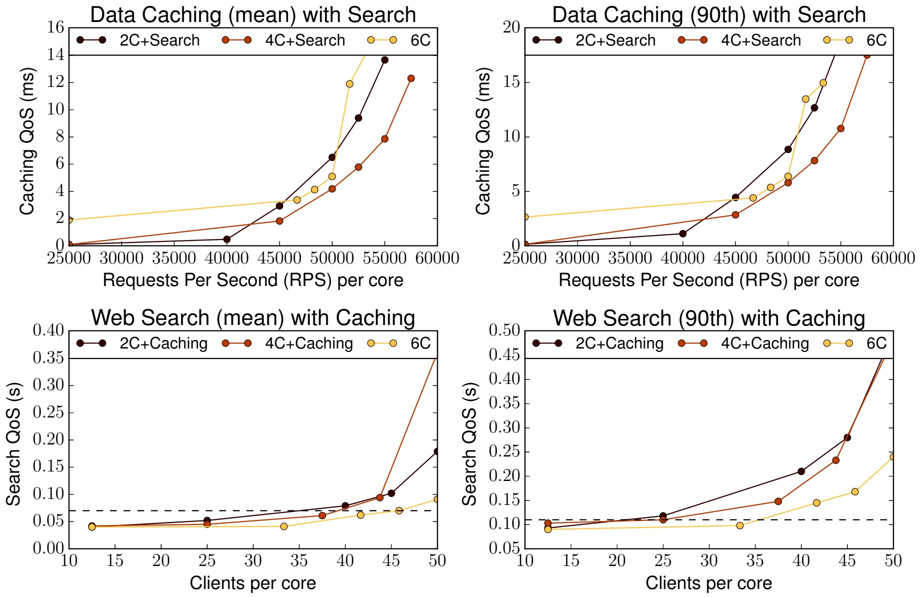


Figure 4.5: Latency scaling with load and cores for Web Search and Data Caching colocated on a Xeon server, without contention reduction techniques. Caching contention is within an acceptable range when colocated versus not, and Web Search exhibits behavior that can be managed using previously studied contention mitigation techniques.

similar or better performance than homogeneous workloads as the memory resources are split between memory intensive data caching and more compute intensive web search. Given that the total load must meet QoS with all cores allocated to one workload at load, (thus dividing peak resource utilization approximately even by the number of cores), we assert that the high latency sensitive workloads will be able to coexist in general with other high latency sensitive workloads. Corner cases that may arise (e.sg. specific cache thrashing access patterns) can be mitigated by dynamic management and recompilation techniques [80, 67, 136] or allocated to non-VMT-enabled servers.

For Web Search, we observe decreased performance across the whole range of clients per core. Here, it is important to observe that even with 6 cores running only Web Search the clients per core are limited by QoS targets to return data to the user. As these are compute heavy workloads and sufficient memory bandwidth was

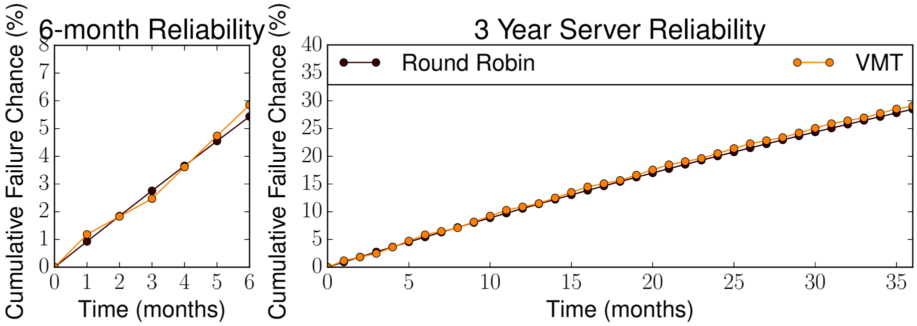


Figure 4.6: Server reliability for round robin versus VMT-WA when 20% of servers are rotated each month (3 months in the hot group, 2 months in the cold group). After 3 years, the cumulative failure rate for VMT-WA is 0.4% higher than for RR.

available with 6 cores, the slowdown is likely caused by cache interference which can be mitigated by BubbleUp and Protean Code [80, 67].

4.3.4 Server Reliability

The impact of thermal wear on computer and server components has been extensively studied [99, 111, 28]. Using VMT, servers in the hot group experience higher average utilization and the temperature of many components increases relative to a round robin or coolest first scheduler (thus exhibiting a higher failure rate) while servers in the cold group experience the opposite.

To ensure even wear leveling across components, servers should therefore be rotated between the hot group and cold group regularly [7, 119, 36, 20]. In Figure 4.6, we plot the 6 month and 36 month (3 year) cumulative failure rates using a RR scheduler and a VMT-WA scheduler.

To model the failure rate, we first begin with a 70,000 mean time before failure (MTBF) at 30 °C based on numbers from Intel [55]. We scale this reliability using the rule of thumb that a 10 °C increase in temperature doubles the failure rate of components [93, 28] to adjust the rate of failure to the temperatures in our test datacenter.

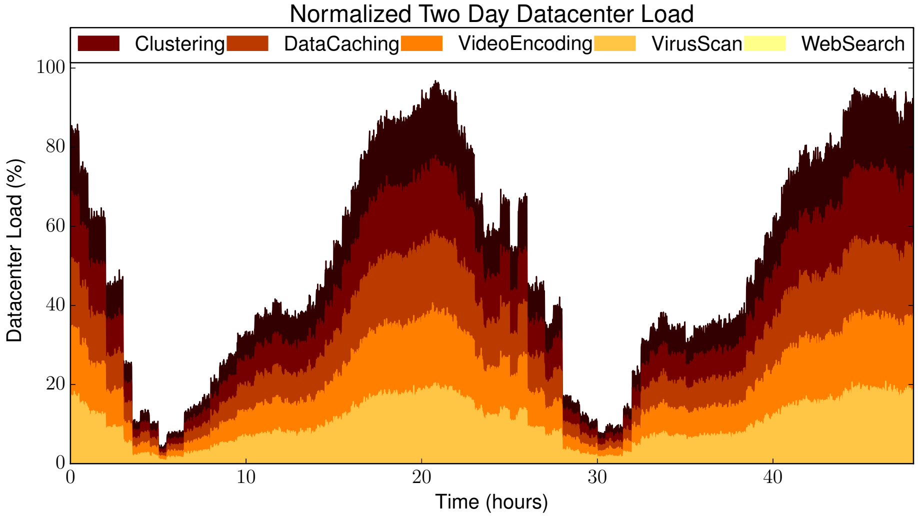


Figure 4.7: Two day load trace (cumulative for 100 servers)

We then assume a 20% rotation per month, where each server spends two months in the cold group and three months in the hot group based upon our breakdown of workloads. After 3 years [14], the cumulative failure rate of all servers for VMT-WA is only 0.6% higher than for round robin.

4.3.5 Simulation Infrastructure

We perform our scale out simulation using DCsim [65], an event-driven simulator to model a cluster of 1,000 servers. The wax model for the server is based upon real hardware measurements to validate a CFD model of a test server [116], and then the CFD result is used to derive model parameters for DCsim. (CFD is more accurate, but computationally infeasible to solve at the granularity needed to evaluate VMT in a cluster-level scale out study.) The cluster results from DCsim are then multiplied linearly to calculate the effects of VMT workload placement policies on the datacenter level.

We use a two day trace of datacenter load from Google [123], normalized using a similar procedure to Kontorinis et al. [65]. The total load is divided between our five workloads, providing a roughly 60-40 split between hot jobs and cold jobs (Figure 4.7).

The load pattern on these two days, up to 95% server utilization, represent the worst case days for the the cooling system. Servers are usually provisioned such that peak daily load is much lower than the total capacity [14], resulting in server and cooling capacity that is underutilized. We consider atypically high day-to-day utilization over two days to realistically stress the cooling system and VMT algorithms in our evaluation.

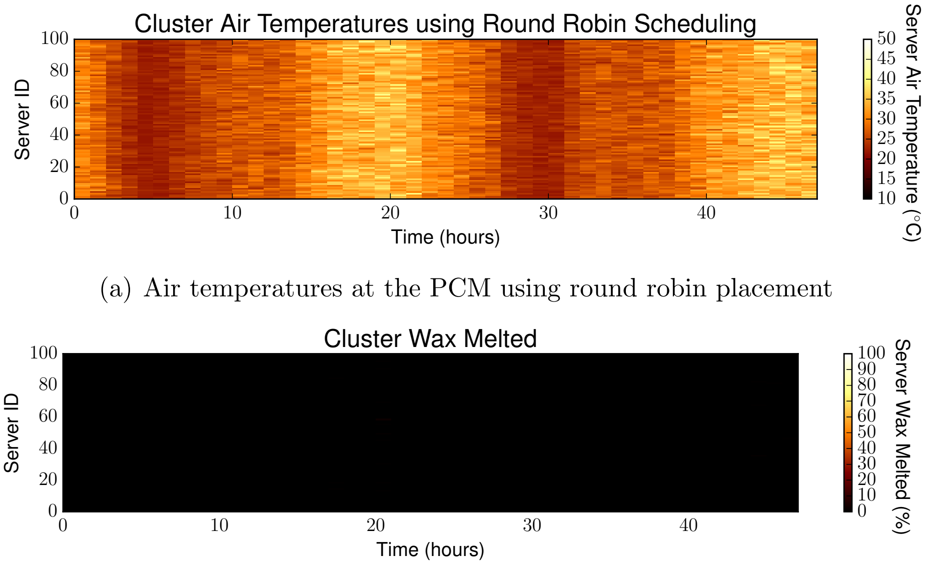
4.3.6 TCO Model

To quantify the cost-saving benefits of VMT, we consider the TCO of the cooling system in a datacenter. When constructing a datacenter, the age of non-IT infrastructure (facilities, cooling, power distribution, etc.) is typically expected to outlast the IT infrastructure (servers, networking equipment, etc.).

To estimate lifetimes, costs and benefits we adapt the TCO calculations from Kontorinis et al. [65] to our datacenter. They use a 10 year linear depreciation for non-IT infrastructure including the cooling system, and a 4 year depreciation for servers.

To calculate the depreciation cost of the cooling system alone, they report a depreciation cost of $7.00 per kilowatt of critical power per month. With a cooling system expected to depreciate over 10 years, this adds up to $84,000 per MW of critical power per year, or $21 million total for 25 MW of critical power.

We evaluate only the cost savings in the cooling system for VMT. The cost to add wax to each server is very small (less than 0.5% of the purchase cost per server at a wax price of $1000/ton), as is the cost savings from utilizing lower electricity prices during the off-peak hours [116].



(a) Air temperatures at the PCM using round robin placement

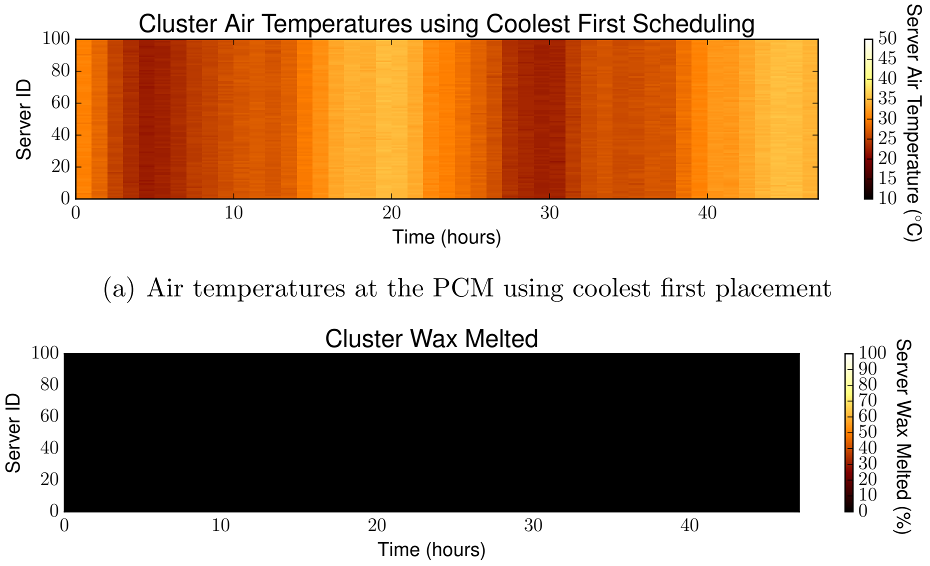
(b) No wax melts using round robin placement

Figure 4.8: Air temperatures and wax melted for 100 servers using round robin placement. The cluster does not benefit from TTS because both the average temperature and individual server temperatures are not hot enough to melt wax.

4.4 Evaluation

In our experiments, we consider two baselines. The first is a round robin scheduler, the same used in prior work on TTS [116]. The second is a more advanced coolest-first scheduler that presumes the coolest servers have the greatest thermal headroom available and schedules on them first.

In Figures 4.8 and 4.9, we plot a heat map of the temperature inside of 100 servers and the portion of wax melted in those servers when jobs are placed according to the round robin and coolest first schedulers, respectively. Both schedulers receive the same workload. Temperature peaks (around 20 hours and 46 hours) and troughs (around 5 and 29 hours) correspond with the peaks and troughs seen in the workload pattern (Figure 4.7). Compared to round robin, coolest first maintains a much tighter temperature distribution between servers as expected of a thermal aware load balancing scheduler, however due to the diverse thermal profiles of these workloads



(a) Air temperatures at the PCM using coolest first placement

(b) No wax melts using coolest first placement either

Figure 4.9: Air temperatures and wax melted for 100 servers using coolest first placement. The cluster does not benefit from TTS because both the average temperature and individual server temperatures are not hot enough to melt wax.

the average temperature in either cluster and the temperatures in each server never reach levels high enough to melt a significant amount of wax.

4.4.1 Thermal Aware VMT

First, we consider VMT-TA in a cluster of 1,000 servers. As noted in Section 4.2, the GV used to calculate the size of the hot and cold groups does not directly correlate to a temperature but is used to control the ratio of servers in the hot group to servers in the cold group.

In Figure 4.11, we plot the average temperature in the hot group of the 1,000 server cluster versus the GV from GV=21 to GV=26. The round robin job placement algorithm almost but does not quite reach the melting temperature. With VMT-TA, the average cluster temperature remains the same as round robin but temperatures in the hot group exceed the melting temperature of the wax.

The degree to which the hot group temperature exceeds the average temperature

Table 4.2: Experimentally derrived mapping between the Grouping Value (GV) and Virtual Melting Temperature (VMT) for the test datacenter.

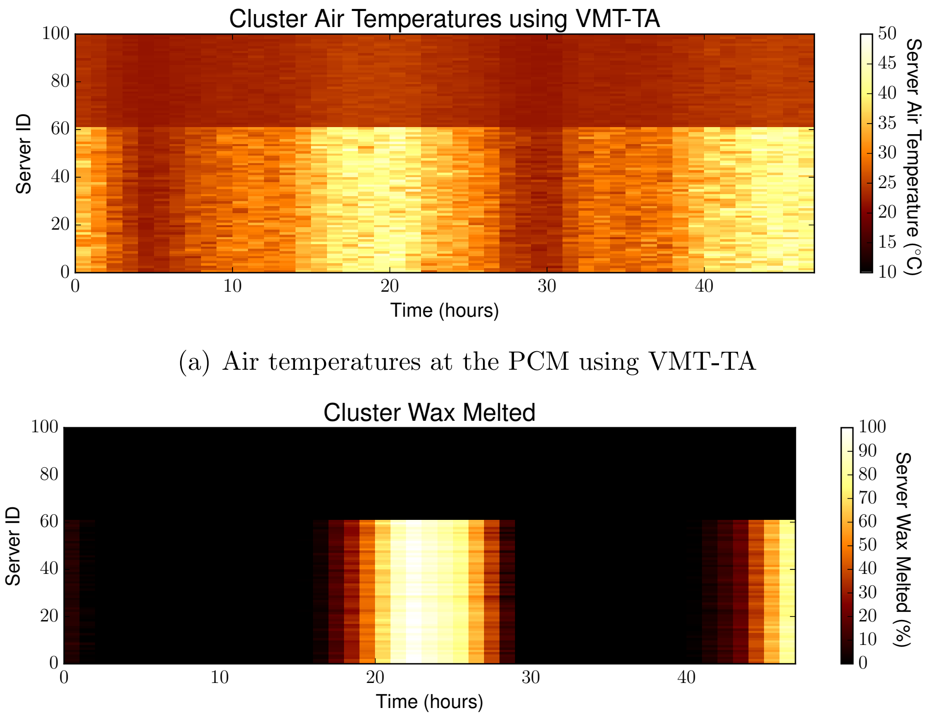
GV	**VMT (°C)**	**ΔPMT (°C)**
20.03	37.7	+2.0
20.14	36.7	+1.0
20.23	35.7	+0.0
20.83	34.7	-1.0
21.25	33.7	-2.0
21.55	32.7	-3.0
21.69	31.7	-4.0
21.84	30.7	-5.0
23.99	29.7	-6.0
30.75	28.7	-7.0

is inversely proportional to the GV value. As the GV setting is decreased the temperature of the hot group increases because there are fewer servers to spread the hot jobs out across, but there is also less thermal energy storage capacity in the hot group because wax is allocated per server.

In Figure 4.12, we plot the cooling load for three GV values: GV=20, GV=22 and GV=24. GV=22 provides the best peak cooling load reduction of 12.8%. GV=24 works about two-thirds as well (8.8%) because the hot group is still hot enough to melt the wax, but not hot enough for long enough to melt all of the wax so some thermal energy storage capacity goes unused. GV=20 on the other hand is even hotter than GV=22, but melts too fast: just over halfway through the peak all of the wax is melted and the thermal storage capacity is exhausted. At this time, the cooling load increases to provide no benefit for the rest of the peak.

4.4.2 Wax Aware VMT

In Figure 4.13 we plot heat maps of server temperature and wax melted on 100 servers using the VMT-WA job placement algorithm with GV=20. At this setting, which does not provide a significant cooling load reduction using VMT-TA because



(a) Air temperatures at the PCM using VMT-TA

(b) Wax melted using VMT-TA

Figure 4.10: Air temperatures and wax melting for 100 servers using VMT-TA, with GV=22.

all of the wax melts prematurely, VMT-WA instead extends the group of hot servers once wax in the hot group servers is melted and continues to melt wax to store energy in the newly added servers.

The temperature impact of this extension is first observable in Figure 4.13(a) at the 19th hour, then more clearly after hour 20 where the hot group is expanded by around 20 servers as more servers in the hot group reach the wax melting threshold. As the hot group is expanded, hot jobs are still scheduled on the servers originally in the hot group to maintain a temperature above the melting temperature. This prevents the premature freezing and release of stored thermal energy; however, additional load that would have gone to those servers now goes to newly added servers with unmelted wax. This has the double advantage of moderating the temperature of the melted servers (at the melting point) and moving new jobs to unmelted servers where more thermal storage capacity is available. As a result, we can see a quick drop in temperatures

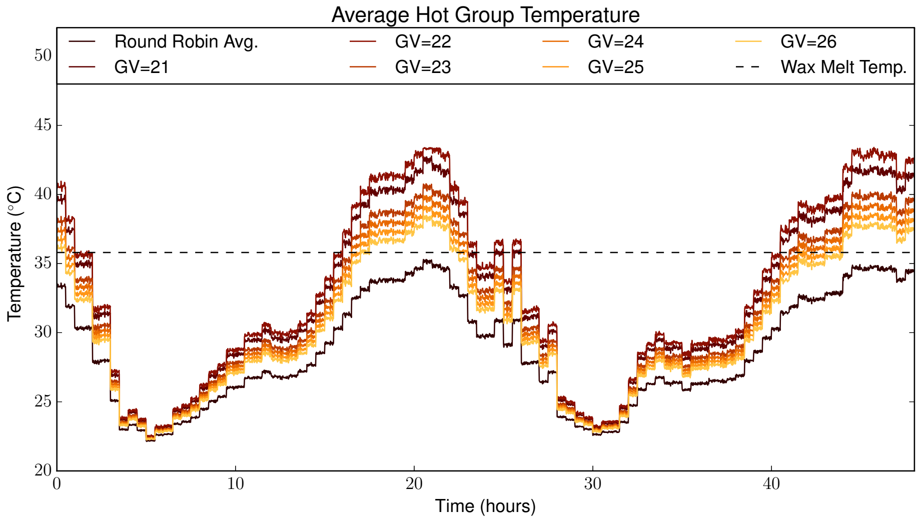


Figure 4.11: Average temperature in the hot group using VMT-TA as the GV is adjusted (for a cluster of 1000 servers)

in the hot group in Figure 4.13(a) and a capping of the cooling load in Figure 4.15 at the same time. This quick drop is a result of the granularity in which VMT-WA adds servers to the hot group.

In Figure 4.13(b), the effects of extending the hot group can be seen in the distribution of wax melted. None of the newly added hot group servers reach a fully melted state, but because the thermal energy storage happens during the melting process and they do melt more wax than otherwise is melted and more thermal energy is stored.

The effect is further visible in Figure 4.14, where we plot the average temperature in the hot group servers on GV values from GV=20 to GV=26. The average temperature drops abruptly (at roughly 20 hours for GV=20 and 21 hours for GV=21) when the wax in the original group of servers for GV=20 and GV=21 melts to the wax threshold. Although the average temperature is now lower, the VMT-WA carefully places jobs to maintain the already melted wax and schedules the newly added servers to exceed the melting temperature and melt as much additional wax as possible. For larger GV values, where the wax never becomes fully melted, the temperature and peak cooling load reduction of VMT-WA closely match that provided by VMT-TA.

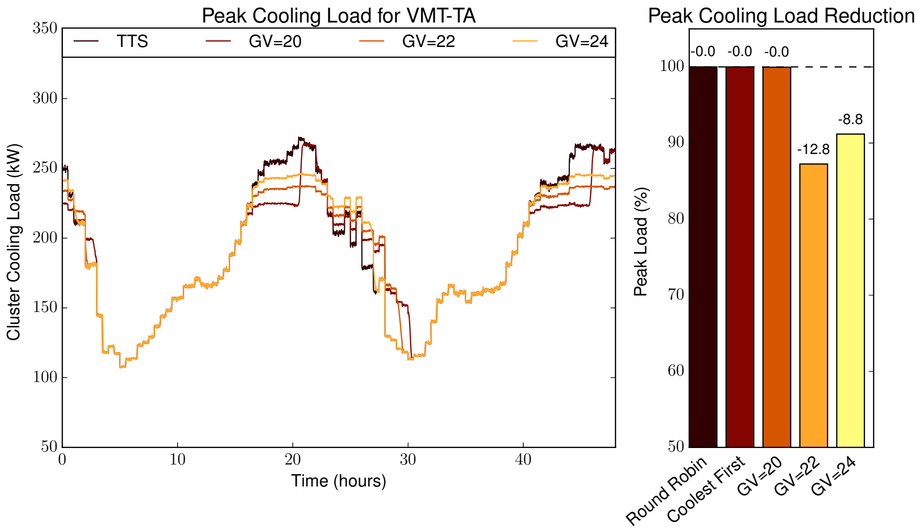
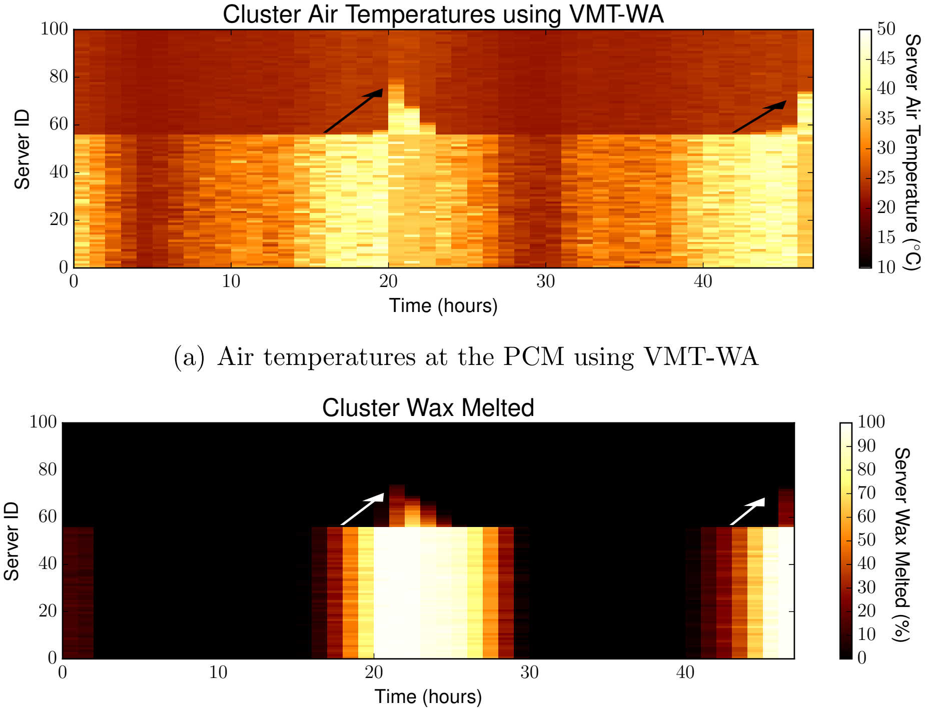


Figure 4.12: Cooling load reduction with VMT-TA at 3 different GVs (for a cluster of 1000 servers). GV=20 begins melting too soon and runs out of wax capacity before the end of the peak. GV=24 begins melting too late, and still has a significant amount of unmelted wax at the end of the peak.

In Figure 4.15, we plot the cooling load for GV=20, GV=22 and GV=24. As with VMT-TA, GV=22 provides the greatest peak cooling load reduction (12.8%). This is expected because the overall workload distribution is very close (approximately 60% hot jobs) to the ratio of GV/PMT used to size the group when GV=22. GV=24 also provides results similar to VMT-TA (8.9%), but GV=20 provides significantly better results.

Unlike VMT-TA, where once the hot group is fully melted the cooling load immediately returns to the level without wax, the cooling load with VMT-WA increase once the wax in the initial hot group is melted but levels off as new servers are added to the hot group and hot jobs are placed on these servers to melt more wax. GV=20 does not provide quite as much benefit as the GV=22 or 24, but still manages a 7.0% reduction in peak cooling load.

Figure 4.16 plots the result of varying the wax threshold, above which VMT-WA considers the wax in a server to be ”fully melted,” from 0.85 to 1.00. (We fix the



(a) Air temperatures at the PCM using VMT-WA

(b) Wax melted using VMT-WA

Figure 4.13: Air temperatures and wax melting for 100 servers using VMT-WA scheduling (GV=20). The hot group servers (bottom) have a consistently higher temperature than the cold group servers (top). Note the expansion of the hot group around 20 and 45 hours correspond with peak load and wax in the hot group reaching the wax threshold.

wax threshold at 0.98, or 98% melted, in all other experiments.) A wax threshold of 1.00 means that all of the wax is melted, however in practice this can be hard to maintain because mild temperature fluctuations can cause small portions of wax to freeze again prematurely. A lower threshold means we are less likely to overshoot the desired temperature, but also risks leaving more wax unmelted and sacrifices some thermal storage. We see from these results that the threshold can be set as low as 0.95 without a noticeable loss in capacity.

4.4.3 VMT-TA vs. VMT-WA

In Figure 4.17, we plot the results of sweeping the GV=10 to GV=30 on 100 servers using VMT-TA and VMT-WA. Both provide peak reduction at GV=22, and

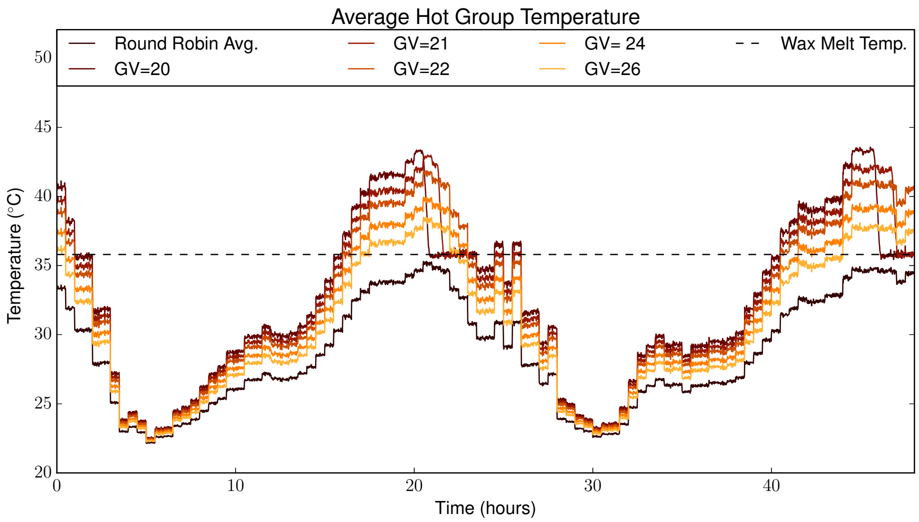


Figure 4.14: Average temperature in the hot group using VMT-WA as the GV is adjusted (for a cluster of 1000 servers). The hot group is extended when the average temperatures for GV=20 and 21 drop.

as the GV is increased both trend downwards together closely. This is the best GV for this specific combination of workloads and PMT, and will vary from datacenter to datacenter. However because VMT gives the ability to control GV, it provides a necessary degree of flexibility and adaptability that TTS does not.

To evaluate VMT-TA versus VMT-WA, the advantage of VMT-WA is most apparent below 22: while the peak cooling load reduction using VMT-TA quickly drops to zero when the hot group melts too quickly and cannot adjust, the reduction using VMT-WA drops to around 6% immediately then continues to decrease much more slowly afterwards.

First, both perform similarly well at GV=22 and above. This is because there is a fixed amount of energy that can be absorbed from the air before the temperature in the hot group will drop below the melting temperature and no more heat can be stored. Even if all of the wax in VMT-WA is melted and the hot group extended, VMT-WA cannot absorb more energy than VMT-TA at the ideal GV setting. The ideal setting may vary as workload composition or daily load levels change.

This shows that the primary advantage of VMT-WA: it is robust. In a scenario

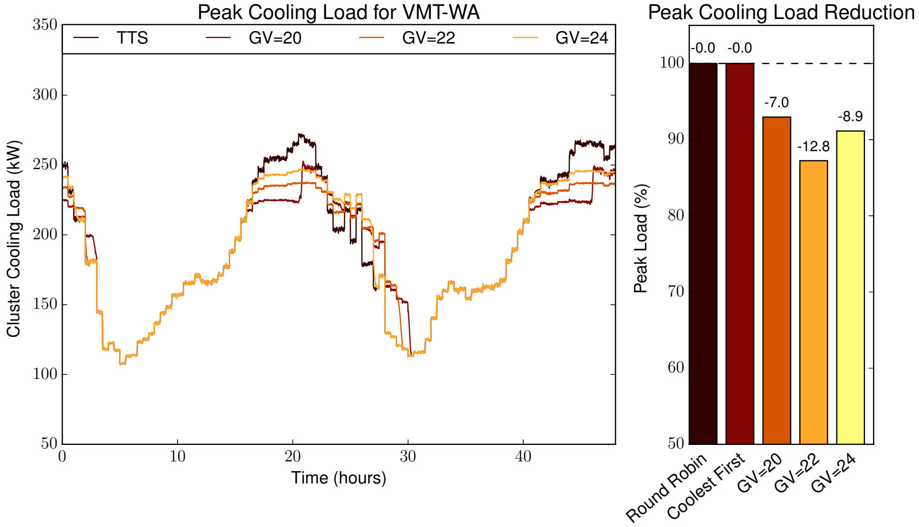


Figure 4.15: Cooling load reduction with VMT-WA at 3 different GV levels for a cluster of 1000 servers. For GV=20 when the hot group becomes fully melted, VMT-WA adds more servers to the hot group to and rebalanced load to continue melting wax.

where the operators can predict load accurately day to day, they can actually change the GV to the optimal value each day. However, with VMT-TA they must choose a conservative value because the risk of selecting a value too low is extreme. With VMT-WA, the risk is more balanced.

4.4.4 Impact of Inlet Temperature Variation

Real datacenters often have some variation in inlet temperature between servers due to airflow [86]. In this section, we consider the impact of server inlet temperature variation on VMT-TA and VMT-WA, and plot the average cooling load reduction from 5 runs with 100 servers each.

In Figure 4.18, we plot the peak cooling load reduction using VMT-TA for inlet temperature standard deviations of 0, 1, and 2 °C (95% within ±0, 2 and 4 °C of the mean) as the GV setting is swept from 16 to 28. We observe that at GV=22 (the peak without inlet temperature variation), no inlet temperature variation provides the best reduction. Below GV=21 or above GV=24 however, non-zero standard deviations

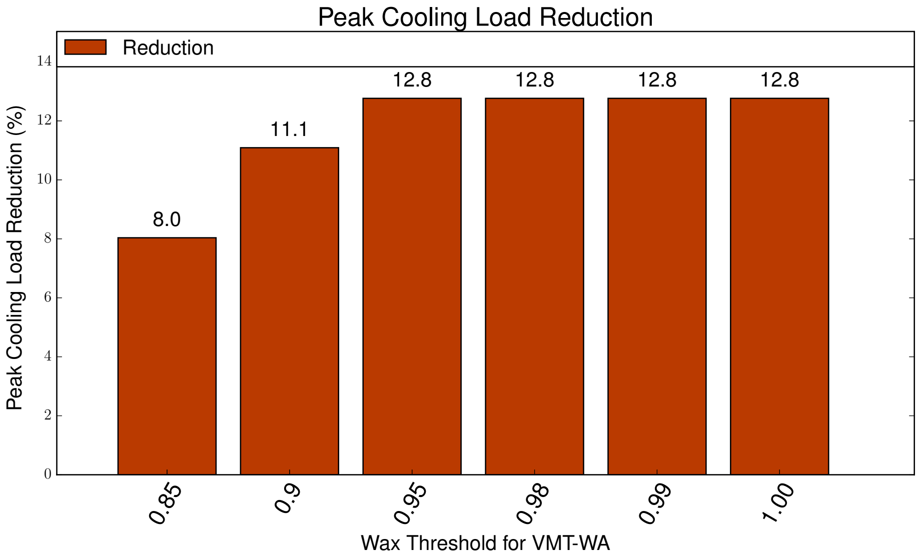


Figure 4.16: Peak cooling load reduction as the Wax Threshold is adjusted for VMT-WA (GV=22) for 100 servers. Maximum reduction is achieved above 0.95.

offer slightly better load reduction than no variation due to the distribution, but still significantly less than near the optimal GV value.

In Figure 4.18, we plot the peak cooling load reduction using VMT-WA across the same range of temperature variations. We observe that like VMT-TA, outside of the best GV range a small deviation provides the same or slightly better reduction. At the peak we also observe a trend where a non-zero standard deviation increases the GV at which peak reduction is achieved. Even with STDEV=2 (95% within ±4 °C), the peak cooling reduction still reaches 10.9%.

We see then that VMT jobs placement is still effective at reducing total cooling load, even in a less uniform environment. The optimal choice of GV increases slightly in this case (because it is better to miss high than miss low); however, we also continue to see that VMT-WA is much more robust with respect to the choice of GV.

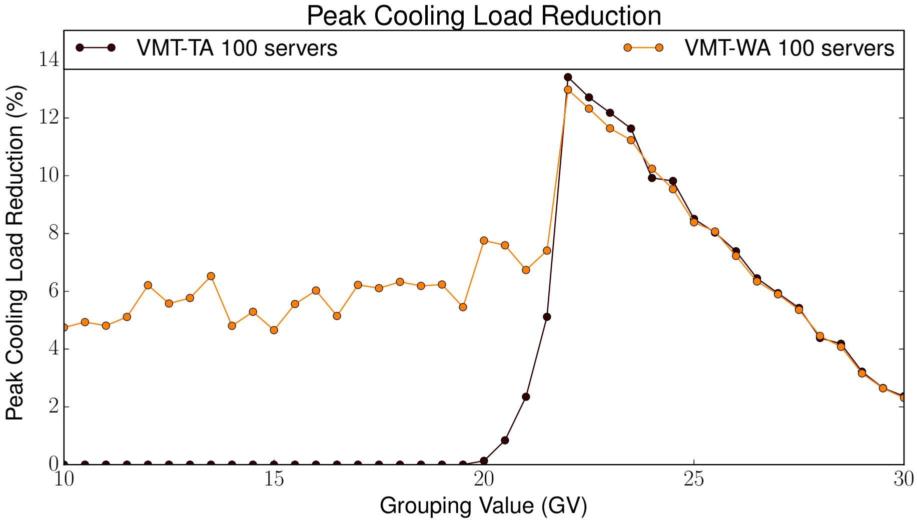


Figure 4.17: Peak cooling load reduction as the GV is adjusted for VMT-TA and VMT-WA (for 100 servers). Both achieve peak cooling load reduction at GV=22.

4.4.5 TCO Benefits of VMT

Lastly, we quantify the potential benefits that come from using VMT to reduce the peak cooling load using a methodology published in prior work [116]. The two primary benefits provided by a reduced peak cooling load are both derived from cooling oversubscription: either that datacenter can now achieve the same throughput with a smaller cooling system, or more servers can be added to increase throughput under the same cooling system. Both provide significant cost savings.

Both VMT-TA and VMT-WA achieve a peak cooling load reduction of 12.8% in a cluster of 1,000 servers, versus less than 0.2% with TTS alone. Considering the 25 MW datacenter from Section 4.3.1, a fully subscribed cooling system would need to remove 25 MW of thermal energy from the datacenter at peak load. (The following cost-savings include cost estimates to deploy wax into every server in the datacenter.)

Decreasing the peak cooling load 12.8% reduces the peak cooling load of the datacenter from 25 MW to 21.8 MW and enables a 12.8% smaller cooling system. This provides a cost savings of $2,690,000 over the lifetime of the datacenter based

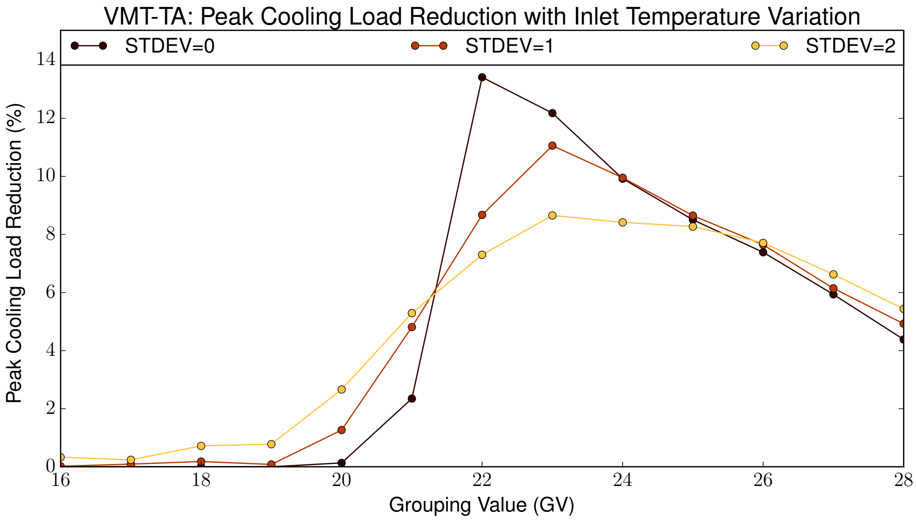


Figure 4.18: Peak cooling load reduction using VMT-TA with normally distributed inlet temperature variation (average of 5 runs of 100 servers each).

upon cooling system cost estimates [65].

(Note that deploying an n-paraffin wax with a melting temperature near 30 °C for TTS to achieve the same peak cooling load reduction would cost on the order of $10 million, four times more than the money with VMT including the cost of deploying commercial wax.)

For a more conservative approach, a datacenter using VMT-WA may choose undertake only a 6% reduction in the cooling system to account for load variation. A 6% reduction in the size of the cooling system still provides a cost savings of $1,260,000.

Alternatively, the reduced peak cooling load may be used to add more servers to the datacenter under the same cooling system size. Using VMT-TA or VMT-WA with the best peak cooling load reduction, VMT enables 14.6% more servers: 146 additional servers per cluster or 7,339 additional servers in a 25 MW datacenter. The conservative 6% percent application of VMT-WA also provides substantial benefit, enabling 6.4% more servers: 64 additional servers per cluster or 3,191 additional servers in the datacenter without increasing the cooling capital expenditure.

The gains from reduce cooling capacity or greater overprovisioning come from

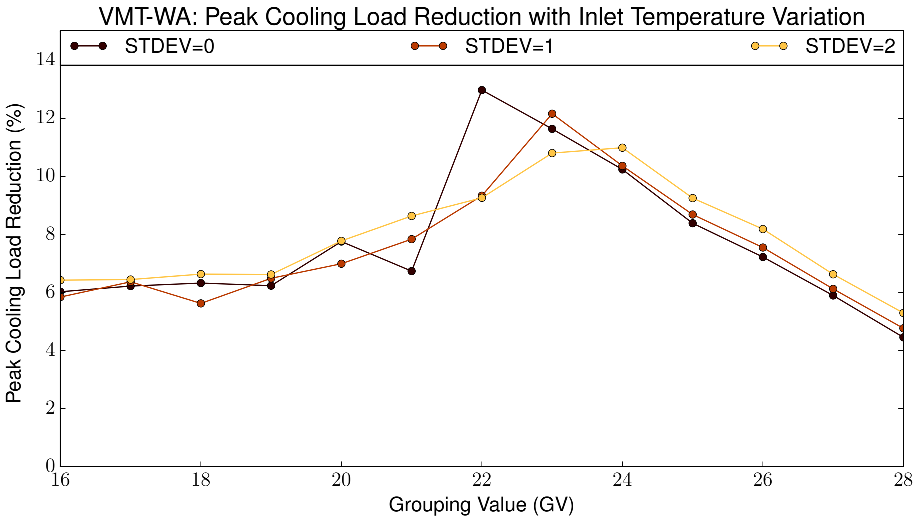


Figure 4.19: Peak cooling load reduction using VMT-WA with normally distributed inlet temperature variation (average of 5 runs of 100 servers each).

using VMT to reduce the peak (annual) cooling load as evaluated in this paper. There may be additional benefits offered by the ability to control the melting temperature day-to-day, such as leveraging less expensive off-peak power or green power when cooling energy can be temporally shifted as well.

4.5 Summary

In this work we introduced Virtual Melting Temperature (VMT), a technique to control the thermal load of a datacenter using workload placement in conjunction with Phase Change Material (PCM)-enabled Thermal Time Shifting (TTS). We presented two algorithms, Thermal Aware VMT (VMT-TA) and Wax Aware VMT (VMT-WA), that manage workload placement in order to maximize melting wax, and thus maximizing energy storage with Thermal Time Shifting (TTS). Both policies group hot jobs together to create warm spots in a subset of servers (which may be distributed throughout the datacenter to maintain balanced power distribution), melting more wax in this subset than if job temperatures were evenly distributed. VMT-WA goes a step further by relocating jobs as wax in the hot group becomes fully melted.

We evaluated these algorithms with a scale out study using a simulated cluster of 1,000 servers enabled with paraffin wax over a two day trace covering a mixture of 5 datacenter workloads with different thermal profiles. We found that both VMT-TA and VMT-WA job placement algorithms provide significant benefits VMT-WA, while slightly more complex to implement than VMT-TA, also incorporates workload movement to create a built-in safety factor against temperature and workload variation. Overall, VMT enables up to a 12.8% reduction in the peak cooling load that corresponds that to over $2.6 million in savings over the life of a datacenter, or adding up to 7,339 additional servers running under the same fixed cooling budget.

CHAPTER V

Thermal Gradient Transportation: Liquid Enhanced Heat Transportation for Energy Storage with Phase Change Materials

The rapid and ongoing growth of datacenters that run the internet and host the cloud has created a multitude of new challenges to build and sustain these warehouse scale computers in an affordable and energy efficient manner. The cooling system in particular is one area where datacenters spend an enormous amount of money and power to prevent overheating and component failure, increasing the datacenter's ecological footprint as a result, but that does not directly add to compute or performance of the datacenter. Direct water cooling is one technology that has been demonstrated to greatly decrease the power consumption of datacenter cooling systems, but has been criticized for a much higher cost to implement.

In this work we introduce Thermal Gradient Transportation (TGT), a technique that leverages direct water cooling to move heat out of the server and into a phase change material (PCM) such as paraffin wax to temporarily store some of that thermal energy. Doing so enables us to reduce the peak cooling load of a dataceter by up to 20% (and 1.97x better that prior work) to enable a smaller and more affordable cooling system, while also handling new workloads such as GPU-accelerated neural networks that prior work on thermal energy storage with paraffin wax could not handle at all.

5.1 Thermal Gradient Transfer

Datacenter resource oversubscription is traditionally accomplished by provisioning resources according to the expected load, even if the theoretical peak resource demand is higher [65, 124, 130, 133]

Prior work showed that by storing thermal energy, the cooling system can be oversubscribed even below the expected peak load by storing a portion of energy to reduce the cooling load during the peak hours of a diurnal cycle [116].

Cooling load is an HVAC metric for how much heat must be removed from a building to maintain a constant temperature. In a datacenter, the cooling load comes almost entirely from the IT equipment and power distribution systems. That is, every watt of electrical power used in the datacenter must be removed as heat, either by active or passive means, to prevent thermal downclocking and component failure.

Thermal Time Shifting (TTS), the technique from prior work, leveraged excess airspace in commodity servers to place a quantity of paraffin wax that undergoes a phase change (melting from a solid to liquid) when the server is heavily loaded, and again (freezing back to a solid) when the server is lightly loaded.

Because the latent energy (the energy required to melt a material) is typically much greater than the sensible energy (the energy required to raise or lower the temperature without a phase change), this class of phase change materials (PCMs) are very efficient at storing thermal energy [97].

Furthermore, by choosing a material with a melting temperature that naturally falls between the high and low temperatures of the diurnal cycle experienced by servers running user-facing datacenter workloads [14], this thermal energy storage can be passively reduce the peak cooling load of a datacenter to enable a greater degree of cooling oversubscription [116].

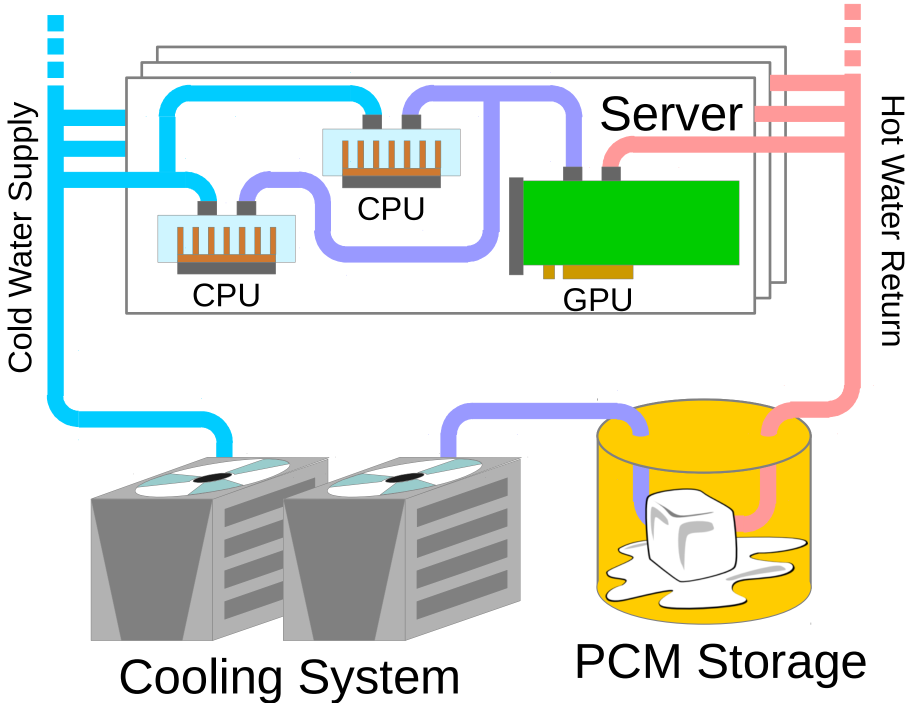


Figure 5.1: Thermal Gradient Transportation (TGT) proposes to leverage server-level water cooling to move heat directly from the CPU and other high power components to a reservoir of PCM, then on to the cooling system.

However, paraffin wax–the best candidate for energy storage in a datacenter [116]–has very low thermal conductivity, the measure of how well the material carries heat [97]. This is critical because only wax that melts is able to store significant energy, and the sensible heat alone does not store enough energy to beneficially lower the peak cooling load.

Prior work addresses this limitation by increasing the surface area of the wax through the use of multiple containment vessels, and by placing the wax directly inside of the server where temperature are greatest (and thus the wax melts the fastest) [116].

Placing wax inside of the server works but severely limits the quantity of wax, and thus amount of energy, that can be stored. Furthermore, new classes of workloads such as deep neural networks and intelligent personal assistants (IPAs) that use accelerators such as GPUs [66, 49, 48] severely diminish, or remove entirely, the space where the wax formerly resided.

These GPU-accelerated systems also use more power and produce more heat, which in turn requires a larger cooling system and more power to the cooling system.

Deploying liquid cooling directly onto the CPU and other high power components has been proposed in a number of works [25, 30, 29, 37, 46, 91, 142]. Direct liquid cooling offers a number of significant benefits to datacenters, including a massive reduction in cooling system power (to less than 5% of total system power, reducing the average PUE to 1.04) [25]. However, the additional material, installation, and testing complexity can add significant upfront costs to the cooling system installation [94].

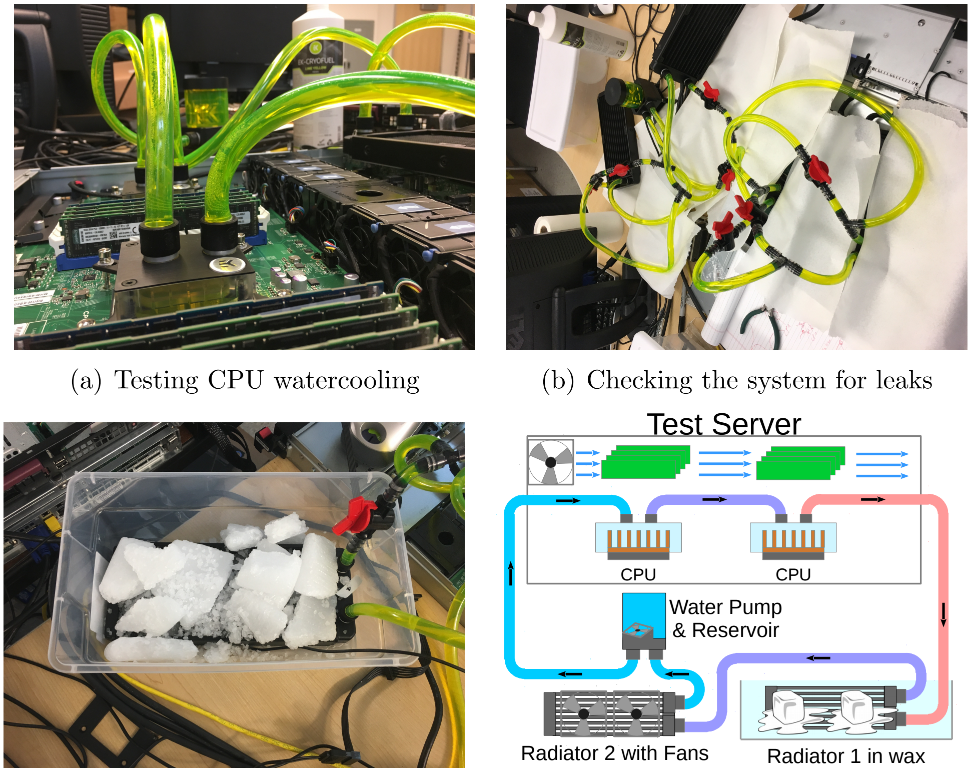
To solve this, we propose Thermal Gradient Transportation (TGT): a technique that leverages direct liquid cooling to move the high temperature difference needed to efficiently melt wax out of the server. Using the excellent cooling characteristics of water, we place a large container of paraffin wax in between the heat source (servers) and heat sink (cooling system) to absorb heat during the peak hours and release it during the off hours (Figure 5.1).

The benefits of TGT are two fold: first, we are able to deploy wax in much larger quantities to enable more energy storage, and thus a greater reduction in the peak cooling load. And second, we are able to coexist alongside a wider range of server architectures including those containing GPUs, while again greatly reducing the peak cooling load. This enable a datacenter to utilize a smaller cooling system to effectively cool a the same actual peak load that previously required a full sized and must more expensive cooling system.

5.2 Real System Tests

To validate the concept of TGT, we built a custom water cooling loop onto a real server and deployed a quantity of paraffin wax with the cooling loop. We use this test platform to investigate the performance of TGT on real hardware, and then use the knowledge gained to build our model for the scale out study in Section 5.4.

5.2.1 Real System Architecture



(a) Testing CPU watercooling

(b) Checking the system for leaks

(c) Adding wax to the container

(d) Test system diagram (water flows clockwise)

Figure 5.2: Constructing the test platform for real system tests. A water block is directly attached to each CPU socket (a). The system is filled with EK Cryofuel, a brand name mix of water, colored dye and a small amount of additives to inhibit corrosion and protect the system (b). Wax is held in a plastic container, submerging a radiator to transfer heat while keeping the wax and water separate (c). The final system provides direct water cooling to both CPUs, while leaving air cooling in place for the remaining components (d). Water travels from the pump, absorbing heat from both CPUs then storing or releasing heat in the wax at radiator 1 until passing through radiator 2 where heat is finally removed from the system before the water returns to the pump and begins the cycle again.

We started with an off-the-shelf Lenovo RD330 server, a dual socket, 1U low power server with two 6-core Intel Sandy Bridge Xeon CPUs and 144 GB of RAM. On this server, we removed the standard CPU heat sinks and added a pair of water blocks, two radiators, and one pump from an EKWB A240 water cooling kit (Figure 5.2(a)), as well as various valves and splitters that enable us to turn on and off flow control to different areas of the system (Figure 5.2(b)).

(Note that while we intentionally use an older server to minimize losses in the event of a catastrophic leak, the results and insights gained here are not architecture-specific and apply to Intel's newest offerings as well.)

The cooling system is filled with EKWB brand CryoFuel, a mix of water and propylene glycol with additives for coloring and to inhibit corrosion. (This mixture is not classified as a health or environmental hazard [27], and was disposed of properly afterwards.)

We conduct our real system experiments with the following liquid path setup (Figure 5.2): starting from the pump/reservoir unit, water is pumped through water blocks attached to both CPUs in series, then through radiator 1 and radiator 2 before returning to the pump/reservoir unit. The total loop contains approximately 800 ml of water coolant.

Radiator 1 was placed in a plastic container and surrounded with 2.0 kg of Rubitherm RT 28 HC wax, a commercial blend of paraffin wax with a melting temperature of 28 °C, a 214 J/g heat of fusion, and a solid density of 0.88 kg per liter [109]. Radiator 2 was left exposed to the air away from Radiator 1's container with two 120 mm box fans to emulate a cooling system removing heat before the liquid returns back to the CPU water blocks again.

Lastly, TEMPer1 USB temperature sensors were attached to the system including four submerged under the wax, one in each corner of Radiator 1's box, to measure thermal properties and a USB webcam wax was attached to the corner of the wax box to provide visual feedback.

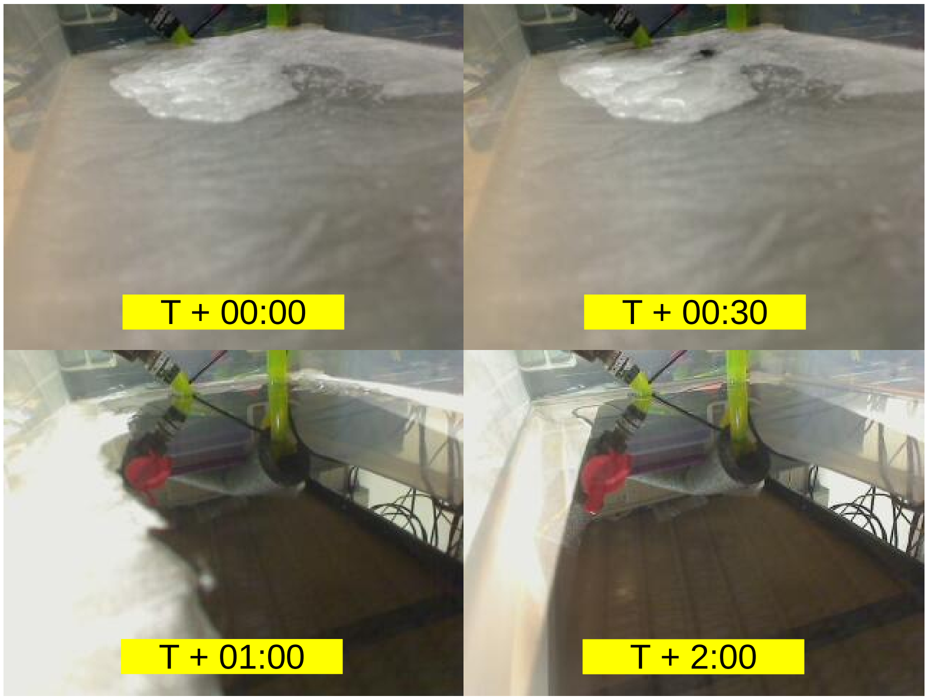


Figure 5.3: Still shots from a webcam monitoring the wax container as it melts by CPU heat transported via direct liquid cooling. The yellow-green tubes in the back carry water in and out. The radiator, black, becomes visible at the bottom of the wax container as wax melts. The melted wax has a reflective, glass-like appearance.

5.2.2 Real System Experiments

In this section, we analyze one experimental trace from our real system test platform. In this experiment, we run 24 instances of SPEC h264 (one per physical thread) looped for 12 hours to maximize load on the server, then run at idle for another 12 hours while the wax refreezes. This ensures all 2.0 kg of wax present in our test system melts in approximately 2 hours (Figure 5.3).

In Figure 5.4, we plot the mean temperature from four USB temperature sensors embedded in the wax container at various positions. In the figure, we identify four major features during the melting cycle: in the very beginning, for the first approximately 30 minutes, the temperature of the wax is being raised up to the melting temperature and a small amount of energy is stored as the wax quickly heats up (1). This sensible energy storage is the energy to raise the temperature without as phase

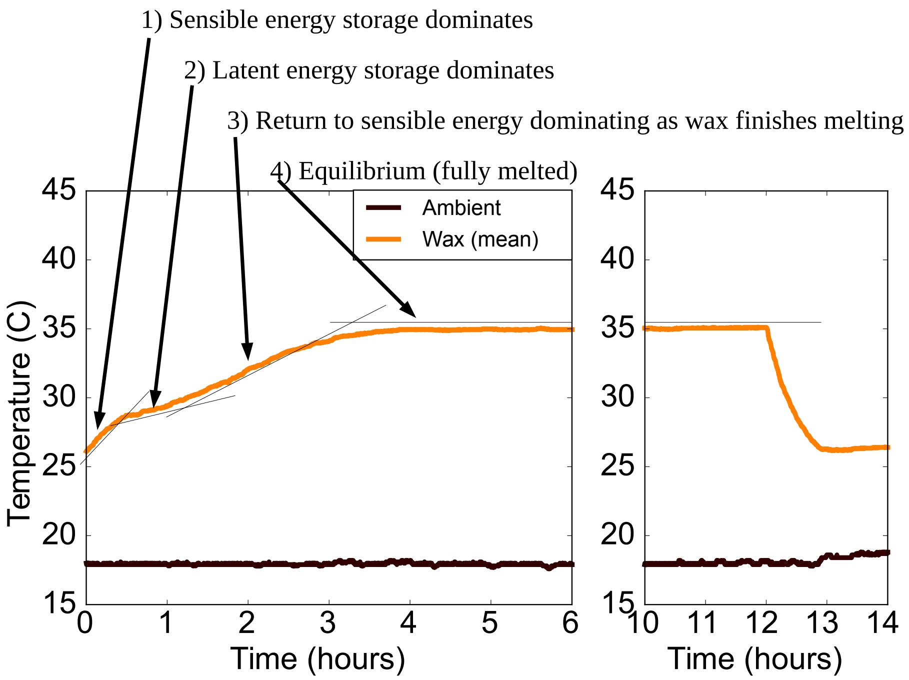


Figure 5.4: Measured temperature in the wax (mean of 4 sensors) while running the server at max load for 12 hours.

change.

Next as the the wax reaches the melting temperature (2), the rate of warming greatly decreases as significantly more heat is needed to melt the wax before the temperature can increase.

As the wax becomes partially melted, the wax enters a mixed state (3) where the wax closest to the heat source is fully melted while the wax furthest from the heat source is still only partially melted. The rate of temperature change gradually increases here as the last remainder of wax melts and sensible energy storage once again dominates, albeit slower than the first phase because the temperature difference is less.

Lastly, around 4 hours into the trace (4), the wax becomes fully melted and reaches its equilibrium temperature. That is: no additional latent energy can be stored because the wax is fully melted (or as fully melted as possible) and no additional sensible energy can be stored because the rate of energy absorbed from the radiator now equals the rate of energy lost through the wax container to the ambient air.

When the 12 hour load trace ends and the server returns to idle load, the wax refreezes and temperature is greatly reduced. The wax freezes much quicker than it melts, and without the clearly identifiable features, due to the location of the wax container (and also lack of insulation around it): while heating the wax container absorbs heat only from the radiator, but while cooling the wax releases heat back into the radiator and both heating and cooling constantly release heat into the ambient room.

(This excess cooling is beneficial as it can offer effectively free cooling if the wax container can be located outside of the artificially cooled datacenter environment, eg. outside the building. However, this is subject to great variation and in our scale out study we assume the rate of heating and cooling is symmetrical to provide a more conservative and realistic setup.)

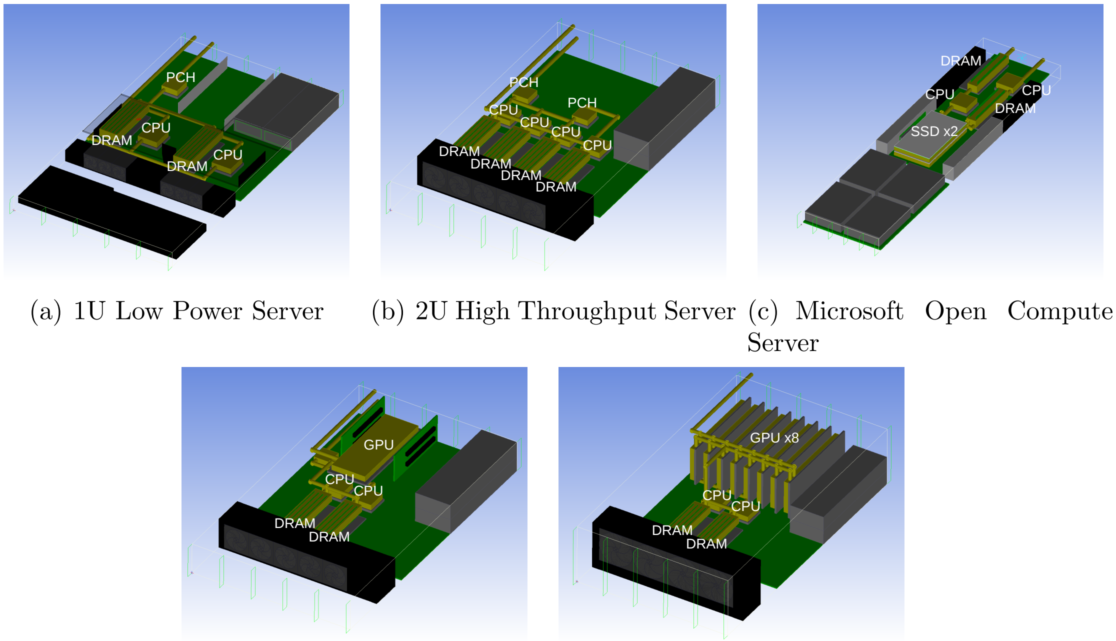
Real System Takeaways - From Figure 5.4, we can see that the best energy storage takes place in region 2, where the most wax is melting, and the beginning of region 3 before sensible energy storage dominates again.

When provisioning wax, we want to make sure that the server will remain in this desirable region to maximize energy storage for the whole duration of the peak load, not just for a single day but with sufficient time and cooling capacity to refreeze such that the next day's peak (and the next day after that, or however long the worst-case cooling load is expected to last) also remains in this range too.

5.3 Introduction

5.3.1 Scale Out Servers

For the scale out study, we consider a cluster each of 5 different server configurations. We model 3 domains of commodity server including a 1U lower power server, a



(a) 1U Low Power Server

(b) 2U High Throughput Server

(c) Microsoft Open Compute Server

(d) Single GPU Server

(e) Eight GPU Server

Figure 5.5: Test servers considered in this study. (a-c) are CPU only servers, while (d-e) add GPU accelerator cards for machine learning and other compute intensive workloads. Primary sources of active power, including the CPUs, DRAM, PCH(s), and GPU(s) (if present), are water cooled (indicated by gold colored water blocks and associated piping).

2U quad socket high throughput server, and a high density Microsoft Open Compute server evaluated in prior work [116]. We also consider two new servers that fill the empty airspace used for wax in prior work with GPU accelerators: a 2U dual socket, single-GPU machine and a 4U machine with 8 GPUs.

1U Low Power Server - The first server we consider is a standard 1U form factor commodity server (Figure 5.5(a)), modeled after the Lenovo RD330 server we perform our real system tests on. We then apply direct water cooling to the PCU, both CPUs, and all DRAM modules present in the server.

On the real system, we measured idle power consumption at 90 W and active power consumption at 185 W (excluding water cooling). Based on our measurements, we estimate that the proposed water cooling loop captures 28% of the heat produced at idle power consumption and 65% of the heat produced at peak active power consumption in the system. (The idle power captured is particularly low in this server because although total energy consumption is low, the server's power proportionality is also low and other components use a lot of power.)

2U High Throughput Server - The second server we consider is a 2U server modeled after the Sun X4470 (Figure 5.5(b)). It designed for compute heavy workloads with four CPUs E7-series processors. We apply direct water cooling onto each CPU and the corresponding DRAM modules, as well both PCH blocks. In this configuration, we estimate the idle power consumption to be 100 W and the peak power consumption to be 400 W before the PSU. Based on these estimates, the liquid cooling captures 60% of the idle power consumption and 90% of the active power consumption.

Microsoft Open Compute Server - The third server we consider for conventional workloads is modeled after the Microsoft Open Compute project server chassis (Figure 5.5(c)) [102]. It contains two Xeon E5-class CPUs, and two SSDs for data serving. We apply direct liquid cooling to both CPUs, their DRAM banks, and both SSDs capturing an estimated 74% of 100 watts of idle power and 80% of 300 watts of total active power.

Single GPU Server - The fourth server, and first server for GPU-accelerated workloads, is based on a commodity server featuring two Xeon E5-class CPUs and a single NVIDIA GPU (Figure 5.5(d)). We apply liquid cooling to both CPUs, the PCH, both DRAM banks and the GPU. Based on measurements of our GPU workloads in Section 5.3.3, we estimate the system idle power to be 120 W and the peak active power to be 470 W before the PSU. From this, we estimate that the liquid cooling loop captures 67% of the idle power and 78% of the peak active power.

Eight GPU Server - The fifth and last server we consider is a massive 4U machine with two Xeon E5-class GPUs managing 8 NVIDIA GPUs (Figure 5.5(e)). Based on measurements from our test system, we estimate idle power to be approx-

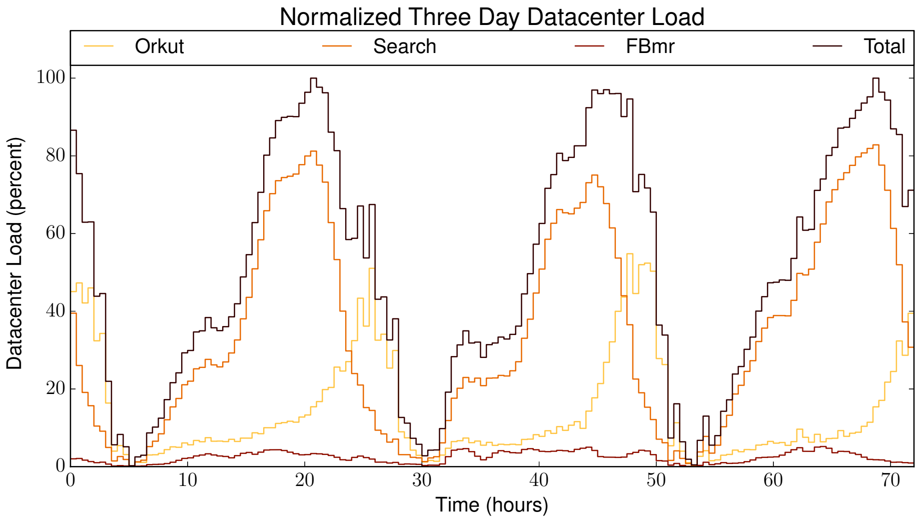


Figure 5.6: Three day datacenter workload trace from Google (normalized to peak throughput).

imately 260 W and peak active power for the workloads in this study to be 1400 W before the PSU. We apply direct water cooling to both CPUs and their DRAM banks, both PCHs, and all 8 GPUs to capture an estimated 85% of the idle power and 95% of the active power.

Liquid Cooling Parameters - We model all of the servers with an inlet water temperature of 20 °C based on typical datacenter water supply temperatures [58]. Although prior work [142] has proposed running this style of direct liquid cooling with exhaust hot water temperatures as high as 60 °C, in this work we constrain the maximum hot water temperatures to a more conservative 45 °C by adjusting the water flow rates. All servers receive the same water supply, and that supply is assumed to be a fixed flow rate. (This simplifies the pump design and cost while still ensuring that any individual server can run at an arbitrary load without overheating.)

5.3.2 Google Load Trace

The workload trace we run for our conventional server architectures is a three day workload trace from Google [123] (Figure 5.6). The trace contains three different workload types: Web Search, Orkut social networking traffic, and MapReduce (FBmr). This data was collected from the Google transparency report [65] and normalized as described by Skach et al. [116] to enable a quantitative comparison against prior work.

The three days selected represent a worst-case scenario [65], where datacenter use is extremely high for an extended period of time. (The third day is added on top of prior work [116] to also examine the day-after-day effects of wax melting temperatures, which become critical with the much larger wax quantities considered in this work. This effect is discussed in more detail in Section 5.4.)

5.3.3 Intelligent Personal Assistant Workload

To properly test our GPU-accelerated servers, we need a different set of workloads. For this we use Sirius Suite, a set of machine learning benchmarks for an IPA [49]. IPAs represent a new class of datacenter workload that are very computationally demanding, but also run well on accelerators such as GPUs [49].

This presents a first order problem for TTS, which relied on excess airspace within the server that the addition of a GPU takes away. Here, TGT offers another important opportunity: TGT enables thermal energy storage from not only the CPUs, but also GPUs without taking up any additional space inside of the servers.

To study the benefits of TGT with GPU-accelerated servers, we consider three major microservices in Sirius Suite: DNN-ASR, GMM and Stemmer (Table 5.1. We benchmark each workload on a K40 GPU (note that as with the CPUs from the real system test, the conclusions from the scale out study derived thereof are relevant for any GPU of similar thermal output).

DNN-ASR is an automatic speak recognition technique that leverages a deep neural network for speech-to-text translation. The Sirius implementation combines two approaches: Kaldi [101] and RWTH's RASR [110] to accomplish this task. We

Table 5.1: Runtime and power consumption measurements for microservices from Sirius Suite

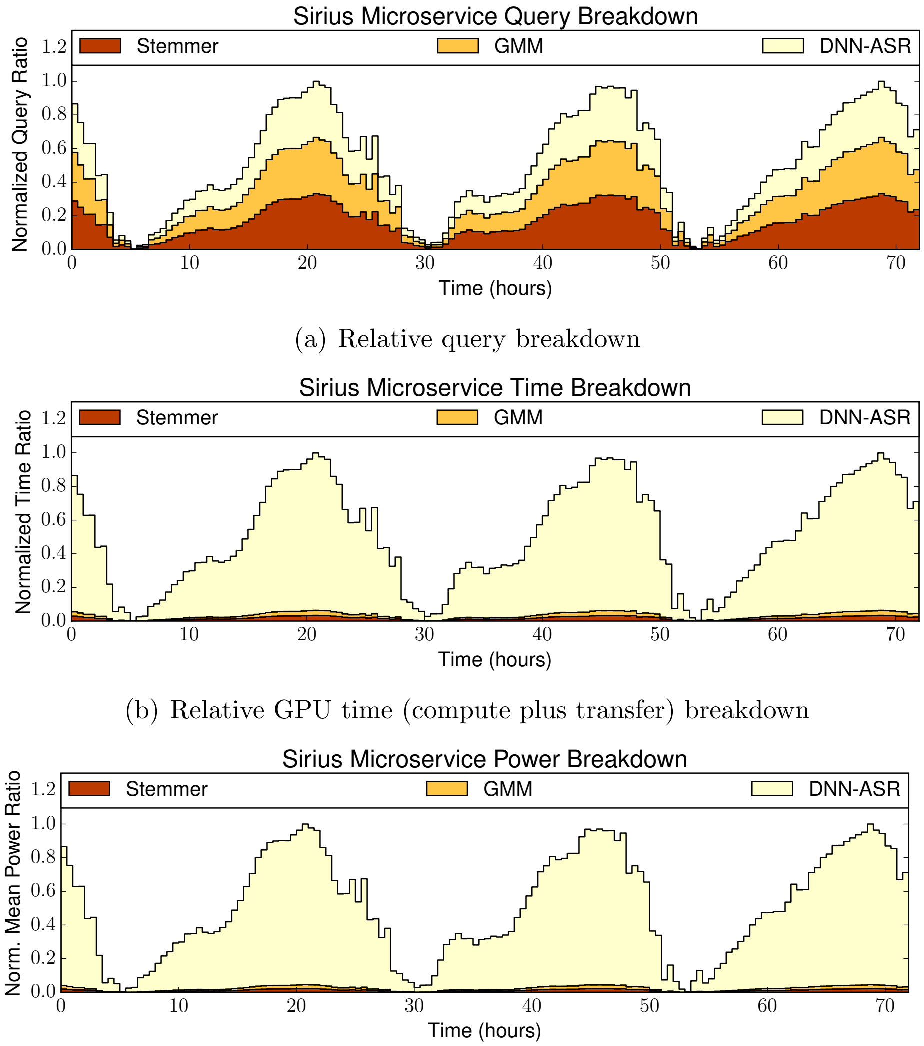
Workload	Runtime (ms)	GPU Power (W)
DNN-ASR	18.28	150
GMM	0.587	113
Stemmer	0.673	98

measured the average runtime of a DNN-ASR query at 18.28 ms on an NVIDIA K40 GPU, and an average (GPU-only) power consumption of 150 W using NVIDIA's System Management Interface (SMI) tool.

GMM is an alternative approach to automatic speech recognition that also benefits significantly from GPU acceleration. The Sirius Suite implementation of GMM uses a version of Sphinx, developed by Carnegie Mellon University [52]. Recent work [72] has shown the benefits of a tournament-style approach combining both GMM and a DNN, which we incorporate when constructing our workload trace from Sirius Suite. We measured the average runtime of a GMM query at 0.587 ms, and the average power consumption at 113 W.

Lastly, Stemmer is a GPU-accelerated version of the Porter Stemming algorithm [100], used to remove prefixes and/or suffixes and identify a root words. We measured the average runtime for a Stemmer query in Sirius Suite at 0.673 ms and the average power consumption at 98 W.

To create the load trace for Sirius, we begin with total load over the three day Google trace but subdivide it evenly between queries to each of our three Sirius Suite microservices (Figure 5.7(a)). This creates an even spread of queries however, as observed in Figure 5.7(b), the DNN-ASR implementation dominates overall GPU time (even including transfer overhead time for Stemmer and GMM). As a result, DNN-ASR also dominates the average power (Figure 5.7(c)) consumption too.



(a) Relative query breakdown

(b) Relative GPU time (compute plus transfer) breakdown

(c) Relative GPU power breakdown

Figure 5.7: Stacked charts of the three day trace for an intelligent personal assistant load trace based on Sirius Suite. Although queries are evenly distributed to each microservice, DNN-ASR dominates the compute time requirements and, both due to it's higher power consumption and higher runtime, power use as well.

5.3.4 Simulator

For the scale out study we use DCSim, an event-based query simulator for datacenter topologies used in several prior works [65, 116, 117, 118]. For each server architecture, we simulate a homogenous cluster of 1,000 servers on an isolated cooling loop with a single, unified wax storage unit. We assume the wax is contained in such a way that no thermal energy is gained or lost into the environment; rather, the water-cooled portions of the servers are responsible for providing all heat to melt the wax and the cooling system responsible for removing all heat.

We base our model of the wax melting off of experiments from our real test system, however some assumptions must be made to scale up the wax quantity significantly. We assume a symmetrical melting and cooling rates, mostly created by the previous assumption of thermally isolated wax. We also assume that the thermal transfer mechanism scales up linearly based upon the results of our real system tests. (This would most likely be water pipes and/or hot plates submerged in the wax, however the detailed implementation of a thermal transfer system into wax has been investigated extensively in prior work [62, 60, 51, 63, 59, 103] and is is outside the scope of this work).

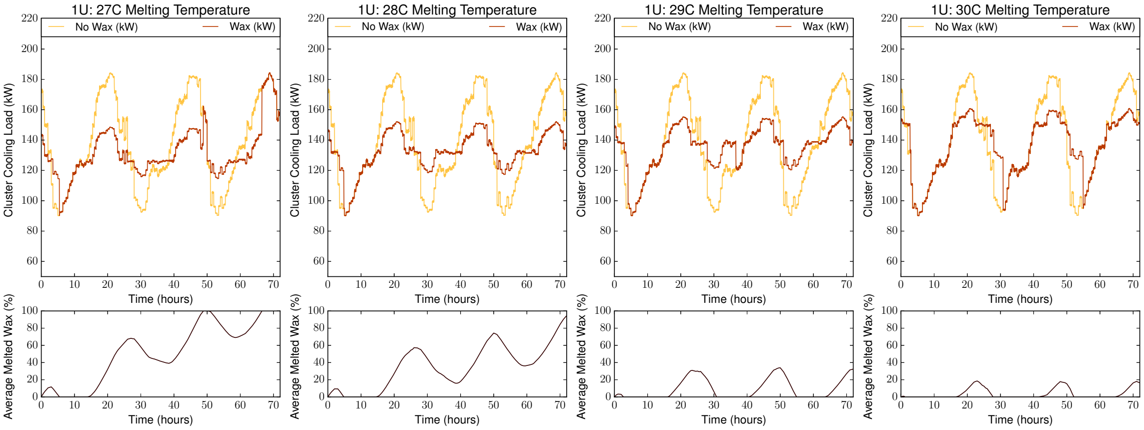
5.4 Evaluation

Using the simulation infrastructure described in Section 5.3, we now perform a scale out study for a cluster of 1,000 of each type of server. First, we dive into the results from the scale out study in Section 5.4.1, then discuss some implications and considerations for a real deployment in Section 5.4.2.

5.4.1 Scale Out Study

1U Server Deep Dive - In Figure 5.8, we plot the cooling load over three days for four different wax melting temperatures: 27 °C, 28 °C, 29 °C, and 30 °C. All four experiments were provisioned with 10 L of wax per server, however this is not necessarily a hard requirement: if less wax is used over the cycle, less than 10 L of wax can be deployed to reduce cost.

With a melting temperature of 27 °C (Figure 5.8 i), the wax melts quickly absorbing the most thermal energy during the first and most of the second days' peak



(i) (ii) (iii) (iv)

Figure 5.8: Cooling load for one thousand 1U low power servers, with and without wax, over two days at various melting temperatures with 10 L of wax per server. The lower the melting temperature is, the faster energy is absorbed however the melting temperature must be high enough to ensure there is sufficient time in the off hours to allow the wax to fully refreeze. If there is not (i), the wax may eventually become fully melted and no longer provide a benefit. However, if the high load is only expected to a limited time period, then wax with a melting temperature that lasts until the end without fully refreezing may be tolerated and still provide excellent benefit (ii) as long as there will be sufficient downtime between reoccurences. Higher melting temperatures can (iii, iv) can provide good benefit while using less wax, or leave more wax capacity in reserve for emergencies.

hours. However, this melting temperature is too low as during the off hours the wax does not have enough time to fully refreeze between peaks. Thus as the cluster enters the second day's peak hours, nearly 50% of the wax is still melted and before the end of the second day the wax is fully melted and can no longer absorb energy (this occurs when the 'wax' trace jumps up to meet the 'no wax' trace, around hour 47). By day 3, the wax runs out even earlier and there is no reduction in the peak cooling load that day.

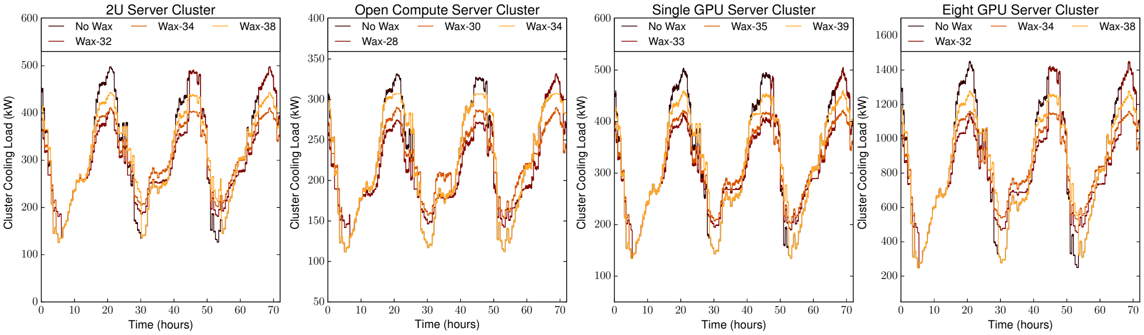
In contrast, with a melting temperature of 28 °C (Figure 5.8 ii) the 1U server achieves a peak cooling load reduction across all three days even though the wax again does not completely refreeze during the off hours. This is because while at the start of day two and three's peak hours some of the wax is already melted, 10 L of wax is enough to prevent the cluster from running out of thermal energy storage capacity before the end of day three. By design, these are the three worst case days [65] so this is a sustainable operating level with 10 L of wax as long as during the next few days the wax is allowed to refreeze before another such worst case pattern is encountered.

A 27 °C melting temperature, or even lower, could potentially become usable by adding more wax to accomplish a result similar to 28 °C however there are diminishing returns: the more wax that must be refrozen, the more time between peak traffic loads must exist and/or the lower those between peak traffic loads there must be.

In Figure 5.8 iii and iv, we plot melting temperatures of 29 °C and 30 °C. Both of these melting temperatures allow sufficient time for all of the wax to melt between peak hours, and are thus indefinitely sustainable for this load pattern. They require less wax as a result of the higher melting temperature and complete refreezing between each cycle (no more than 4 L and 3 L per server, respectively), but store less thermal energy too.

Over the whole 3 day trace, 27 °C provides no benefit to the peak cooling load, 28 °C reduces the peak cooling load by 17.5%, 29 °C reduces the peak cooling load by 15.8%, and 30 °C reduces the peak cooling load by 12.8% with continuously diminishing returns above there. Using the 3 day sustainable temperature, this corresponds to a 17.5% smaller cooling system (1.97x higher than prior work [116]) or a 15.9% smaller cooling system with the indefinitely sustainable peak cooling load reduction. Alternatively, this reduction can be used to install 18.7% to 21.1% more servers under the same cooling budget (approximately 1011 to 1140 additional servers per megawatt of critical power).

Overview of Other Servers - In Figure 5.9, we plot condensed versions of the previous plot for each additional server: the 2U High Throughput server (Figure 5.9 i), the Open Compute server (Figure 5.9 ii), the Single GPU server (Figure 5.9 iii), and



(i)

(ii)

(iii)

(iv)

Figure 5.9: Cooling load for the 2U, Open Compute, Single GPU and Eight GPU servers, with three different melting temperatures of wax (one above, one the produces the best cooling load reduction, and one below). The 2U server achieves a peak cooling load reduction of 17.4% with a melting temperature of 34 °C. The Open Compute server achieves a peak cooling load reduction of 12.5% with a melting temperature of 30 °C. The Single GPU server achieves a peak cooling load reduction of 16.4% with a melting temperature of 35 °C. The Eight GPU server achieves a peak cooling load reduction of 20.0% with a melting temperature of 34 °C.

the Eight GPU server (Figure 5.9 iv). Each configuration contains 20 L of wax per server except for the Eight GPU server, which is provisioned with 50 L of wax per server because of it's significantly higher power consumption per server.

The 2U High Throughput server achieves the best peak cooling load reduction with a melting temperature of 34 °C. Above or below this temperature, the load reduction is either unsustainable for all three days, or melted too late and stored less energy as a result. In the 2U and other servers, the ideal melting temperatures do not exactly correlate to total power consumption: the rate at which the wax melts is a function of the temperature of the hot water carrying thermal energy from the servers to the wax. Although the water is generally warmer for higher power servers in this set of experiments, the flow rate is also adjusted accordingly to maintain a safe set of both liquid and component temperatures.

With a melting temperature of 34 °C, the peak cooling load for the cluster of 2U servers was reduced by 17.4% (1.45x prior work [116]). This corresponds to a

17.4% smaller cooling system, or 21.0% more servers under the same cooling budget (approximately 420 of this server configuration per megawatt of critical power in the datacenter).

For the Open Compute server cluster, a melting temperature of 30 °C achieved the best peak cooling load reduction of 12.5%. This enables a 12.5% smaller cooling system (1.5x higher than prior work [116]) or 14.2% more servers under the same cooling budget (approximately 430 more Open Compute servers per MW of critical power).

Moving on to the IPA workload, the addition of the GPU creates a new challenge that prior work [116] could not address at all: the server design where there is no space remaining to place wax inside of the server. Here, the results are similar as seen so far because TGT enables us to achieve a very good peak cooling load reductions without excess air space.

The Single GPU server cluster achieves a 16.4% peak cooling load reduction with 20 L of wax per server and a melting temperature of 35 °C (corresponding the same reduction in cooling system size or 19.6% more servers–up to 392 per megawatt of critical power– under the same cooling budget). The metric of additional servers is particularly important in a retrofit scenario where the original cooling system, which typically lasts several generations of servers [14], may not have been originally provisioned for the thermal energy levels produced by GPU-equipped servers.

The Eight GPU server cluster, with it's enormous power consumption, achieves up to a 20% reduction in peak cooling load but requires nearly 50 L of wax per server with a melting temperature of 34 °C. This corresponds to an equivalent reduction in cooling system size, or 25.0% more servers (up to 172 more servers containing 1379 GPUs) in the same budget. Although it will vary for different server and workload configurations, both of these load levels and melting temperatures for the Single GPU and Eight GPU clusters are indefinitely sustainable.

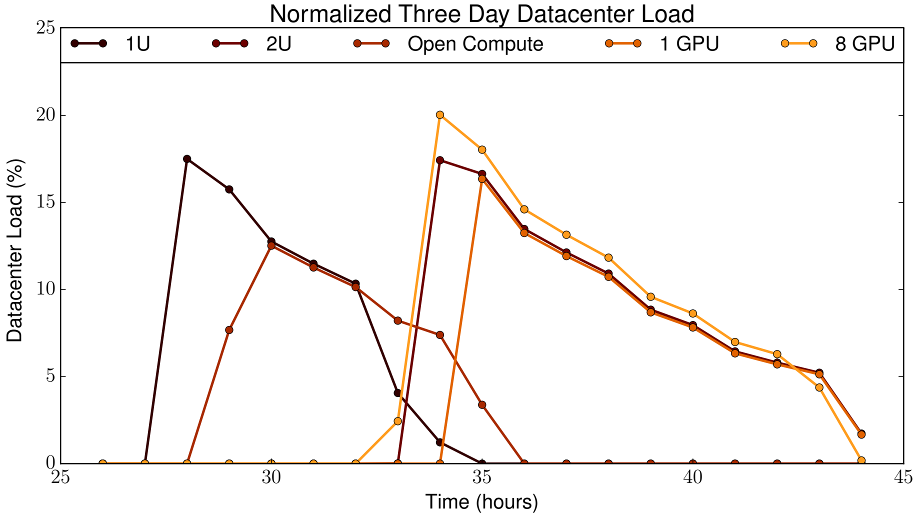


Figure 5.10: Sweeping cooling load reduction across wax melting temperatures. As soon as the melting temperature reaches a threshold where melting and freezing can be sustained for three days, there is a big jump in peak cooling load reduction followed by slowly decreasing benefit. The greatest benefit is not always sustainable for an arbitrary amount of time, however the next melting wax configuration above it typically is.

Sweeping the Melting Temperature - In Figure 5.10, we sweep the wax melting temperature for each server cluster from 26 °C to 44 °C and plot the resulting peak cooling load reduction.

In general, all five servers follow a similar trend: they provide zero (or near-zero) benefit until the peak load reduction can be sustained for at least the three day load trace. Then, each cluster sees decreasing peak load reduction thereafter until the melting temperature becomes too high and no significant wax melts anymore.

Also, while there appears a slight trend with the lowest power servers (the 1U and Open Compute servers) benefiting from the lower melting temperatures, this is an artifact of the scale out methodology. The actual relationship, as previously noted, depends on the water temperatures which are themselves a function of both power output and flow rate. (Note that the Eight GPU machine, while have almost 3x higher power consumption than the 2U server, also has the same best melting temperature of 34 °C.)

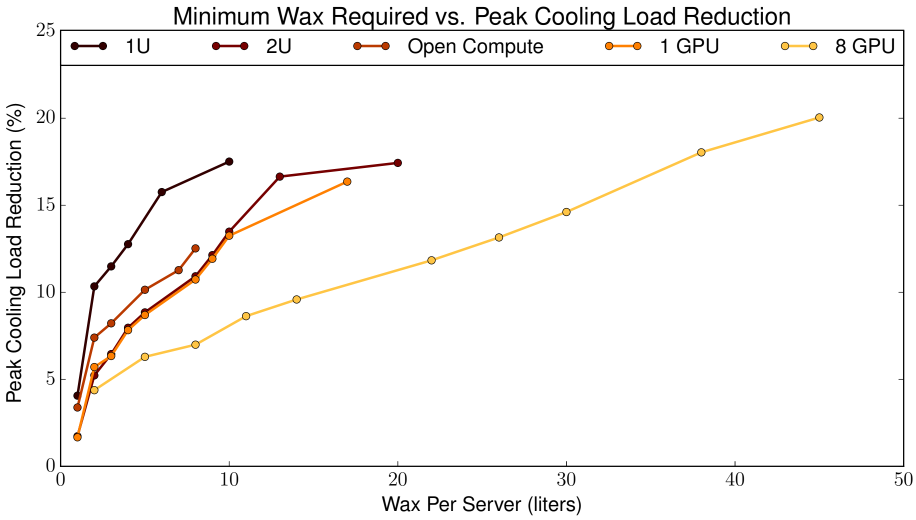


Figure 5.11: Peak cooling load reduction vs. the minimum amount of wax required to obtain that peak cooling load by varying the melting temperature. The first 5-10% peak reduction requires the least wax, with the amount of wax per percent reduction increasing significantly thereafter. Additionally, the higher the peak power output of the server, the more wax it requires for even the same peak load reduction as more thermal energy must be stored.

Examing the Quantity of Wax - In Figure 5.11, we plot data in a different way: with the peak cooling load reduction as a function of the minimum amount of wax per server (rounded up to the nearest liter) required to achieve that reduction.

Here, we can see the effect of the power level on each server: The 1U server cluster requires by far the least amount of wax to achieve the same cooling load reduction while the Eight GPU server requires by far the most as expected: the wax has a fixed amount of thermal energy capacity per unit of volume, and once that is exhausted the only way to achieve more thermal energy storage is by adding wax.

This figure also shows the diminishing returns of adding wax. Due to the shape of the Google trace's diurnal cycle, closer to a sine wave than a square wave, the greatest peak cooling load reduction per unit of stored energy occurs at the peak of the trace and increases superlinearly as as the total peak cooling load reduction increases.

We can observe this effect in each server, for example the 1U server requires just 1 additional liter of wax to go from a 4.1% reduction (1 L) to a 10.3% reduction (2 L), while it requires six times as much wax again to reduce the peak cooling by 4x at 15.8% (6 L). Similarly, the Eight GPU server requires over three times the wax to increase the peak cooling load reduction 2x from a 6.3% (5 L) to a 13.1% (26L), or seven times the wax to for a 3x increase to 20.0% (45 L).

5.4.2 Discussion of Deployment Considerations

In this section, we discuss some of the considerations for deploying TGT in a datacenter including the relationship between TGT and prior work on virtualized melting temperatures, TGT's tolerance for load variation, and the security implications of TGT.

5.4.2.1 TGT and Virtual Melting Temperatures

Virtual Melting Temperature (VMT) [118] is technique from prior work that proposed to schedule workloads in such a way as to maximize the amount of thermal energy stored in wax. VMT was designed for a TTS-style architecture where each server has its own reservoir of wax to store energy in, and proposed two techniques to schedule workloads: Thermal Aware VMT and Wax Aware VMT.

Thermal Aware VMT proposes that, when the average temperature in a cluster of mixed job types is not hot enough to melt wax, to group the hot jobs together in the same servers to melt wax in that subset of servers. This technique is applicable with TGT, but must be implemented slightly differently: as the water temperature at the wax is determined by mixing the return hot water from every server in same cooling loop, all of those servers must be treated together as the hot group and a separate loop of servers designated the cold group. This provides the same functionality as Thermal Aware VMT, however it offers significantly less flexibility to the datacenter operator in determine the hot group and cold group sizes as they may only be adjusted at the granularity of the number of servers connected to one discrete cooling loop.

Wax Aware VMT begins with the same premise as Thermal Aware VMT, but also tracks how much of the wax is melted over time and dynamically resizes the hot group and cold group to maximize thermal energy storage even further. This technique is theoretically possible with TGT, but requires the ability to move servers from one cooling loop to the other on the fly. A system of networked flow valves could perform this task, however the added cost and complexity likely outweighs the benefits that Wax Aware VMT offers over Thermal Aware VMT.

Neither of the workload mixtures we consider in this work benefit strongly from VMT: both contain workloads that, although especially in the case of the GPU-accelerated Sirius Suite workload, have unbalanced per-query and/or overall execution time, they are relatively balanced power-wise. VMT offers the greatest benefit when workloads have strongly unbalanced power usage and as a result, plus the aforementioned implementation difficulties, we do not attempt to quantify the benefits of TGT with VMT in this work.

5.4.2.2 Load Variation Tolerance

Another downside of wax placement inside the server is it's limited capability to handle unpredictable workloads. Because the wax quantity for TTS [116] is fixed based upon available space inside of the server, the only way to increase the margin of error for thermal energy storage is by decreasing the amount of wax melted during typical cycles and leaving the extra, unmelted wax to account for longer and/or hotter peak hours.

Wax Aware VMT further proposes a partial solution by rebalancing workloads to leverage unmelted wax in other servers, however this depends on both (a) the workloads being sufficiently portable that rearranging them is practical and (b) that there are other servers with unmelted wax available [118].

TGT on the other hand offers a much simpler solution: simply add more wax. Regardless of the type of workload or architecture of the servers present, TGT enables arbitrary quantities of wax. This means that even for highly unpredictable workloads, eg. some days with extra long peak hours or very short off hours, sufficient wax may be deployed to absorb energy for 24 hours, 48 hours, or even longer if needed so long as at some time in the future cooling system capacity will be available to refreeze the wax.

5.4.2.3 Security Implications

Recent work [56] has proposed thermal side channels attacks against multi-tenant datacenters, and identified TTS as a possible attack vector [38].

As long as servers share cooling resources within a datacenter this vector cannot be entirely, but at a minimum TGT greatly mitigates the risk posed by TTS by sharing thermal energy storage capacity between many machines (versus each server having a discreet, highly limited supply). This removes the ability for an attacker to target individual servers, instead requiring a potential attacker to exploit a large portion of the entire cluster.

Furthermore, because TGT offers the opportunity to deploy significantly greater quantities of wax than TTS, in parallel with or in addition to adding fault tolerance, excess thermal energy storage capacity can be used mitigate or ”ride out” unexpected thermal side channel attacks.

5.5 Conclusion

In this chapter we introduced Thermal Gradient Transportation (TGT), a new technique that leverages direct water cooling and phase change materials (PCMs) such as paraffin wax to store energy and decrease the peak thermal load of a datacenter. By storing thermal energy during the hours of peak utilization during the day, and releasing it during the off hours, we enable up to a 20% smaller–and thus more affordable–cooling system or up to 25% more servers to be deployed without increasing the size of the cooling system.

Compared to prior work in this area, TGT with paraffin wax offers a competitively affordable and energy efficient alternative to chilled water and water ice tanks for thermal energy storage. Against techniques that do use paraffin wax, TGT offers an up to 1.97x better reduction in peak cooling load over prior work. TGT also enables thermal energy storage on new architectures and workloads such as GPU-accelerated servers for machine learning and intelligent personal assistant (IPA) applications where prior work will not function at all.

CHAPTER VI

Conclusion

Datacenters have been rapidly growing in size, number, and complexity for over a decade now with no signs of slowing down. Inside of these datacenters, the quest for efficiency–both for ecological and cost saving reasons–has far reaching implications. After IT equipment, a datacenter's cooling system is often one of the most power hungry, expensive, and inefficient parts of the datacenter in no small part due to the unbalanced and diurnal nature of user-facing workloads that make up most datacenter's traffic.

In this dissertation, we discussed and analyzed three new technqiues to reduce the size of datacenter cooling systems to increase their utilization, reduce costs, and (hopefully) encourage more energy efficient deployments. The first technique, Thermal Time Shifting (TTS), temporarily decouples the cooling load of a datacenter from work done. Using a phase change material (PCM), TTS allows a datacenter to temporarily store thermal energy during periods of high utilization and release it during periods of low utilization to better balance the cooling load.

With Virtual Melting Temperatures (VMT), we investigate a problem that arises directly from TTS: mixed workloads with unbalanced power profiles. We propose two algorithms to dynamically rebalance such workloads and effectively virtualize the melting temperature of the PCM, storing energy even when the PCM would not normally melt.

Lastly, with Thermal Gradient Transportation (TGT), we leverage highly efficient but expensive direct water cooling to bring the thermal energy out of the server to PCM remotely. This enables not only more PCM, and thus more thermal energy storage, but also handles high power accelerators such as GPUs, a brand new class of datacenter architecture.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Data center cooling market size by application, May 2016.

[2] Upload files to Google Drive. https://goo.gl/lz3OKG, 2016. Online; accessed 15-Nov-2016.

[3] International energy outlook 2017. Technical report, USDOE Energy Information Administration (EIA), Washington, DC (United States). Office of Energy Analysis, 2017.

[4] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[5] D. Abts, M. R. Marty, P. M. Wells, P. Klausler, and H. Liu. Energy proportional datacenter networks. In *ACM SIGARCH Computer Architecture News*, volume 38, pages 338–347. ACM, 2010.

[6] M. Alizadeh, T. Edsall, S. Dharmapurikar, R. Vaidyanathan, K. Chu, A. Fingerhut, F. Matus, R. Pan, N. Yadav, G. Varghese, et al. Conga: Distributed congestion-aware load balancing for datacenters. In *ACM SIGCOMM Computer Communication Review*, volume 44, pages 503–514. ACM, 2014.

[7] D. G. Andersen and S. Swanson. Rethinking flash in the data center. *IEEE micro*, 30(4):52–54, 2010.

[8] A. S. Andrae and T. Edler. On global electricity usage of communication technology: trends to 2030. *Challenges*, 6(1):117–157, 2015.

[9] P. Apparao, R. Iyer, X. Zhang, D. Newell, and T. Adelmeyer. Characterization & analysis of a server consolidation benchmark. In *Proceedings of the fourth ACM SIGPLAN/SIGOPS international conference on Virtual execution environments*, pages 21–30. ACM, 2008.

[10] M. Avgerinou, P. Bertoldi, and L. Castellazzi. Trends in data centre energy consumption under the european code of conduct for data centre energy efficiency. *Energies*, 10(10):1470, 2017.

[11] C. A. Balaras, J. Lelekis, E. G. Dascalaki, and D. Atsidaftis. High performance data centers and energy efficiency potential in greece. *Procedia environmental sciences*, 38:107–114, 2017.

[12] L. A. Barroso, J. Dean, and U. Holzle. Web search for a planet: The google cluster architecture. *IEEE micro*, 23(2):22–28, 2003.

[13] L. A. Barroso and U. Hölzle. The case for energy-proportional computing. 2007.

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

[15] A. Beloglazov, J. Abawajy, and R. Buyya. Energy-aware resource allocation heuristics for efficient management of data centers for cloud computing. *Future generation computer systems*, 28(5):755–768, 2012.

[16] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, M. Q. Dang, and K. Pentikousis. Energy-efficient cloud computing. *The computer journal*, 53(7):1045–1051, 2010.

[17] A. Burdick. *Strategy Guideline: Accurate Heating and Cooling Load Calculations*. US Department of Energy, Energy Efficiency & Renewable Energy, Building Technologies Program, 2011.

[18] R. Buyya, A. Beloglazov, and J. Abawajy. Energy-efficient management of data center resources for cloud computing: a vision, architectural elements, and open challenges. *arXiv preprint arXiv:1006.0308*, 2010.

[19] V. Cardellini, M. Colajanni, and P. S. Yu. Dynamic load balancing on web-server systems. *IEEE Internet computing*, 3(3):28–39, 1999.

[20] A. M. Caulfield, L. M. Grupp, and S. Swanson. Gordon: using flash memory to build fast, power-efficient clusters for data-intensive applications. *ACM Sigplan Notices*, 44(3):217–228, 2009.

[21] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 1–14. ACM, 2007.

[22] T. Chen, Z. Du, N. Sun, J. Wang, C. Wu, Y. Chen, and O. Temam. Diannao: A small-footprint high-throughput accelerator for ubiquitous machine-learning. *ACM Sigplan Notices*, 49(4):269–284, 2014.

[23] R. Chitta. *Kernel-based clustering of big data*. PhD thesis, Michigan State University, 2015.

[24] I. Chowdhury, R. Prasher, K. Lofgreen, G. Chrysler, S. Narasimhan, R. Mahajan, D. Koester, R. Alley, and R. Venkatasubramanian. On-chip cooling by superlattice-based thin-film thermoelectrics. *Nature Nanotechnology*, 4(4):235–238, 2009.

[25] M. P. David, M. Iyengar, P. Parida, R. Simons, M. Schultz, M. Gaynes, R. Schmidt, and T. Chainer. Experimental characterization of an energy efficient chiller-less data center test facility with warm water cooled servers. In *Semiconductor Thermal Measurement and Management Symposium (SEMI-THERM), 2012 28th Annual IEEE*, pages 232–237. IEEE, 2012.

[26] D. M. Dias, W. Kish, R. Mukherjee, and R. Tewari. A scalable and highly available web server. In *Compcon'96.'Technologies for the Information superhighway'Digest of Papers*, pages 85–92. IEEE, 1996.

[27] EKWB. EK-ekoolant material safety data sheet, September 2011.

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

[29] M. Ellsworth, L. Campbell, R. Simons, M. Iyengar, R. Schmidt, and R. Chu. The evolution of water cooling for ibm large server systems: Back to the future. In *Thermal and Thermomechanical Phenomena in Electronic Systems, 2008. ITHERM 2008. 11th Intersociety Conference on*, pages 266–274. IEEE, 2008.

[30] M. J. Ellsworth and M. K. Iyengar. Energy efficiency analyses and comparison of air and water cooled high performance servers. In *ASME 2009 InterPACK Conference collocated with the ASME 2009 Summer Heat Transfer Conference and the ASME 2009 3rd International Conference on Energy Sustainability*, pages 907–914. American Society of Mechanical Engineers, 2009.

[31] S. Fan, S. M. Zahedi, and B. C. Lee. The computational sprinting game. In *Proceedings of the Twenty-First International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 561–575. ACM, 2016.

[32] X. Fan, W.-D. Weber, and L. A. Barroso. Power provisioning for a warehouse-sized computer. In *ACM SIGARCH Computer Architecture News*, volume 35, pages 13–23. ACM, 2007.

[33] M. E. Femal and V. W. Freeh. Boosting data center performance through non-uniform power allocation. In *Autonomic Computing, 2005. ICAC 2005. Proceedings. Second International Conference on*, pages 250–261. IEEE, 2005.

[34] M. Ferdman, A. Adileh, O. Kocberber, S. Volos, M. Alisafaee, D. Jevdjic, C. Kaynak, A. D. Popescu, A. Ailamaki, and B. Falsafi. Clearing the clouds: a study of emerging scale-out workloads on modern hardware. In *ACM SIGPLAN Notices*, volume 47, pages 37–48. ACM, 2012.

[35] Fusion-io products. https://goo.gl/iGudU0, 2014. Online; accessed 01-Aug-2014.

[36] E. Gal and S. Toledo. Algorithms and data structures for flash memories. *ACM Computing Surveys (CSUR)*, 37(2):138–163, 2005.

[37] T. Gao, M. David, J. Geer, R. Schmidt, and B. Sammakia. Experimental and numerical dynamic investigation of an energy efficient liquid cooled chiller-less data center test facility. *Energy and buildings*, 91:83–96, 2015.

[38] X. Gao, Z. Xu, H. Wang, L. Li, and X. Wang. Reduced cooling redundancy: A new security vulnerability in a hot data center. 01 2018.

[39] D. Garday and J. Housley. Thermal storage system provides emergency data center cooling. *White Paper Intel Information Technology, Intel Corporation*, 2007.

[40] A. Ghiasi, R. Baca, G. Quantum, and L. Commscope. Overview of largest data centers. In *Proc. 802.3 bs Task Force Interim meeting*, 2014.

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

[42] Í. Goiri, T. D. Nguyen, and R. Bianchini. Coolair: Temperature-and variation-aware management for free-cooled datacenters. In *ACM SIGPLAN Notices*, volume 50, pages 253–265. ACM, 2015.

[43] S. Govindan, A. Sivasubramaniam, and B. Urgaonkar. Benefits and limitations of tapping into stored energy for datacenters. In *Computer Architecture (ISCA), 2011 38th Annual International Symposium on*, pages 341–351. IEEE, 2011.

[44] S. Govindan, D. Wang, A. Sivasubramaniam, and B. Urgaonkar. Leveraging stored energy for handling power emergencies in aggressively provisioned datacenters. In *ACM SIGPLAN Notices*, volume 47, pages 75–86. ACM, 2012.

[45] S. Govindan, D. Wang, A. Sivasubramaniam, and B. Urgaonkar. Aggressive datacenter power provisioning with batteries. *ACM Transactions on Computer Systems (TOCS)*, 31(1):2, 2013.

[46] S. Greenberg, E. Mills, B. Tschudi, P. Rumsey, and B. Myatt. 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, 2006.

[47] D. Hale, M. Hoover, and M. ONeill. Phase-change materials handbook. 1972.

[48] J. Hauswald, Y. Kang, M. A. Laurenzano, Q. Chen, C. Li, T. Mudge, R. G. Dreslinski, J. Mars, and L. Tang. Djinn and tonic: Dnn as a service and its implications for future warehouse scale computers. In *Computer Architecture (ISCA), 2015 ACM/IEEE 42nd Annual International Symposium on*, pages 27–40. IEEE, 2015.

[49] J. Hauswald, M. A. Laurenzano, Y. Zhang, C. Li, A. Rovinski, A. Khurana, R. G. Dreslinski, T. Mudge, V. Petrucci, L. Tang, et al. Sirius: An open end-to-end voice and vision personal assistant and its implications for future warehouse scale computers. In *ACM SIGPLAN Notices*, volume 50, pages 223–238. ACM, 2015.

[50] J. L. Henning. Spec cpu2006 benchmark descriptions. *ACM SIGARCH Computer Architecture News*, 34(4):1–17, 2006.

[51] L. Huang, M. Petermann, and C. Doetsch. Evaluation of paraffin/water emulsion as a phase change slurry for cooling applications. *Energy*, 34(9):1145–1155, 2009.

[52] D. Huggins-Daines, M. Kumar, A. Chan, A. W. Black, M. Ravishankar, and A. I. Rudnicky. Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 1, pages I–I. IEEE, 2006.

[53] H. Ibrahim, A. Ilinca, and J. Perron. Energy storage systems–characteristics and comparisons. *Renewable and Sustainable Energy Reviews*, 12(5):1221–1250, 2008.

[54] C. G. C. Index. Forecast and methodology, 2015-2020 white paper, 2016.

[55] Intel. Intel server board S3420GP server system SR1630GP server board SR1630HGP server chassis SC5650UP calculated MTBF estimates. Intel White Paper, 2009. Online; accessed 01-Oct-2017.

[56] M. A. Islam, S. Ren, and A. Wierman. Exploiting a thermal side channel for power attacks in multi-tenant data centers. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pages 1079–1094. ACM, 2017.

[57] M. A. Islam, X. Ren, S. Ren, A. Wierman, and X. Wang. A market approach for handling power emergencies in multi-tenant data center. In *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 432–443. IEEE, 2016.

[58] M. Iyengar, M. David, P. Parida, V. Kamath, B. Kochuparambil, D. Graybill, M. Schultz, M. Gaynes, R. Simons, R. Schmidt, et al. Server liquid cooling with chiller-less data center design to enable significant energy savings. In *Semiconductor Thermal Measurement and Management Symposium (SEMI-THERM), 2012 28th Annual IEEE*, pages 212–223. IEEE, 2012.

[59] M. Jaworski. Thermal performance of heat spreader for electronics cooling with incorporated phase change material. *Applied Thermal Engineering*, 35:212–219, 2012.

[60] S. P. Jesumathy, M. Udayakumar, and S. Suresh. Heat transfer characteristics in latent heat storage system using paraffin wax. *Journal of mechanical science and technology*, 26(3):959–965, 2012.

[61] V. Jimenez, F. Cazorla, R. Gioiosa, E. Kursun, C. Isci, A. Buyuktosunoglu, P. Bose, and M. Valero. Energy-aware accounting and billing in large-scale computing facilities. *IEEE Micro*, 31(3):60–71, 2011.

[62] B. Kamkari and H. J. Amlashi. Numerical simulation and experimental verification of constrained melting of phase change material in inclined rectangular enclosures. *International Communications in Heat and Mass Transfer*, 88:211–219, 2017.

[63] R. Kandasamy, X.-Q. Wang, and A. S. Mujumdar. Transient cooling of electronics using phase change material (pcm)-based heat sinks. *Applied thermal engineering*, 28(8-9):1047–1057, 2008.

[64] S. Kanev, K. Hazelwood, G.-Y. Wei, and D. Brooks. Tradeoffs between power management and tail latency in warehouse-scale applications. *Power*, 20:40, 2014.

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

[66] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[67] M. A. Laurenzano, Y. Zhang, L. Tang, and J. Mars. Protean code: Achieving near-free online code transformations for warehouse scale computers. In *Microarchitecture (MICRO), 2014 47th Annual IEEE/ACM International Symposium on*, pages 558–570. IEEE, 2014.

[68] D. B. LD and P. V. Krishna. Honey bee behavior inspired load balancing of tasks in cloud computing environments. *Applied Soft Computing*, 13(5):2292–2303, 2013.

[69] J. Leverich, M. Monchiero, V. Talwar, P. Ranganathan, and C. Kozyrakis. Power management of datacenter workloads using per-core power gating. *IEEE Computer Architecture Letters*, 8(2):48–51, 2009.

[70] H. Li and A. Michael. Intel motherboard hardware v2.0. Open Compute Project, 2011.

[71] K. Li, G. Xu, G. Zhao, Y. Dong, and D. Wang. Cloud task scheduling based on load balancing ant colony optimization. In *Chinagrid Conference (ChinaGrid), 2011 Sixth Annual*, pages 3–9. IEEE, 2011.

[72] S. Liu and K. C. Sim. On combining dnn and gmm with unsupervised speaker adaptation for robust automatic speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 195–199. IEEE, 2014.

[73] Z. Liu, Y. Chen, C. Bash, A. Wierman, D. Gmach, Z. Wang, M. Marwah, and C. Hyser. Renewable and cooling aware workload management for sustainable data centers. In *ACM SIGMETRICS Performance Evaluation Review*, volume 40, pages 175–186. ACM, 2012.

[74] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew. Geographical load balancing with renewables. *ACM SIGMETRICS Performance Evaluation Review*, 39(3):62–66, 2011.

[75] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew. Greening geographical load balancing. In *Proceedings of the ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems*, pages 233–244. ACM, 2011.

[76] Z. Liu, A. Wierman, Y. Chen, B. Razon, and N. Chen. Data center demand response: Avoiding the coincident peak via workload shifting and local generation. *Performance Evaluation*, 70(10):770–791, 2013.

[77] J. Maida. Increase in data center construction will significantly augment the global data center precision air conditioning market until 2020, says technavio. https://goo.gl/9uX1Gj, Jun 2016. Online; accessed 11-Nov-2016.

[78] M. Malik and H. Homayoun. Big data on low power cores: Are low power embedded processors a good fit for the big data workloads? In *Computer Design (ICCD), 2015 33rd IEEE International Conference on*, pages 379–382. IEEE, 2015.

[79] K. T. Malladi, B. C. Lee, F. A. Nothaft, C. Kozyrakis, K. Periyathambi, and M. Horowitz. Towards energy-proportional datacenter memory with mobile dram. In *ACM SIGARCH Computer Architecture News*, volume 40, pages 37–48. IEEE Computer Society, 2012.

[80] J. Mars, L. Tang, R. Hundt, K. Skadron, and M. L. Soffa. Bubble-up: Increasing utilization in modern warehouse scale computers via sensible co-locations. In *Proceedings of the 44th annual IEEE/ACM International Symposium on Microarchitecture*, pages 248–259. ACM, 2011.

[81] D. Meisner, B. T. Gold, and T. F. Wenisch. Powernap: eliminating server idle power. In *ACM Sigplan Notices*, volume 44, pages 205–216. ACM, 2009.

[82] D. Meisner, C. M. Sadler, L. A. Barroso, W.-D. Weber, and T. F. Wenisch. Power management of online data-intensive services. In *Computer Architecture (ISCA), 2011 38th Annual International Symposium on*, pages 319–330. IEEE, 2011.

[83] D. Meisner and T. F. Wenisch. Peak power modeling for data center servers with switched-mode power supplies. In *Low-Power Electronics and Design (ISLPED), 2010 ACM/IEEE International Symposium on*, pages 319–324. IEEE, 2010.

[84] R. Miller. The billion dollar data centers. https://goo.gl/bhfteC, Apr 2009. Online; accessed 11-Nov-2016.

[85] A. K. Mishra, J. L. Hellerstein, W. Cirne, and C. R. Das. Towards characterizing cloud backend workloads: insights from google compute clusters. *ACM SIGMETRICS Performance Evaluation Review*, 37(4):34–41, 2010.

[86] J. Moore, J. S. Chase, and P. Ranganathan. Weatherman: Automated, online and predictive thermal mapping and management for data centers. In *Autonomic Computing, 2006. ICAC'06. IEEE International Conference on*, pages 155–164. IEEE, 2006.

[87] J. D. Moore, J. S. Chase, P. Ranganathan, and R. K. Sharma. Making scheduling" cool": Temperature-aware workload placement in data centers. In *USENIX annual technical conference, General Track*, pages 61–75, 2005.

[88] K. Ovtcharov, O. Ruwase, J.-Y. Kim, J. Fowers, K. Strauss, and E. S. Chung. Accelerating deep convolutional neural networks using specialized hardware. *Microsoft Research Whitepaper*, 2(11), 2015.

[89] P. Padala, X. Zhu, Z. Wang, S. Singhal, K. G. Shin, et al. Performance evaluation of virtualization technologies for server consolidation. *HP Labs Tec. Report*, 2007.

[90] Paraffin wax listings on alibaba. http://www.alibaba.com/. Online; accessed 07-Nov-2016.

[91] P. R. Parida, M. David, M. Iyengar, M. Schultz, M. Gaynes, V. Kamath, B. Kochuparambil, and T. Chainer. Experimental investigation of water cooled server microprocessors and memory devices in an energy efficient chiller-less data center. In *Semiconductor Thermal Measurement and Management Symposium (SEMI-THERM), 2012 28th Annual IEEE*, pages 224–231. IEEE, 2012.

[92] C. D. Patel, R. Sharma, C. E. Bash, and A. Beitelmal. Thermal considerations in cooling large scale high compute density data centers. In *Thermal and Thermomechanical Phenomena in Electronic Systems, 2002. ITHERM 2002. The Eighth Intersociety Conference on*, pages 767–776. IEEE, 2002.

[93] M. K. Patterson. The effect of data center temperature on energy efficiency. In *Thermal and Thermomechanical Phenomena in Electronic Systems, 2008. ITHERM 2008. 11th Intersociety Conference on*, pages 1167–1174. IEEE, 2008.

[94] M. K. Patterson and D. Fenwick. The state of datacenter cooling. *Intel Corporation White Paper. Available at http://download. intel. com/technology/eep/data-center-efficiency/stateof-date-center-cooling. pdf*, 2008.

[95] S. Pelley, D. Meisner, T. F. Wenisch, and J. W. VanGilder. Understanding and abstracting total data center power. In *Workshop on Energy-Efficient Design*, 2009.

[96] K. Pielichowska and K. Pielichowska. Phase change materials for thermal energy storage. In *Progress in Materials Science*, volume 65, pages 67–123, 2014.

[97] K. Pielichowska and K. Pielichowska. Phase change materials for thermal energy storage. In *Progress in Materials Science*, volume 65, pages 67–123, 2014.

[98] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath. Load balancing and unbalancing for power and performance in cluster-based systems. In *Workshop on compilers and operating systems for low power*, volume 180, pages 182–195. Barcelona, Spain, 2001.

[99] E. Pinheiro, W.-D. Weber, and L. A. Barroso. Failure trends in a large disk drive population. In *FAST*, volume 7, pages 17–23, 2007.

[100] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[101] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al. The kaldi speech recognition toolkit. In *IEEE 2011 workshop on automatic speech recognition and understanding*, number EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[102] A. Putnam, A. M. Caulfield, E. S. Chung, D. Chiou, K. Constantinides, J. Demme, H. Esmaeilzadeh, J. Fowers, G. P. Gopal, J. Gray, et al. A reconfigurable fabric for accelerating large-scale datacenter services. In *Computer Architecture (ISCA), 2014 ACM/IEEE 41st International Symposium on*, pages 13–24. IEEE, 2014.

[103] Z. Qu, W. Li, J. Wang, and W. Tao. Passive thermal management using metal foam saturated with phase change material in a heat sink. *International Communications in Heat and Mass Transfer*, 39(10):1546–1549, 2012.

[104] A. Raghavan, L. Emurian, L. Shao, M. Papaefthymiou, K. Pipe, T. Wenisch, and M. Martin. Utilizing dark silicon to save energy with computational sprinting. 2013.

[105] A. Raghavan, L. Emurian, L. Shao, M. Papaefthymiou, K. P. Pipe, T. F. Wenisch, and M. M. Martin. Computational sprinting on a hardwaresoftware testbed. volume 48, pages 155–166. ACM, 2013.

[106] A. Raghavan, Y. Luo, A. Chandawalla, M. Papaefthymiou, K. P. Pipe, T. F. Wenisch, and M. M. Martin. Computational sprinting. In *High Performance Computer Architecture (HPCA), 2012 IEEE 18th International Symposium on*, pages 1–12. IEEE, 2012.

[107] A. Raghavan, Y. Luo, A. Chandawalla, M. Papaefthymiou, K. P. Pipe, T. F. Wenisch, and M. M. Martin. Designing for responsiveness with computational sprinting. *Micro, IEEE*, 33(3):8–15, 2013.

[108] K. Roth, R. Zogg, and J. Brodrick. Cool thermal energy storage. *ASHRAE journal*, 48(9):94–96, 2006.

[109] Rubitherm. RT28HC data sheet, May 2011.

[110] D. Rybach, S. Hahn, P. Lehnen, D. Nolden, M. Sundermeyer, Z. Tüske, S. Wiesler, R. Schlüter, and H. Ney. Rasr-the rwth aachen university open source speech recognition toolkit. In *Proc. IEEE Automatic Speech Recognition and Understanding Workshop*, 2011.

[111] B. Schroeder, E. Pinheiro, and W.-D. Weber. Dram errors in the wild: a large-scale field study. In *ACM SIGMETRICS Performance Evaluation Review*, volume 37, pages 193–204. ACM, 2009.

[112] L. Shao, A. Raghavan, L. Emurian, M. C. Papaefthymiou, T. F. Wenisch, M. M. Martin, and K. P. Pipe. On-chip phase change heat sinks designed for computational sprinting. In *Semiconductor Thermal Measurement and Management Symposium (SEMI-THERM), 2014 30th Annual*, pages 29–34. IEEE, 2014.

[113] A. Sharma, V. Tyagi, C. Chen, and D. Buddhi. Review on thermal energy storage with phase change materials and applications. volume 13, pages 318–345. Elsevier, 2009.

[114] M. Shaw and M. Goldstein. Open cloudserver blade specification v1.0. Open Compute Project, 2014.

[115] M. Shaw and M. Goldstein. Open cloudserver chassis specification v1.0. Open Compute Project, 2014.

[116] M. Skach, M. Arora, C.-H. Hsu, Q. Li, D. Tullsen, L. Tang, and J. Mars. Thermal time shifting: Leveraging phase change materials to reduce cooling costs in warehouse-scale computers. In *Proceedings of the 42nd Annual International Symposium on Computer Architecture*, pages 439–449. ACM, 2015.

[117] M. Skach, M. Arora, C.-H. Hsu, Q. Li, D. Tullsen, L. Tang, and J. Mars. Thermal time shifting: Decreasing data center cooling costs with phase-change materials. *IEEE Internet Computing*, 21(4):34–43, 2017.

[118] M. Skach, M. Arora, D. Tullsen, L. Tang, and J. Mars. Virtual melting temperature: Managing server load to minimize cooling overhead with phase change materials. 2018.

[119] T. J. Stachecki and K. Ghose. Short-term load prediction and energy-aware load balancing for data centers serving online requests.

[120] Y. Sverdlik. Here's how much energy all us data centers consume. https://goo.gl/UA8u97, Jun 2016. Online; accessed 11-Nov-2016.

[121] Switch. SUPERNAP Las Vegas digital exchange campus. https://goo.gl/mmi1km, 2016. Online; accessed 16-Nov-2016.

[122] Q. Tang, S. K. S. Gupta, and G. Varsamopoulos. Energy-efficient thermal-aware task scheduling for homogeneous high-performance computing data centers: A cyber-physical approach. *IEEE Transactions on Parallel and Distributed Systems*, 19(11):1458–1472, 2008.

[123] Google Transparency Report. https://goo.gl/lz3OKG, 2011. Online; accessed 2011.

[124] O. Tuncer, K. Vaidyanathan, K. Gross, and A. K. Coskun. Coolbudget: Data center power budgeting with workload and cooling asymmetry awareness. In *Computer Design (ICCD), 2014 32nd IEEE International Conference on*, pages 497–500. IEEE, 2014.

[125] M. Uddin and A. A. Rahman. Server consolidation: An approach to make data centers energy efficient and green. *arXiv preprint arXiv:1010.5037*, 2010.

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

[127] A. Verma, G. Dasgupta, T. K. Nayak, P. De, and R. Kothari. Server workload analysis for power minimization using consolidation. In *Proceedings of the 2009 conference on USENIX Annual technical conference*, pages 28–28. USENIX Association, 2009.

[128] F. Volle, S. V. Garimella, and M. A. Juds. Thermal management of a soft starter: transient thermal impedance model and performance enhancements using phase change materials. *Power Electronics, IEEE Transactions on*, 25(6):1395–1405, 2010.

[129] D. Wang, C. Ren, and A. Sivasubramaniam. Virtualizing power distribution in datacenters. In *ACM SIGARCH Computer Architecture News*, volume 41, pages 595–606. ACM, 2013.

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

[131] D. Wong and M. Annavaram. Knightshift: Scaling the energy proportionality wall through server-level heterogeneity. In *2012 45th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 119–130. IEEE, 2012.

[132] D. Wong and M. Annavaram. Implications of high energy proportional servers on cluster-wide energy proportionality. In *2014 IEEE 20th International Symposium on High Performance Computer Architecture (HPCA)*, pages 142–153. IEEE, 2014.

[133] Q. Wu, Q. Deng, L. Ganesh, C.-H. Hsu, Y. Jin, S. Kumar, B. Li, J. Meza, and Y. J. Song. Dynamo: Facebook's data center-wide power management system. In *Computer Architecture (ISCA), 2016 ACM/IEEE 43rd Annual International Symposium on*, pages 469–480. IEEE, 2016.

[134] H. Xu, C. Feng, and B. Li. Temperature aware workload management in geo-distributed datacenters. In *Proceedings of the 10th International Conference on Autonomic Computing (ICAC 13)*, pages 303–314, 2013.

[135] H. Xu, C. Feng, and B. Li. Temperature aware workload management in geo-distributed data centers. *IEEE Transactions on Parallel and Distributed Systems*, 26(6):1743–1753, 2015.

[136] H. Yang, A. Breslow, J. Mars, and L. Tang. Bubble-flux: Precise online qos management for increased utilization in warehouse scale computers. In *ACM SIGARCH Computer Architecture News*, volume 41, pages 607–618. ACM, 2013.

[137] J. Zhang, M. Shihab, and M. Jung. Power, energy and thermal considerations in ssd-based i/o acceleration. In *Advanced Computing Systems Association: HotStorage 2014*. USENIX, 2014.

[138] Y. Zhang, M. A. Laurenzano, J. Mars, and L. Tang. Smite: Precise qos prediction on real-system smt processors to improve utilization in warehouse scale computers. In *Microarchitecture (MICRO), 2014 47th Annual IEEE/ACM International Symposium on*, pages 406–418. IEEE, 2014.

[139] Y. Zhang, G. Prekas, G. M. Fumarola, M. Fontoura, I. Goiri, and R. Bianchini. History-based harvesting of spare cycles and storage in large-scale datacenters. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation*, OSDI-12, Berkeley, CA, USA, 2016. USENIX.

[140] Y. Zhang, Y. Wang, and X. Wang. Testore: exploiting thermal and energy storage to cut the electricity bill for datacenter cooling. In *Proceedings of the 8th International Conference on Network and Service Management*, pages 19–27. International Federation for Information Processing, 2012.

[141] W. Zheng, K. Ma, and X. Wang. Exploiting thermal energy storage to reduce data center capital and operating expenses. In *Proceedings of the IEEE 19th International Symposium on High-Performance Computer Architecture*, pages 132–141. IEEE, 2014.

[142] S. Zimmermann, I. Meijer, M. K. Tiwari, S. Paredes, B. Michel, and D. Poulikakos. Aquasar: A hot water cooled data center with direct energy reuse. *Energy*, 43(1):237–245, 2012.