Title

Thermal Time Shifting: Decreasing Data Center Cooling Costs with Phase-Change Materials

Permalink

https://escholarship.org/uc/item/5s9429mj

Journal

IEEE Internet Computing, 21(4)

ISSN

1089-7801

Authors

Skach, Matt
Arora, Manish
Hsu, Chang-Hong
et al.

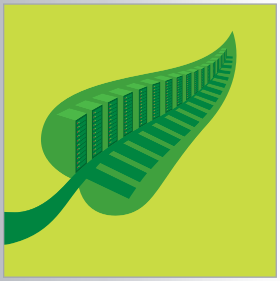
Publication Date

2017

DOI

10.1109/mic.2017.2911418

Peer reviewed



Thermal Time Shifting: Decreasing Data Center Cooling Costs with Phase-Change Materials

As data centers increase in size and computational capacity, their growth comes at a cost: an increasing thermal load that must be removed to prevent overheating. Here, the authors propose using phase-change materials (PCMs) to shape a data center's thermal load, absorbing and releasing heat when it's advantageous. They evaluate three important opportunities for cost savings.

Matt Skach
University of Michigan

Manish Arora
AMD Research and University of California, San Diego

Chang-Hong Hsu
University of Michigan

Qi Li and Dean Tullsen
University of California, San Diego

Lingjia Tang and Jason Mars
University of Michigan

Increasingly, a significant portion of the world's computation and storage is concentrated in the cloud, where it takes place in large data centers; these data centers are also referred to as *warehouse-scale computers* (WSCs).[1] One implication of this centralization of the world's computing infrastructure is that these data centers 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 reductions in energy usage that are visible on a national level.[2]

Because of these systems' increasing computing density, 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 data center. The scale of cooling infrastructure can cost more than US$8 million,[2] even if the data center only reaches peak utilization for a fraction of a load cycle. The cooling system also might become inadequate as servers are upgraded or replaced and the data center's thermal characteristics change.

To mitigate these challenges, we propose the use of *phase-change materials* (PCMs: materials that absorb or release a significant amount of heat when melting or freezing) 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 might not be immediately obvious, because heat isn't being eliminated,

1089-7801/17/$33.00 © 2017 IEEE

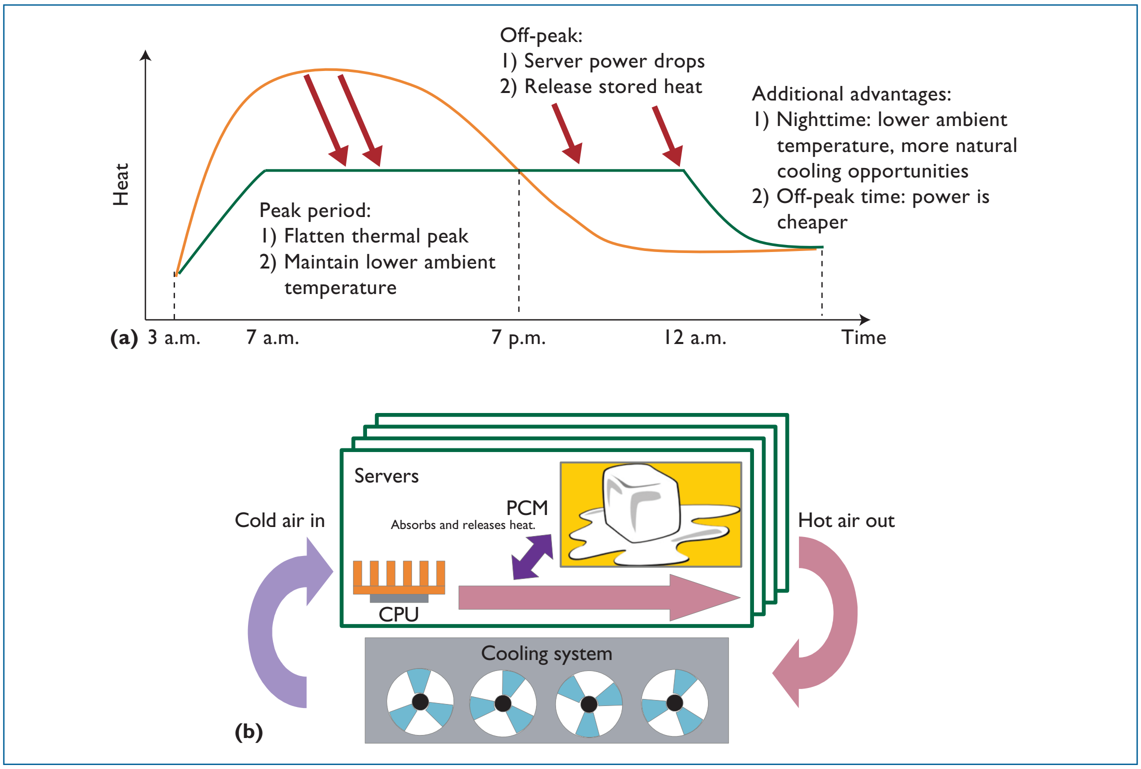


Figure 1. Thermal time shifting. (a) Thermal time shifting using phase-change materials (PCMs). (b) Integrating PCM into a data center.

only temporarily stored and then released at a later time. However, the key insight of this work is that the ability to store heat lets us shape the data center's thermal behavior, releasing the heat only when it is advantageous.

Figure 1a illustrates this thermal time shifting. This figure presents a diurnal pattern with a peak utilization and heat output during the middle of the day (7 a.m. to 7 p.m.). 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 much cheaper cooling system with a smaller cooling capacity.

This PCM-enabled thermal time shifting lets us 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[2,3] demonstrates the ability to produce a flat power demand in the face of uneven diurnal power peaks; however, cooling needs still trend and peak with the workload. This work allows the cooling power also to be flattened, placing a tighter cap on total data center power.

Alternatively, we can use PCM to pack more computational capacity into the warehouse of an existing data center with a given cooling infrastructure without adding cooling capacity: this better amortizes the fixed infrastructure costs of the entire data center. Furthermore, given a load pattern such as the one in Figure 1a, the ability to shift cooling demands from peak hours to 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.[3,4]

Last, we can use PCM in a thermally constrained data center to handle short peaks above the thermal limit while still scheduling non-latency-sensitive batch jobs around the thermal needs of latency-critical jobs without violating thermal constraints. Recent work to reduce contention between jobs colocated on the same chip multicore processors (CMPs)[5-7] can enable significantly higher resource usage in a data center environment, provided that the thermal constraints still can be met. With a computationally light analytical model to decide when to schedule batch jobs, PCM enables these data centers to greatly increase the throughput of non-latency-critical batch jobs while still meeting the peak latency-critical demand.

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

- We need to investigate the tradeoffs of various PCMs and identify the material that fits best in the data center environment. Selecting the correct PCM is critical to maximize impact while minimizing the total cost of ownership (TCO).
- We need to investigate suitable design strategies for integrating PCM in thousands of servers. Modern commodity servers are designed with excess cooling and interior space to allow for many applications, and there are ways to leverage this reconfigurability to enhance PCM performance.
- We need to quantify the potential cost savings of using PCM. Data center cooling systems are expensive, and even a small reduction can save hundreds of thousands or millions of dollars.

In this work, we present the advantages of PCM on a data center scale. We consider several PCMs for deployment in a data center, and select one for further investigation. We previously validated our simulator using real hardware,[8] and use the simulator to perform a scale-out study of PCM on three different server configurations to predict the impact of PCM deployed in a data center. In an unconstrained data center, we find PCM enables a 12 percent reduction in peak cooling utilization or the deployment of 14.6 percent more servers under the same thermal budget. In a thermally constrained data center (for example, where there are more servers than the cooling system can cool), we find PCM can increase peak throughput by up to 69 percent while delaying the data center from reaching a thermal limit by more than three hours. And a wax-aware scheduler enables an 11 percent reduction in peak cooling load at the same time batch jobs are added during off hours, to increase daily average usage by 36–52 percent.

Integrating PCM in WSCs

To enable thermal time shifting, this work proposes to place a quantity of PCM inside each server, as Figure 1b shows. 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 resolidify and release energy until the PCM is solid again.

Placing PCM directly in contact with the heat spreader of a single processor is beneficial for computational sprinting[9] and other short-term cooling applications, but we require a much greater quantity of PCM in a data center-sized cooling system with a 24-hour thermal cycle[1] that significantly impedes CPU cooling if placed in direct contact. Placing the 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 a layer of PCM outside the data center or adding a layer of insulation in the walls and ceiling (reducing the ability of heat to escape when ambient conditions are favorable) require infrastructure to move heat to the PCM and suffer a lower temperature differential, because of heat loss and mixing over the travel distance.

Thus, the advantages of our PCM-enabled system are simple: the PCM is entirely passive. There's no power, software, or floor space overhead required to add PCM to a data center (although software components might be implemented for additional benefits) and minimal labor is needed after installation to achieve the potential cost savings.

Investigating PCM Characteristics

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

Melting temperature is critical, because it determines when our PCM absorbs and releases significant amounts of heat. In a data center, 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 PCM's energy density defines how much energy it can store and is proportional to the heat of fusion (melting energy) and the PCM's density in both solid and liquid phases. A high-energy density is desirable to maximize energy storage, using the small amount of space available inside 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.

After carefully considering available PCMs, we select commercial paraffin wax for our data center scale-out study.[8] Out of five solid-liquid PCM categories, paraffin waxes offer a melting temperature well suited to data center ranges, a lower density but midrange heat of fusion, very good material stability, and low electrical conductivity and low corrosivity in the event of a spill.[10]

Commercial-grade paraffin is a less-refined wax consisting of a mixture of *n*-paraffin molecules. It has a slightly lower heat of fusion of 200 joules per gram (J/g), but is much less expensive than eicosane (an *n*-paraffin considered to cool microcontrollers in prior work).[9] 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: 50× cheaper for 20 percent lower energy per gram compared to eicosane, which we deem as a reasonable tradeoff.

Methodology

In this section, we introduce our methodology and candidate machines for a scale-out study on PCM data centers. In prior work,[8] we validated our experimental methodology using real hardware. We use ANSYS Icepak, a commercial computational fluid dynamic suite designed to simulate heating and cooling in electronic devices, to perform our design space exploration where physical experiments such as component layout and precise variation of airflow would not be practical.[8]

We select three homogeneous data center configurations — each provisioned with a different type of machine — and evaluate each data center using real workload traces from Google. The results of our scale-out study are presented in the next section.

Servers

We consider three different server designs for our scale-out study.

One rack unit (1U) commodity server. The Lenovo RD330 we validated[8] is a low-power, 1U commodity server. We conduct a series of experiments in Icepak blocking airflow with a uniform grille downwind of the CPU heat sinks (see Figure 2b, part 1). From 0 percent (no air blocked) up to 90 percent of airflow blocked, we observe a 14°C increase in air temperatures at the outlet while maintaining a safe CPU temperature. We block 70 percent of airflow downwind of the CPUs to add 1.2 liters of wax in sealed containers, as modeled in Figure 2a (part 1).

2U commodity server. We consider a high-throughput commodity server, modeled after the Sun X4470, with up to four Intel E7-4800 processors. We model the server in Icepak[8] in Figure 2a (part 2). In Figure 2b (part 2), we plot temperature in the server as air is blocked by a uniform grille. When less than 50 percent of the airflow through our 2U commodity server is blocked, we observe an almost negligible impact on outlet and CPU temperatures, while at above 50 percent the temperature increases exponentially. We add four 1-liter aluminum boxes filled with wax (colored gold in Figure 2a, part 2). These boxes block 69 percent of the airflow through the server, increasing the outlet and CPU temperatures by less than 6°C.

Open Compute blade server. We model the Open Compute server in Icepak[8] based on published dimensions and specifications for the form factor, CPUs, hard drives, and motherboard.[11] (We don't model the volume or power requirements of the Catapult FPGA board.[11])

In Figure 2a (parts 3 through 5), we present three Icepak models of the Open Compute configurations. Figure 2a (part 3) shows the production Open Compute configuration. To increase the wax capacity, we consider an alternate configuration where we switch the CPU location with that of the solid-state drives (SSDs) to increase the downwind volume, and the redundant hard disk drives have been replaced with a second set of SSDs to achieve 1.5 liters of wax, as Figure 2a (part 5) shows, without increasing the airflow blockage versus the production blade.

Google Workload

We use a two-day workload trace from Google[2] to evaluate the effects of wax on our three data center server configurations. The workload we consider has three different job types: web

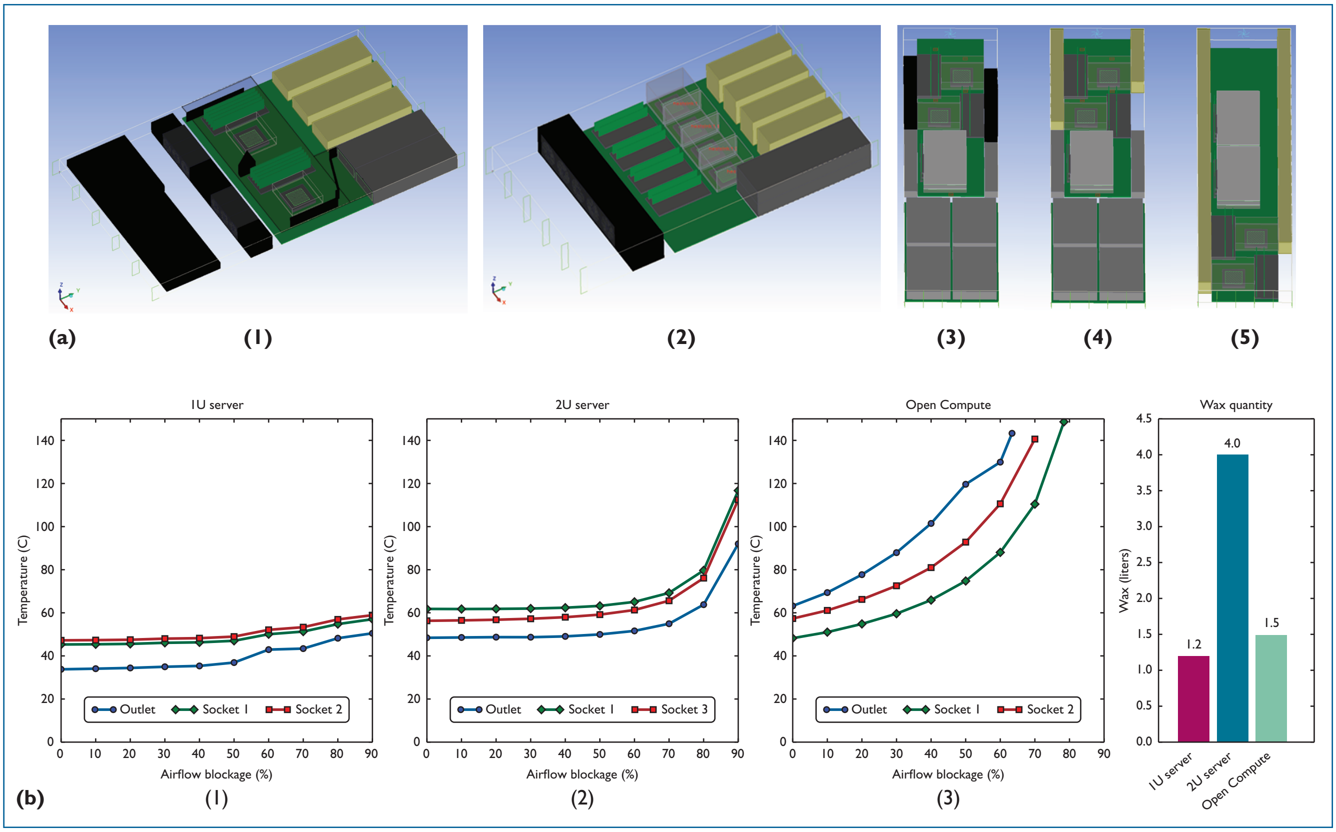


Figure 2. Variety of servers considered for PCM deployment. (a) First, we consider (1) a one rack unit (1U) low-power server modeled in Icepak with 1.2 liters of wax (gold); (2) a 2U high-throughput server with four CPU sockets and 4 liters of wax; (3) a Microsoft Open Compute server;[11] (4) Open Compute with airflow inhibitors replaced with wax containers; and (5) Open Compute reconfigured with 1.5 liters of wax. (b) Server temperatures as airflow through each server is blocked. (1) CPU temperatures in the 1U server rise less than 2°C below 50 percent, and begin to rise quicker thereafter. (2) Temperatures in the 2U server are stable below 60 percent, but quickly rise to unsafe levels above 70 percent obstructed airflow. (3) Temperatures in the Open Compute server rise to unsafe levels as soon as almost any airflow is obstructed.

search, social networking (Orkut), and MapReduce from 17–18 November 2010. This data was acquired as described by Vasileios Kontorinis and colleagues,[2] and normalized for a 50 percent average load and 95 percent peak load for a cluster of 1,008 servers of each configuration. After 2011, Google changed the format of its transparency report, so newer data are unavailable.

To model traffic and data center throughput, we use DCSim, a traffic-based simulator previously used by Kontorinis and colleagues.[2] 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 data center. 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.

Batch Job Scheduling

Although PCM-enabled thermal time shifting doesn't require a scheduler to provide benefit, it can be advantageous to add one in certain cases. To maximize resource usage and improve the return on investment, it's advantageous to schedule non-latency-sensitive batch jobs (such as video encoding and offline data analytics) during off-peak hours.[7] Recent work[5,6] has made significant progress to reduce contention between co-running jobs when one job is latency-insensitive, enabling runs at degraded performance levels without impacting the quality-of-service (QoS) targets of the critical job. These jobs must be scheduled to ensure sufficient wax capacity is

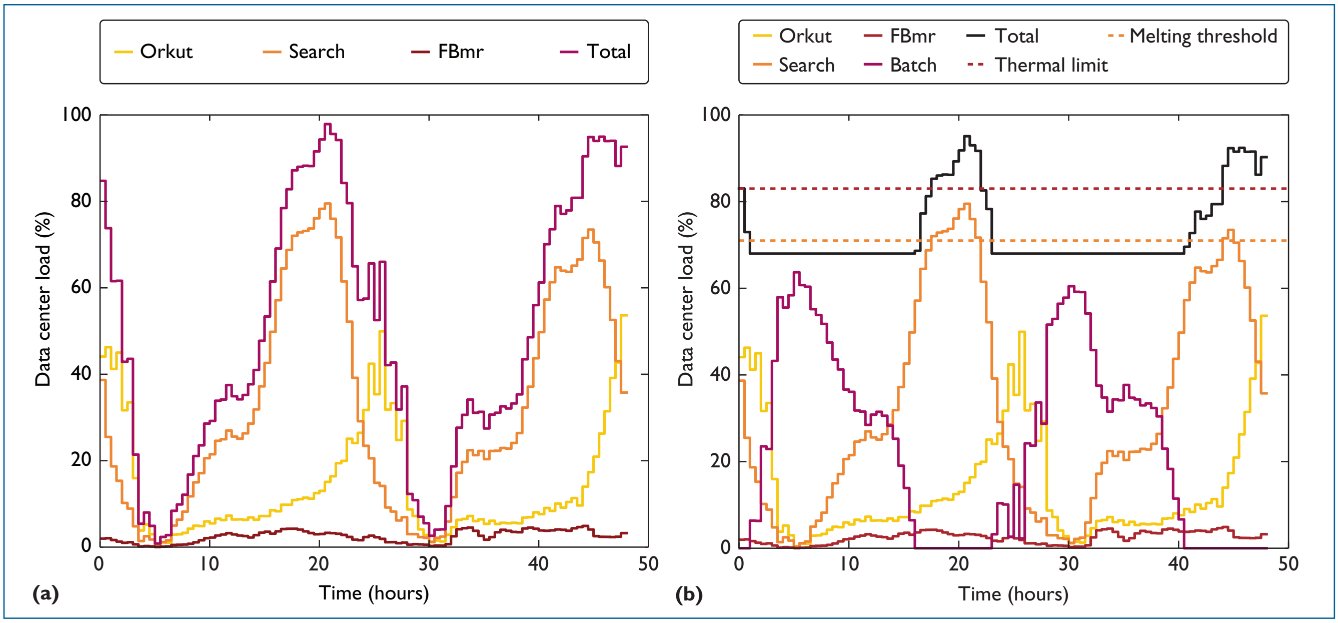


Figure 3. Two-day workload graphs. (a) Two-day data center workload trace from Google,[2] normalized to peak throughput. (b) Two-day data center load trace, where thermal headroom has been filled with non-latency-sensitive batch jobs.

available during peak hours to meet peak demand (Figure 3b). However, because we can't exceed the data center's thermal limit we must ensure data center temperatures allow the wax to melt.

Evaluation

In this section, we evaluate three opportunities for cost savings using wax to reduce the cooling load of a data center.

PCM to Reduce Cooling Load

We first consider a data center with a fully subscribed cooling system that can remove the peak cooling load indefinitely. The cooling load of a data center is the power that must be removed to maintain a constant cold-aisle temperature,[12] and allows a direct comparison between different server, temperature, and data center configurations. In Figure 4a, we plot the peak cluster cooling load for a cluster of 1,008 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 select the appropriate melting temperature to minimize cooling load. The range of melting temperature available in commercial-grade paraffin lets us select one with an optimal melting threshold to reduce the peak cooling load of each cluster, and the best melting temperature is determined by the load trace's shape and length: for the Google trace, we find that the best wax typically begins to melt when a server exceeds 75 percent load and melts quickly thereafter.

As shown, we achieve an 8.3 percent reduction in peak cooling in the Open Compute cluster, up to an 8.9 percent reduction in the cluster of 1U servers, and 12 percent in the cluster of 2U servers as the wax absorbs heat and melts. In a data center with 10 megawatts of critical power, this corresponds to a cost savings of up to $3.2 million per year.[8]

PCM to Increase Throughput

Next, we consider an oversubscribed data center where the cooling system is significantly smaller than the thermal output of the data center with all servers active. Such circumstances can arise as old servers are replaced with new denser servers, and as algorithms or workload patterns change.

In this oversubscribed data center, thermal management techniques such as downclocking/DVFS or relocating work to other data centers[13-15] must be applied to prevent the data center from overheating.

In Figure 4b, we plot the cluster throughput if the thermal limit didn't exist and downclocking isn't imposed, the throughput without wax, and the throughput with wax. In the trace without wax, downclocking to 1.6 GHz is imposed

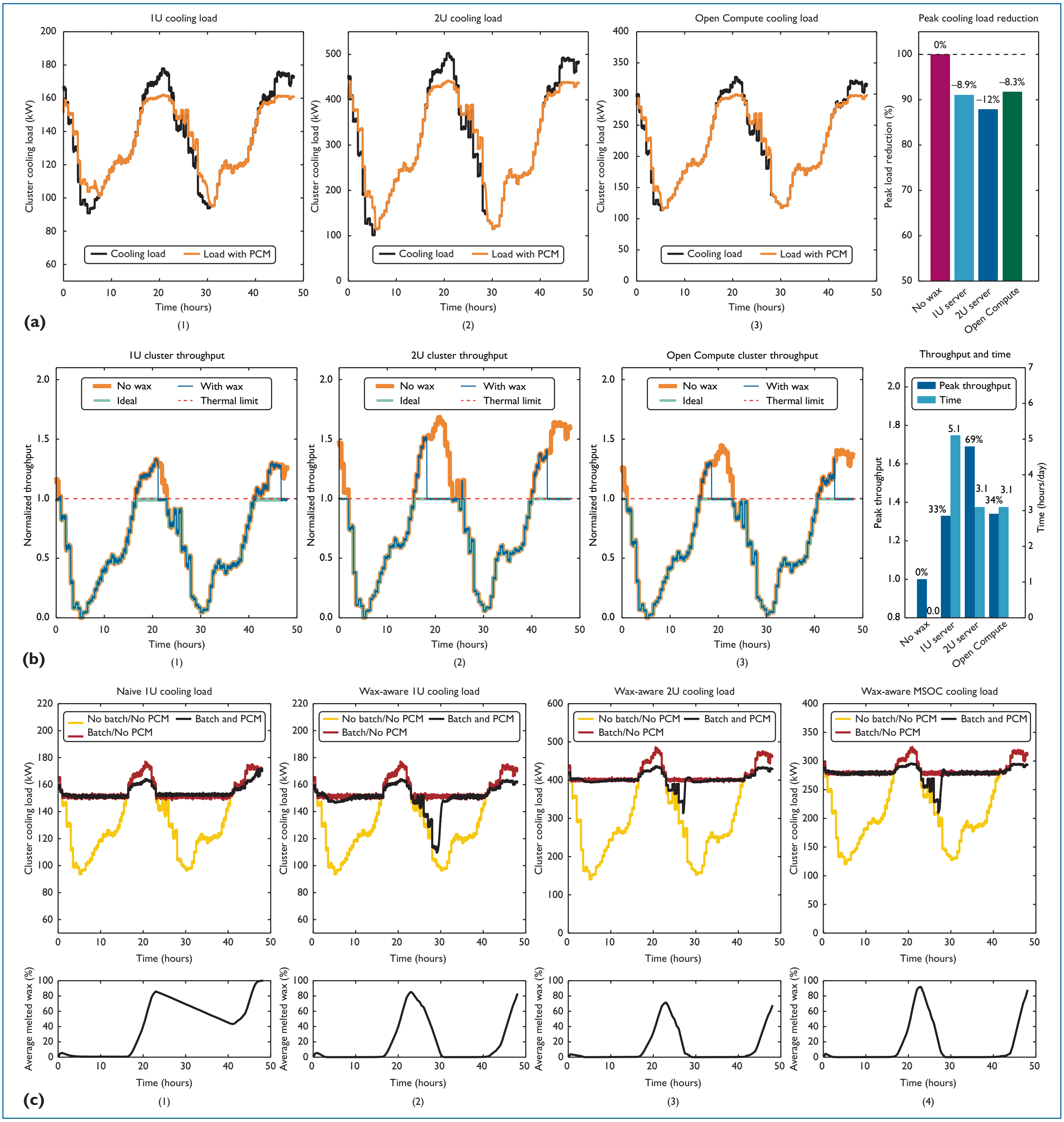


Figure 4. Two-day runs. (a) Using PCM to reduce cooling load — cooling the load per cluster over a two-day Google trace in a data center with a fully subscribed cooling system. (1) PCM reduces peak cooling load by 8.9 percent in a cluster of low-power 1U servers; (2) 12 percent in a cluster of 2U high-throughput commodity servers; and (3) by 8.3 percent in a cluster of high-density Open Compute servers. (b) Using PCM to increase throughput — the Google workload throughput is normalized to peak throughput in a thermally constrained data center. (1) PCM increases peak throughput by 33 percent over 5.1 hours in the 1U server; (2) 69 percent over 3.1 hours in the 2U server; and (3) 34 percent over 3.1 hours in the Open Compute server. (c) Using PCM-aware scheduling for batch jobs — cooling the load and wax melting state for four cluster configurations (naive 1U and wax-aware 1U, 2U, and Open Compute). (1) The naive scheduler violates thermal limits at the end of day 2 in the trace, while (2 through 4) the wax-aware schedulers delay scheduling batch jobs until the wax has resolidified, ensuring no wax is melted at the next peak's start.

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're able to maintain clock speeds and/or use as the wax absorbs thermal energy, until the wax's thermal capacity is full. Once the wax is melted and can absorb no more energy downclocking, job relocation must be applied to prevent the data center from overheating — 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 percent increase in peak throughput during that time. In the 1U low power cluster, PCM delays thermal constraints by 5.1 hours, with a 33 percent 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 percent.

PCM-Aware Scheduling for Batch Jobs

Last, we consider a data center implementing the cooling overprovisioning techniques. In this data center, approximately two-thirds of the day is spent below the thermal limit, leaving computing power available and corresponding cooling resources available to run the servers at higher load levels.

Scheduling non-latency-sensitive batch jobs in a data center greatly increases hardware utilization,[7] but introduces new challenges in a PCM-enhanced data center. Two constraints in particular must be addressed to prevent violations of thermal constraints: First, batch jobs must be scheduled on servers such that load levels during off-peak hours don't melt wax (see Figure 3b). That is, the total load of batch jobs and non-batch jobs must not produce temperatures inside the servers that are greater than the deployed wax's melting temperature, or else some or all of the data center's thermal energy storage won't be available at the start of the peak hours.

This can be accomplished using existing techniques and sensors. To deploy batch jobs, the cluster scheduler must already have the ability to monitor server performance and usage, and know whether launching additional batch jobs will cause latency-sensitive jobs to violate QoS targets. From the cluster scheduler's perspective, whether the server will melt wax or not is a similar binary constraint, given a job of known properties (either dispatch more jobs or don't). Each server can then utilize preprofiled job parameters or temperature sensors already present in most servers to determine if additional jobs will raise temperatures above the wax melting threshold.

Second, if the scheduler places batch jobs up to the thermal threshold immediately after exiting peak hours, then the wax won't sufficiently freeze before the next thermal peak (see Figure 4c, part 1). To ensure that the cluster is ready for the next thermal peak, the cluster-level scheduler must be aware of the wax's melted state and delay scheduling enough batch jobs to ensure that the wax is sufficiently frozen before the next peak's beginning.

To accomplish this, we add a simple computational model running on each server to track the wax's state. Only a coarse-grain simulation is required: tracking temperature and time at the temperature is enough for a server to estimate the current wax state using a lookup table of known temperature and the melting or freezing rate at that temperature. This lookup table must be measured and produced once, but then it can be copied and deployed across the entire data center. To avoid model drift, both the model and real wax must become either fully melted or fully cooled once per cycle (because neither can go below 0 percent or above 100 percent melted).

In our experimental setup, each server updates its wax state once per minute and reports the results to the cluster-level scheduler. The cluster-level scheduler then adds one additional Boolean decision: if a server's wax is above 5 percent melted, the cluster scheduler doesn't place batch jobs on that server yet.

In Figure 4c (parts 2 through 4), we plot the cooling load and wax melting state for the 1U, 2U, and Open Compute server clusters. The effects of the scheduling policy are immediately visible following the first peak, when load

The effects of the scheduling policy are immediately visible following the first peak, when load levels continue to drop ...

levels continue to drop (similar to the load with no batch jobs) until the wax is melted, at which time the scheduling of batch jobs resumes.

Using the wax-aware scheduler, average use can be improved by 36 percent in the 1U cluster, 49 percent in the 2U cluster, and 52 percent in the Open Compute cluster at the same time that thermal time shifting enables an up to 11 percent reduction in peak cooling load versus data center configurations (with wax but without wax-aware scheduling).

In this work, we introduce thermal time shifting; the ability to reshape a thermal load by storing and releasing energy when beneficial. We study paraffin wax, a PCM that we place inside a real server to demonstrate thermal time shifting in a single server. We perform a scale-out study to show that thermal time shifting with a PCM can be used to reduce peak cooling load by up to 12 percent or increase the number of servers by up to 14.6 percent (5,300 additional servers) without increasing the cooling load. In a thermally constrained data center, we demonstrate that PCM can increase peak throughput by up to 69 percent while simultaneously postponing the onset of thermally mandated throughput reduction by more than three hours. And when batch jobs are placed with a wax-aware scheduler, we show that average daily use can be increased safely by 36–52 percent while still reducing the peak cooling load by up to 11 percent. In future work, we'll explore the impact of workload variation and different workload compositions and show how active management of thermal time shifting can make the technique even more broadly applicable.

References

1. 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*, vol. 4, no. 1, 2009, pp. 1–108.
2. V. Kontorinis et al., "Managing Distributed UPS Energy for Effective Power Capping in Data Centers," *Proc. Int'l Symp. Computer Architecture*, 2012, pp. 488–499.
3. S. Govindan, A. Sivasubramaniam, and B. Urgaonkar, "Benefits and Limitations of Tapping into Stored Energy for Datacenters," *Proc. Int'l Symp. Computer Architecture*, 2011, pp. 341–351.
4. I. Goiri et al., "Parasol and Greenswitch: Managing Datacenters Powered by Renewable Energy," *ACM Sigarch Computer Architecture News*, vol. 41, no. 1. 2013, pp. 51–64.
5. J. Mars et al., "Bubble-Up: Increasing Utilization in Modern Warehouse Scale Computers via Sensible Co-Locations," *Proc. 44th Ann. IEEE/ACM Int'l Symp. Microarchitecture*, 2011, pp. 248–259.
6. M.A. Laurenzano et al., "Protean Code: Achieving Near-Free Online Code Transformations for Warehouse Scale Computers," *Proc. 47th Ann. IEEE/ACM Int'l Symp. Microarchitecture*, 2014, pp. 558–570.
7. Y. Zhang et al., "History-Based Harvesting of Spare Cycles and Storage in Large-Scale Datacenters," *Proc. 12th Usenix Symp. Operating Systems Design and Implementation*, 2016, pp. 755–770.
8. M. Skach et al., "Thermal Time Shifting: Leveraging Phase Change Materials to Reduce Cooling Costs in Warehouse-Scale Computers," *ACM Sigarch Computer Architecture News*, vol. 43, no. 3. 2015, pp. 439–449.
9. A. Raghavan et al., "Computational Sprinting," *Proc. IEEE Int'l High-Performance Computer Architecture*, 2012; doi:10.1109/HPCA.2012.6169031.
10. K. Pielichowska and K. Pielichowski, "Phase Change Materials for Thermal Energy Storage," *Progress in Materials Science*, vol. 65, 2014, pp. 67–123.
11. A. Putnam et al., "A Reconfigurable Fabric for Accelerating Large-Scale Datacenter Services," *Proc. Int'l Symp. Computer Architecture*, 2014, pp. 13–24.
12. C.D. Patel et al., "Thermal Considerations in Cooling Large-Scale High Compute Density Data Centers," *Proc. 8th Intersociety Conf. Thermal and Thermomechanical Phenomena in Electronic Systems*, 2002, pp. 767–776.
13. Z. Liu et al., "Greening Geographical Load Balancing," *Proc. ACM Sigmetrics Joint Int'l Conf. Measurement and Modeling of Computer Systems*, 2011, pp. 233–244.
14. Z. Liu et al., "Geographical Load Balancing with Renewables," *ACM Sigmetrics Performance Evaluation Rev.*, vol. 39, no. 3, 2011, pp. 62–66.
15. Z. Liu et al., "Data Center Demand Response: Avoiding the Coincident Peak via Workload Shifting and Local Generation," *Performance Evaluation*, vol. 70, no. 10, 2013, pp. 770–791.

Matt Skach is a PhD candidate in the Department of Computer Science and Engineering at the University of Michigan, Ann Arbor. His research interests include novel design and technologies for energy-efficient data centers and large-scale systems. Skach has an MS in electrical engineering from the University of Michigan. Contact him at skachm@umich.edu.

Manish Arora is a member of the technical staff at AMD Research and a PhD candidate in the Computer Science and Engineering Department at the University of California San Diego. His research interests include CPU-GPU systems, thermal management, and enabling computational density in systems. Arora has a MS in computer engineering from the University of Texas at Austin. Contact him at manish.arora@amd.com.

Chang-Hong Hsu is a PhD candidate in the Department of Computer Science and Engineering at the University of Michigan, Ann Arbor. His research focuses on performance optimization of large-scale computer systems, principally in solving power and energy-inefficiency problems in data centers and improving the tail latency of emerging cloud and machine learning workloads. Hsu has an MS in electrical engineering from National Taiwan University. He was a honorary member of the Phi Tau Phi Scholastic Honor Society at National Taiwan University, and a recipient of the Chia-Lun Lo Fellowship from the University of Michigan. Contact him at hsuch@umich.edu.

Qi Li is a PhD student in the Computer Science and Engineering Department at the University of California, San Diego. Li has an MS in computer science from the University of California, San Diego. Contact him at liqi@eng.ucsd.edu.

Dean Tullsen is the chair of the Computer Science and Engineering Department at the University of California, San Diego. His research interests include architecture and compilers for multicore and multithreaded architectures. Tullsen has a PhD in computer science from the University of Washington. He's a Fellow of IEEE and ACM. Contact him at tullsen@cs.ucsd.edu.

Lingjia Tang is an assistant professor in the Computer Science and Engineering Department at the University of Michigan. She's also a co-founder and CSO of Clinc, an AI startup to take the world's most advanced AI research and make it useful for improving quality and reducing complexity in people's lives. Her research focuses on designing large-scale systems for intelligent applications. Tang has a PhD in computer science from the University of Virginia. Contact her at lingjia@umich.edu.

Jason Mars is a professor of computer science at the University of Michigan, where he directs Clarity Lab. He's also an active part of the IBM Watson Conversational System Research Center and participates in ongoing collaborative research with Intel and the US National Science Foundation. Jason is also a co-founder and CEO of Clinc, an AI startup that developed Finie, a customizable, voice-activated personal financial assistant and intelligence platform for banks. His research focuses on cross-layer systems architectures, as well as data center and warehouse-scale computer architectures. Mars has a PhD in computer science from the University of Virginia. Contact him at profmars@umich.edu.