# Cooling as You Wish: Component-Level Cooling for Heterogeneous Edge Datacenters

Qiangyu Pei, Shutong Chen, Yongjie Yuan, Qixia Zhang, Xinhui Zhu, Ziyang Jia, Fangming Liu*, *Senior Member, IEEE*

**Abstract**—As the computing frontier drifts to the edge, edge datacenters play a crucial role in supporting various real-time applications. Different from cloud datacenters, the requirements of *proximity to end-users*, *high density*, and *heterogeneity*, present new challenges to cool the edge datacenters efficiently. Although warm water cooling has become a promising cooling technique for this infrastructure, the one-size-fits-all cooling control would lower the cooling efficiency considerably because of the severe thermal imbalance across servers, hardware, and even inside one hardware component in an edge datacenter. In this work, we propose CoolEdge, a hotspot-relievable warm water cooling system for improving the cooling efficiency and saving costs of edge datacenters. Specifically, through the elaborate design of water circulations, CoolEdge can dynamically adjust the water temperature and flow rate for each heterogeneous hardware component to eliminate the hardware-level hotspots. By redesigning cold plates with vapor chambers, CoolEdge can quickly disperse the chip-level hotspots without manual intervention. We further quantify the power saving achieved by the warm water cooling theoretically, and propose a fine-grained cooling solution to decide an appropriate water temperature and flow rate periodically. We also develop a cost-effective semi-fine-grained cooling solution named CoolEdge$^+$ integrated with the power capping approach. Based on a hardware prototype and real-world traces from SURFsara and Alibaba PAI, the evaluation results show that CoolEdge reduces the cooling energy consumption by 81.81% at most, and CoolEdge$^+$ saves 35.24% more costs than CoolEdge with comparable energy consumption.

**Index Terms**—edge datacenter energy, warm water cooling, heterogeneity, hotspot relieving, vapor chamber

## 1 Introduction

Edge datacenters are emerging as a critical infrastructure for edge computing. To provide real-time services in close proximity to end-users, edge datacenters are widely distributed from commercial buildings to industrial complexes in the form of micro datacenters or server clusters. Gartner predicts around 75% of enterprise-generated data will be created and processed at the edge by 2025, though the value is only 10% in 2018 [1], bringing explosive growth in the number of edge datacenters. According to a report for edge computing [2], the edge datacenters will cost as high as $100 billion in information technology (IT) equipment capital expenditures in 2028. Although the power rating of an edge datacenter is only 10's to 100's of kW that is three orders of magnitude smaller than a cloud datacenter, such a growing number of edge datacenters will inevitably bring heavy energy burden. By 2028, the energy demand of edge datacenters will reach the same order of magnitude as that of the global datacenters in 2020 [3], [4].

- *This work was supported in part by National Key Research & Development (R&D) Plan under grant 2022YFB4501703, and in part by The Major Key Project of PCL (PCL2022A05). (Corresponding author: Fangming Liu)*
- *Q. Pei, S. Chen, Y. Yuan, Q. Zhang, and X. Zhu are with the National Engineering Research Center for Big Data Technology and System, the Services Computing Technology and System Lab, Cluster and Grid Computing Lab in the School of Computer Science and Technology, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan 430074, China. E-mail: {peiqiangyu, zhangqixia427, xhzhu}@hust.edu.cn, shutongcs@gmail.com, jayayuan@outlook.com*
- *Z. Jia is with the School of Computer Science and Technology, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan 430074, China. E-mail: zjia016@ucr.edu*
- *F. Liu is with Huazhong University of Science and Technology, and Peng Cheng Laboratory, China. E-mail: fangminghk@gmail.com*

*Manuscript received xxxx xx, 2023; revised xxxx xx, 2023.*

Despite the small power capacity, the power density of an edge datacenter is generally much higher than that of a cloud datacenter due to the area restriction. Inspur proposes an edge server NE5260M5 whose depth is 65% of the standard depth in Open Compute Project [5]. For one thing, such a short depth makes the implementation more flexible and space-saving, so that the server can be mounted on a short rack or even on the wall. For another, the short rack and compact-aisle arrangement further increase the power density. For instance, a well-designed edge datacenter can work at 2.1 kW per square foot [6], which is one magnitude higher than the power density of a cloud datacenter.

Recently, Tencent Cloud has opened its first edge datacenter to provide real-time services of video processing, cloud gaming, smart healthcare, and so on [7]. Although traditional lightweight workloads like Web services are suitable to be scheduled on a central processing unit (CPU), the emerging computational edge workloads like deep learning inference, rely heavily on accelerators, such as graphics processing units (GPUs), tensor processing units, field-programmable gate arrays, smart network interface cards, etc. Hence, to support these diverse performance-critical edge applications, the edge server needs to comprise various high-powered heterogeneous hardware, leading to a high power provisioning to edge servers [3], [8], [9].

The specific requirements of edge datacenters, including *proximity to end-users*, *high density*, and *heterogeneity*, make existing cooling techniques inefficient or even impracticable. The free cooling technique requires specific low-temperature locations with free cooling sources like the cold outdoor air, contradicting the edge's demand for proximity to end-users. In addition, high density and heterogeneity further increase

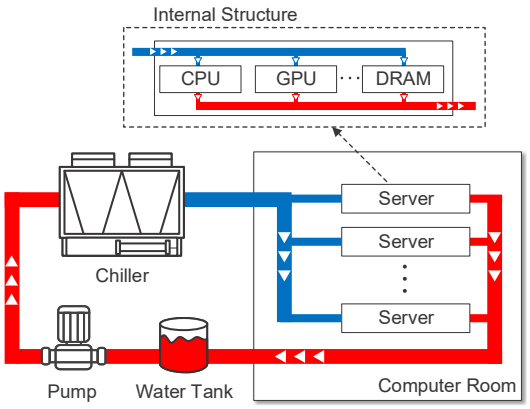



Fig. 1: Water cooling architecture in an edge datacenter.

the difficulty of efficient cooling in edge datacenters. As the power density grows dramatically, air cooling would be no more suitable and even unsafe for the edge datacenters [10], especially when dealing with thermal imbalance of heterogeneous hardware [11], [12]. Hence, we argue that water cooling is a promising technique for edge datacenters and it also has great potential for saving energy.

Currently, warm water cooling (e.g., 40°C∼50°C) emerges to reduce cooling costs by avoiding over-cooling servers running at a low utilization and allowing less or even no use of the chiller [13], [14]. Recent studies indicate that the cooling costs can be reduced by 40% through raising the water temperature [15], which is significant for the edge datacenters located in populated areas with a high electricity pricing. By employing warm water cooling, less cooling demand also enables a chiller with a smaller size and lower cooling capacity, saving about $100∼$300 per kW of the cooling capacity [16].

In spite of these promising advantages, state-of-the-art coarse-grained warm water cooling would be highly inefficient for edge datacenters due to the severe hotspot issue [13] at multiple levels. On the one hand, the imbalanced hardware utilization as well as different thermal specifications of heterogeneous hardware leads to thermal imbalance across servers and hardware. To cool down even a small portion of hotspot hardware components for their safety, the inlet water for every hardware component should be chilled to a very low temperature synchronously. This over-provisioning strategy is exceedingly inefficient since the centralized chiller needs to consume extra energy to provide cold water to other non-hotspots at the same time [13], [14], [17]. One possible solution might be installing distributed chillers or pumps for each hardware component, but the high costs, additional space demands, and other technical problems make this infeasible [13], [18]. On the other hand, the thermal imbalance also exists inside a hardware component because of different thermal specifications and imbalanced utilization of internal units. As shown in later Fig. 2 and Sec. 2.2, the temperature difference inside a CPU core can be over 20°C, and the value exceeds 30°C inside a GPU. Concerning the temperature measurement granularity, undetected localized hotspots will not only result in performance degradation and a shorter lifespan, but also increase the cooling costs by over-cooling other non-hotspot units [19], [20], [21]. Thus, if the above multiple-level hotspots can be dispersed effortlessly, the cooling efficiency can be further improved while ensuring the safety.

In summary, conventional cold water cooling wastes vast amounts of unnecessary energy in cooling many low-utilization servers, while coarse-grained warm water cooling raises the hotspot issue. To make the best of their advantages while avoiding these negative impacts, we propose CoolEdge, a fine-grained warm water cooling system for relieving multiple-level hotspots and saving cooling energy of high-density and heterogeneous edge datacenters. Specifically, we make the following contributions:

- We summarize the new challenges to efficiently cool the high-density and heterogeneous edge datacenters, and argue that a solution to the hotspot issue is extremely urgent due to the rapid growth of edge computing.
- We put forward a hotspot-relievable warm water cooling architecture CoolEdge with two major innovations. Specifically, through fine-grained cooling control under well-designed water circulations, hardware-level hotspots can be eliminated with high efficiency; by means of our newly developed cold plates with vapor chambers, chip-level hotspots can be dispersed without manual intervention.
- To the best of our knowledge, we are the first to theoretically quantify energy savings achieved by warm water cooling. Based on the quantification, we propose a custom-designed cooling solution using the Heat Dissipation Oriented (HDO) or Chiller Power Oriented (CPO) strategy to provide cooling setting adaptation to heterogeneous hardware while improving the cooling efficiency.
- Based on CoolEdge, we further develop a semi-fine-grained cooling solution CoolEdge$^+$ to lower the capital expenditures for implementing the customized cooling control. By employing a dynamic power capping approach, CoolEdge$^+$ can achieve comparable cooling efficiency improvement as CoolEdge but incurs lower costs, and enables a flexible trade-off between cooling energy consumption and hardware performance while ensuring the safety.
- We build a hardware prototype to validate the practicability of CoolEdge and CoolEdge$^+$, and conduct datacenter-level simulations to show their remarkable performance in balancing multiple-level hotspots and saving cooling costs. The evaluation results reveal that compared with baselines, CoolEdge reduces 81.81% of the cooling energy at most, and CoolEdge$^+$ behaves similar in energy savings. A cost saving analysis estimates that CoolEdge$^+$ can save up to $3,598,400 yearly in a city, 35.24% higher than CoolEdge.

## 2 Background and Motivation

In this section, we first recap existing cooling techniques and discuss their limitations of achieving high cooling efficiency in edge datacenters. Then, we disclose the hotspot issue from both the hardware level and the chip level, and explain our motivation to design a new cooling architecture in the end.

TABLE 1: Thermal specifications of some IT hardware components

| Hardware type | Intel Xeon E5-2680 v4 CPU [22] | Nvidia GeForce RTX 2080 Ti GPU [23], [24] | Nvidia A100 80GB PCIe GPU [25], [26] | DRAM [11] | Samsung 983 DCT SSD [27] |
|---|---|---|---|---|---|
| **MOT (°C)** | 86 | 89 | 100 | 85 | 70 |
| **TDP (W)** | 120 | 250 | 300 | Typically $\leq 10$ | Read: 8.7, Write: 10.6 |

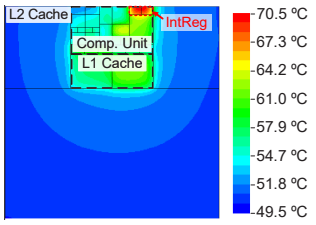



Fig. 2: Temperature distribution inside a CPU core.

### 2.1 State-of-the-Art Cooling Techniques VS. Demands of Edge Datacenters

First of all, we present a brief synopsis of existing cooling techniques, and analyze their inefficiency from the perspective of three requirements of edge datacenters: *proximity to end-users*, *high density*, and *heterogeneity*.

**Free cooling vs. proximity to end-users:** Free cooling, a way of directly cooling servers by free cooling sources, e.g., the outdoor air and lake water, has been well studied in recent years [28]. A growing number of cloud datacenters are built in cold and dry areas, or near the sea or lake, with free coolers like the dry cooler for access to cold air or water. For example, Iceland has become one of the world's most cost-effective destinations for datacenters owing to its ideal weather [29], and Microsoft even built its datacenter under the sea [30]. However, in order to provide low-latency services, edge datacenters should be sited in proximity to end-users and widely distributed in cities, which can hardly meet the strict requirements of free cooling.

**Air cooling vs. high density:** The contradiction between the increasing demand for edge datacenters and the shortage of urban land forces servers to be stacked up at a higher density. This dramatically increases the cooling demand since it becomes trickier to timely take away the heat, especially when the servers run at 100% utilization for sustained periods. To ensure high performance of edge applications, air cooling struggles to satisfy the strict cooling demand at such a high power density [31], because of its low heat conduction capacity and the difficulty in managing the airflow efficiently when the rack and aisle are increasingly compact. Although there are some techniques using elaborate engineering on the air-aisle arrangement, like the circular pattern of racks designed by Vapor IO [32] and the high-density cooling proposed by Intel [33], these techniques show poor performance in efficiency, scalability, and/or adaptivity for edge datacenters.

**Water cooling vs. heterogeneity:** Water cooling emerges as an energy-efficient paradigm for datacenters. As water has a higher density and greater thermal capacity per unit volume than air, water cooling supports a higher power density [34]. Currently, cloud datacenters mainly use direct-to-chip cooling [12]. As shown in Fig. 1, a cold plate with water flowing inside is directly pressed on the surface of a hardware component to absorb the heat. After absorbing heat from different hardware components on different branches, the water gathers together and then is cooled to some temperature by the centralized chiller (also by the cooling tower in some large-scale cloud datacenters). For the safety of hardware components, this temperature is usually set low enough so as to cool down some high-utilization hardware components with a high temperature. In this coarse-grained water cooling system, however, since different hardware components share the same inlet water temperature and flow rate in spite of their specific cooling demands, a lot of cooling energy would be wasted, showing excessively low efficiency.

Nowadays, some cloud providers propose the warm water cooling technique to reduce cooling costs. However, the coarse-grained warm water cooling suffers from a severe hotspot issue, which we will expatiate in the following.

### 2.2 The Hotspot Issue in Edge Datacenters

Compared with cloud datacenters, the hotspot issue becomes more severe in edge datacenters due to the requirements of high density and heterogeneity, along with skewed hardware utilization of edge workloads [9] and non-ideal ambient conditions of the edge. Typically, there are two kinds of thermal imbalance in edge datacenters, i.e., at the hardware level and the chip level.

**The hotspot issue at the hardware level:** Previous works [13], [35] have shown the hotspot issue exists among homogeneous hardware (e.g., CPUs or dynamic random-access memories (DRAMs) of the same type). For heterogeneous hardware, the thermal imbalance becomes more significant due to their divergent thermal specifications and dynamic characteristics. Table 1 illustrates their thermal specifications of Maximum Operating Temperature (MOT) and Thermal Design Power (TDP) [36]. As we can see, there are vast differences in both MOT and TDP from one hardware type to another, especially between the computing hardware and the memory or storage hardware. We also evaluate the dynamic characteristics of heterogeneous hardware components when changing their load levels. As plotted in Fig. 3[1], these hardware components show different operating temperatures and temperature variation rates in the same status. Usually, the operating temperature of computing hardware is above 40°C, while the operating temperature of memory hardware is lower than 40°C. When changing their statuses, the temperature of computing hardware goes up/down significantly faster than that of the memory, and reaches a stable level more quickly. Hence, it is essential to design a custom-designed solution for heterogeneous hardware, which will be introduced in Sec. 4.4.

**The hotspot issue at the chip level:** Considering the hardware type and workload characteristics, different internal units inside the hardware component may be at different utilization and power levels, which brings the hotspot issue at the chip level. We investigate this hotspot issue under four

1. The details of hardware components are presented in Section 6.1.

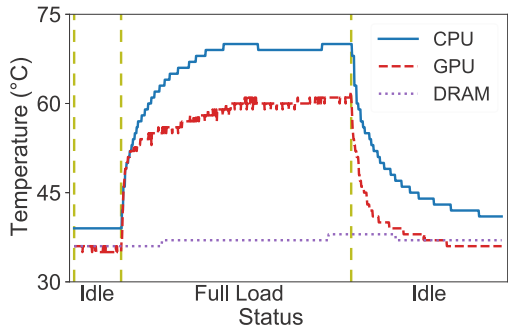



Fig. 3: Temperature variation of heterogeneous hardware.

cases: among CPU cores, inside a CPU core, inside a GPU, and inside a DRAM, respectively.

(1) Hotspots among CPU cores: A CPU usually contains many processing units, i.e., cores. A prior work shows their average temperature imbalance can be over 7°C [21]. Our experimental results in later Fig. 21 also reveal these hotspots.

(2) Hotspots inside a CPU core: A micro CPU core contains several parts from low-powered cache units to high-powered computing units. We use the HotSpot simulator [37] to acquire temperature distribution inside a CPU core, as presented in Fig. 2. In particular, the temperature difference between computing units and cache units can be over 20°C. When running integer workloads, there exist several hotspots especially in the integer register marked as IntReg in Fig. 2.

(3) Hotspots inside a GPU: A GPU consists of multiple units including computing units, memory units, etc. According to the measurement result of an AMD GPU for a stress test, inside the GPU, the hotspots can reach above 100°C and the maximum temperature difference is over 30°C.

(4) Hotspots inside a DRAM: A DRAM is mainly composed of several DRAM chips and one buffer chip which have different rated temperatures [35]. According to the previous research, the temperature difference among DRAM chips is over 15°C while the value between DRAM chips and the buffer chip can reach more than 30°C [35], [38].

### 2.3 Why We Need a New Cooling Architecture to Address the Hotspot Issue for Edge Datacenters?

Many software-based solutions can be implemented to relieve hotspots in a cloud datacenter, including power throttling [20], [39], [40], [41], workload deferral [42], and workload balancing [11], [20], [21], [35], [43], [44]. However, it is usually necessary to consider the tradeoff between the performance guarantee and hotspot relieving. For example, avoiding hotspots by lowering hardware frequency is likely to degrade hardware performance. Also, since some mission-critical edge applications, such as smart traffic management [45], would have no deferrable workloads, the hotspots may unavoidably emerge constantly. As a result, it is necessary to propose a workload-agnostic solution to the hotspot issue for general cases at the edge. Since these software-based solutions require no hardware-level support and have access to higher level information, such as workload deadlines, they can be combined with this workload-agnostic solution to further reduce energy usage through relieving hotspots.

Recently, Jiang et al. [13] propose a thermoelectric cooler-based (TEC-based) solution to address the hotspot issue in a homogeneous cloud datacenter with only CPUs. Specifically, the authors apply warm water to cool CPUs equally and integrate each CPU with a TEC to provide extra cooling capacity for hotspots. However, the TEC-based solution fails to meet the specific requirements of edge datacenters, i.e., high density and heterogeneity. On the one hand, it requires considerable modifications to the internal structure of servers, which is somewhat impractical for already high-density servers. In particular, given the high energy demand of the TEC, it will be disabled when the CPU becomes a non-hotspot. In this case, a copper plate of double the CPU's size and an additional cold plate are necessary for transferring heat. On the other hand, it cannot be extended to support heterogeneous hardware due to the following reasons. First, the required copper plate cannot cater to the physical layouts of various hardware types. Taking the GPU as an example, as several large capacitors are scattered around the computing units, there is no room for this copper plate. Second, the TECs cannot be applied to high-powered hardware due to their limited cooling capacity. The maximum heat load that an economical TEC can transfer effectively is usually less than 150 W [46], hardly meeting the cooling demand of high-powered hardware like GPUs whose TDP can reach 400 W [26].

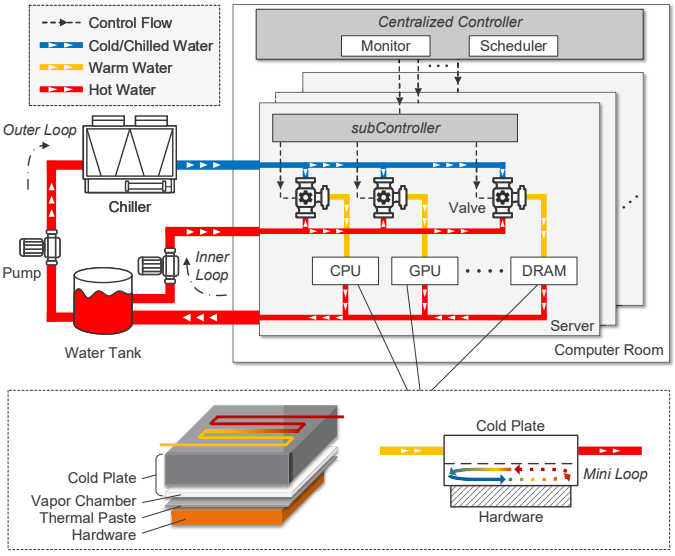



Fig. 4: Hotspot-relievable warm water cooling system.

In short, prior works are inefficient for latency-sensitive edge workloads or lack support for the heterogeneity of edge datacenters. Differing from them, we propose a warm water cooling system tailored to owner-operated edge datacenters with full consideration for heterogeneous hardware.

## 3 System Architecture

In this section, we formally propose CoolEdge, a hotspot-relievable warm water cooling system for edge datacenters. We begin with the system overview and then elaborate on the design details.

### 3.1 System Overview

As shown in Fig. 4, each server consists of many heterogeneous hardware components, including CPU, GPU, DRAM,

etc. There are three major parts in our proposed cooling system: Inner-and-Outer Loop, Mini Loop, and controllers.

(1) Inner-and-Outer Loop contains two water circulations, i.e., Inner Loop and Outer Loop, to cool the hardware components. In particular, Inner Loop is a hot water circulation directly recycling the "used" water after cooling the hardware; Outer Loop is a cold water circulation where the hot water is pumped to the chiller and turned into "refreshed" chilled water again. Unlike the conventional water cooling system, we use a valve to provide an appropriately customized inlet water temperature and flow rate for each hardware component, by mixing a certain amount of hot water from Inner Loop and cold water from Outer Loop.

(2) Mini Loop is a small vapor-fluid circulation inside a two-phase vapor chamber, which is creatively deployed on the cold plate to enhance thermal conductivity and reduce local hotspots inside the hardware component.

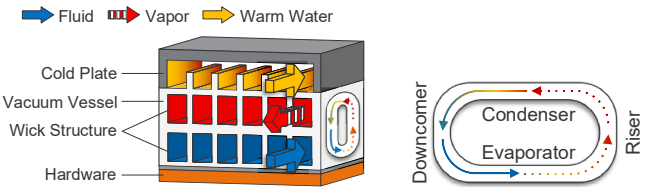
(3) Controllers include a Centralized Controller and multiple subControllers, i.e., one subController for each server. Specifically, based on the information collected in real time (e.g., hardware utilization and temperature) and the specific cooling strategy (introduced in Sec. 4), the Centralized Controller periodically decides on the best cooling setting, i.e., the inlet water temperature and flow rate for each hardware component. Then it sends the control command to each subController to adjust water temperature and flow rate accordingly.

### 3.2 Inner-and-Outer Loop: Hardware-Level Hotspot Elimination with Mixed Water

As illustrated in Sec. 2, both homogeneous and heterogeneous hardware components have different cooling demands at different times. In order to handle hotspots among different hardware components, we design two water cooling circulations, i.e., Inner Loop and Outer Loop, to achieve fine-grained and flexible cooling control in an edge datacenter. Specifically, we use a pulse-width modulation (PWM) controlled proportional solenoid valve [47] at the inlet of each hardware component. The water temperature and flow rate can be regulated at the desired values by mixing different amounts of hot and cold water based on each hardware component's real-time cooling demand.

As plotted in Fig. 4, Inner Loop is a hot water circulation, gathering "used" water from the outlet of each hardware component to the water tank and pumping it to the inlet again. Since the hot water from Inner Loop cannot cool down some high-utilization hardware components, Outer Loop pumps hot water from the water tank to the chiller and then sends the chilled water to the inlet. Based on the control command from controllers, different amounts of hot water from Inner Loop and cold water from Outer Loop will be sent to the hardware component. As compared to merely sending the chilled water, the mix of both hot and cold water not only reduces the required amount of chilled water and saves cooling energy at the current time, but also increases natural heat dissipation (discussed in Sec. 4) which in turn saves cooling energy thereafter.

As the proportional valves are somewhat costly (the purchase price is about $30 for each hardware component [48]), they can be replaced with economical on/off valves (about $14 for each hardware component [49]) to save capital expenditures. As an on/off valve either allows unimpeded flow or stops flow completely, only three discrete water temperature values can be regulated when using two such valves by allowing hot water only, cold water only, or the mix of both the hot and cold water that generates warm water. However, directly simplifying the fine-grained cooling architecture shown in Fig. 4 to this *semi-fine-grained* one can reduce the cooling efficiency improvement largely. To keep high cooling efficiency, we devise a dynamic cooling control mechanism with a power capping approach which considers the cooling demand and computing performance jointly by adjusting the maximum allowed hardware power. We call this solution as CoolEdge$^+$ which will be detailed in Sec. 5.




(a) Structure of the cold plate integrated with the vapor chamber (b) Working principle

Fig. 5: Illustration of Mini Loop.

### 3.3 Mini Loop: Chip-Level Hotspot Dispersion with Two-Phase Vapor Chambers

To relieve hotspots inside a hardware component, we integrate a two-phase vapor chamber into the cold plate and realize vapor-fluid Mini Loop inside the chamber. As shown in Fig. 4, the cold plate is attached to the hardware component to transfer heat into the cooling water. Between the hardware component and the cold plate is the thermal paste, used to eliminate air and thus provide higher thermal conductivity. It is worth noting that the vapor chamber is typically standalone, and attached between a heat source and a cooling component to conduct heat directly. However, we find this is exceedingly inefficient in transferring heat from a hardware component to the cooling water inside an intact cold plate, due to the long thermal path and an extra layer of the thermal paste. Therefore, instead of directly attaching the vapor chamber to the bottom of the cold plate, we replace the commonly used cold plate's baseplate with the vapor chamber for achieving higher thermal conductivity, as shown in later Fig. 11b. We also perform an experiment to verify this observation, and the results are presented in Appendix A. Now we introduce the physical structure, working principle, and attractive characteristics of the vapor chamber as follows.

**Physical structure:** As shown in Fig. 5a, the vapor chamber consists of a sealed vacuum vessel and an internal wick structure. The outside of the vacuum vessel is typically made of copper or aluminum to achieve high thermal conductivity, while the wick structure contains a small amount of working fluid in equilibrium with its vapor to transfer heat from one side of the chamber to the other [50]. Another performance metric influencing the thermal conductivity is the filling ratio, defined as the ratio of the volume of the working fluid out of the total volume of the vapor chamber [51]. Usually, the value is set at 20%∼45% [52].

**Working principle:** As shown in Fig. 5b, the vapor chamber includes an evaporator and a condenser. The evaporator consists of a wick structure, where the working fluid flows. The fluid absorbs heat from hardware components, and immediately vaporizes and rises to the condenser driven by the pressure difference. Once arriving at the condenser, the vapor condenses again and releases the latent heat of condensation to the cooling water. The condensed fluid finally returns to the evaporator by the capillary action of the wick structure and also by gravity [50]. Through this heat circulation, the hotspots inside each hardware component can be dispersed automatically and a relatively uniform temperature distribution can be realized.

**Attractive characteristics:** Compared with the conventional single-phase cooling system, there are several benefits from the hotspot-relievable two-phase vapor chambers. First, because of the huge latent heat of vaporization, the vapor chamber has higher thermal conductivity than the commonly used cold plate and thus reduces the hardware temperature to a lower level. To this extent, the inlet warm water temperature can be raised further to save more cooling energy. Second, as the thermal conductivity of vapor chambers grows with the power density of the heat source, more heat can be absorbed from hotspot areas inside the hardware component [53], [54]. In all, the vapor chamber not only increases the general thermal conductivity, but also smooths the temperature distribution of the heat source in an automatic manner. In Sec. 6.5, we conduct several experiments to demonstrate these characteristics using our newly developed vapor chamber-based cold plate.

In addition, the two-phase vapor chamber also has many unique properties desirable to edge datacenters:

- Space-saving: The vapor chamber is usually as thin as a copper baseplate of the commonly used cold plate (e.g., $2\sim3\,\mathrm{mm}$) but much lighter.
- Cheap: The vapor chamber can be bought for about \$5 from *Alibaba.com* [55]. Besides, about \$1 for the copper baseplate can be saved.
- Reliable: The Mean Time Before Failure (MTBF) of the vapor chamber is 80,000 hours (i.e., about 10 years) [50], larger than the hardware lifespan.
- Environment-friendly: The vapor chamber consumes no power, and all the raw materials (e.g., purified water as the working fluid) are environment-friendly, significant for today's green datacenter infrastructure.

### 3.4 Controllers

In CoolEdge, there is a Centralized Controller and each server is directly controlled by an independent subController.

**Centralized Controller:** The Centralized Controller is composed of two parts, i.e., a monitor and a scheduler. In each adaptation period, the monitor first collects each hardware component's temperature and utilization/power information from subControllers. Then, the scheduler decides the best inlet water temperature and flow rate for each hardware component based on the collected information and the cooling strategy (introduced in Sec. 4). Finally, the scheduler sends the cooling control commands back to each subController.

**subController:** Each subController periodically collects the temperatures of CPUs and DRAMs with the `lm_sensors` tool, utilization of CPUs and DRAMs by the `/proc filesystem`, and temperatures and powers of GPUs with the `nvidia-smi` tool, and then sends them to the Centralized Controller. Once receiving the control command from the Centralized Controller, each subController sends the control signal to the valves installed on the server. By connecting the valves to a 4-pin power connector on the motherboard, it is convenient to tune each valve by the PWM signal [47], and the relationship between the water flow rate and duty cycle is presented in [47], [56]. As the response time of such a valve is less than 1 s [57], real-time control can be achieved. Besides, such a valve's power consumption is only several Watts [58], which is negligible compared with the IT and cooling equipment.

## 4 Fine-Grained Cooling Solution

In this section, we first theoretically quantify the power saving achieved by the warm water cooling. According to the quantification, we propose two warm water cooling strategies to decide on the best cooling setting. At last, we present a custom-designed cooling control solution for heterogeneous hardware components.

### 4.1 Key Proposition of Warm Water Cooling

The state-of-the-art literature on warm water cooling states that increasing the inlet water temperature has great potential for saving cooling energy [13], [14], [15]. As the warm water temperature is higher than the ambient temperature, there exist significant natural heat dissipation phenomena in pipes and tanks, lowering the cooling energy consumption of the chiller compared with the cold water cooling. In this section, we provide the first theoretical analysis on the efficiency of warm water cooling from the aspect of natural heat dissipation. In the following part, we take the heat dissipation in pipes as the representative, since the tank can be viewed as a wider pipe and analyzed in a similar way. All the theoretical derivations and detailed discussions are provided in Appendix B.

***Proposition 1.*** The natural heat dissipation efficiency depends on (1) $\Delta T$: the temperature difference between the cooling water in the pipe and the outer air, (2) $v$: the water flow rate, (3) $h$: the convective heat transfer coefficient of the air, and (4) $\xi$ and $\mu$: parameters related to physical characteristics of the pipe and water (e.g., the pipe's radius and the density of water), respectively. Based on Fourier's law of heat conduction [59] and Newton's law of cooling [60], the dissipated heat $P$ (in Watts) through the pipe can be calculated by:

$$P = \xi v \Delta T\big(1 - \exp(-\mu h/v)\big), \tag{1}$$

Eq. (1) indicates that compared with the water flow rate, increasing the water temperature contributes significantly to the heat dissipation under the same ambient condition. The convective heat transfer coefficient of the air $h$ represents the thermal resistance of a relatively stagnant layer of air between a pipe surface and the air medium [61]. Higher air velocity increases $h$ and thus the dissipated heat. The

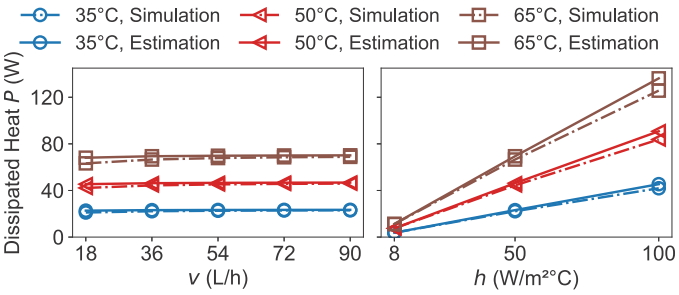



Fig. 6: The effect of the inlet water temperature, $v$, and $h$ on $P$ based on the simulation and estimation.

convective heat transfer coefficient of the free air is usually 2.5∼25 $\mathrm{W/m^2°C}$, while in the forced convection environment, e.g., with fans blowing around, the value can be up to 10∼500 $\mathrm{W/m^2°C}$ [62].

Using Fluent software [63], we simulate the process that the water flows through a 1-meter copper pipe when the ambient air temperature is stable at 20°C. Fig. 6 shows the effect of the inlet water temperature on the heat dissipation under varying $v$ and $h$ from both the simulation results and estimation results based on Eq. (1). Note that when the inlet water temperature and $h$ are high enough, the amount of dissipated heat can be equal to a CPU's TDP (as listed in Table 1), showing the great potential of warm water cooling to save cooling energy. Although we find that the estimation of heat dissipation is generally accurate, there is still an estimation error of at most 8.4% when the inlet water temperature and $h$ are high. The main reason is that Eq. (1) cannot accurately describe the temperature of the air and the water near the pipe wall [64]. Here we use an attenuation factor $\beta$ to make up the estimation error, whose detailed expression is introduced in Appendix B.

Proposition 1 analyzes the key factors affecting the water cooling efficiency. We find that from the aspect of natural heat dissipation, a higher temperature of the cooling water shows a remarkable effect on increasing cooling efficiency. To sum up, the warm water cooling saves more cooling energy as more heat can be dissipated in a natural way, and the amount of dissipated heat is affected by several factors.

### 4.2 Heat Dissipation Oriented Strategy

Based on Proposition 1, we find that the inlet water temperature and its flow rate are two tunable factors that determine the natural dissipated heat $P$ presented in Eq. (1) and thus the cooling efficiency. To make full use of natural heat dissipation, the cooling system should cool each hardware component with the best cooling setting of the inlet water temperature and flow rate. Let $T_{hot}$ and $T_{cold}$ represent the temperatures of inlet hot water and inlet cold water, respectively. For the hardware component $i$ whose power is $P_i$, the mixed inlet warm water temperature is denoted by $T_{warm,i}$. As discussed in Sec. 3, the temperature and flow rate of both inlet hot water from Inner Loop and cold water from Outer Loop together determine the temperature of the inlet warm water. We use $v_{hot,i}$ and $v_{cold,i}$ to represent the flow rate of the inlet hot water and inlet cold water, respectively. The temperature and flow rate of the inlet warm water for the hardware component $i$ is given by $T_{warm,i} = (v_{hot,i}T_{hot} + v_{cold,i}T_{cold})/(v_{hot,i} + v_{cold,i})$ and $v_{warm,i} = v_{hot,i} + v_{cold,i}$, respectively. Based on the law of conservation of energy [65], the temperature of the hot water from the $i$-th outlet $T_{out,i}$ can be calculated by:

$$T_{out,i} = T_{warm,i} + \frac{P_i}{c\rho v_{warm,i}}, \tag{2}$$

where $\rho$ and $c$ represent the density and specific heat capacity of the inlet water, respectively.

Based on Eq. (1), and $T_{out,i}$ and $v_{warm,i}$ for each hardware component, the dissipated heat (in Joule) of the pipe $E_{pipe} = P_{pipe}t = \sum_i P_{pipe,i}t$ and the tank $E_{tank} = P_{tank}t$ can be calculated accordingly, where $t$ is the time slot. Then, we can obtain the best cooling setting of the water temperature and flow rate for each hardware component to maximize the amount of dissipated heat $R_{HDO}$:

$$R_{HDO} = \frac{E_{pipe} + E_{tank}}{COP_c} - E_{pump}, \tag{3}$$

where $COP_c$ is the coefficient of performance (COP) [66] of the chiller, and $E_{pump}$ is the power consumption of the pump, whose expression is presented in Sec. 6.2. Since we maximize natural heat dissipation here, we call this cooling strategy as **Heat Dissipation Oriented (HDO)** strategy.

### 4.3 Chiller Power Oriented Strategy

Although the HDO strategy provides the precise warm water cooling setting, it could be impractical to achieve in reality. For instance, some parameters in Eq. (1), such as the size of pipes and $h$, are inaccessible or varying dynamically. Moreover, the heavy computational overhead would also affect the edge's performance. To ensure the practicability of the cooling management, we present a simple and more general strategy.

It is obvious that the lower warm water temperature and higher flow rate we set, the larger volume of chilled water will be used to cool the hardware component within a given time slot, and hence the temperature of outlet hot water is lower. In this case, according to Proposition 1, the amount of dissipated heat would fall, reducing the overall cooling efficiency. As a result, we regard the cooling setting with the least chilled water provision, i.e., the least energy consumption of the chiller as the best choice and call this **Chiller Power Oriented (CPO)** strategy. Here, we can obtain the best cooling setting of water temperature and flow rate for each hardware component to minimize the chilled water provision $R_{CPO}$:

$$R_{CPO} = \sum_i v_{cold,i} = \sum_i v_{warm,i} \cdot \frac{T_{hot} - T_{warm,i}}{T_{hot} - T_{cold}}. \tag{4}$$

### 4.4 Fine-Grained Cooling Control

After introducing the two strategies for improving the efficiency of warm water cooling, we present a custom-designed cooling control solution for heterogeneous hardware here. Since the computing hardware (e.g., CPU and GPU) shows different thermal specifications and dynamic characteristics from the memory hardware (e.g., DRAM) as discussed in Sec. 2.2, we quantify their thermal profiles at first.

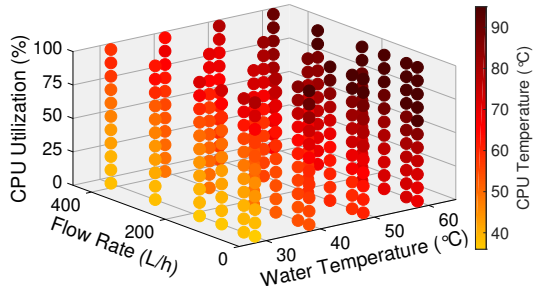



Fig. 7: Measurement results of CPU temperature with different inlet water temperature, flow rate, and CPU utilization.

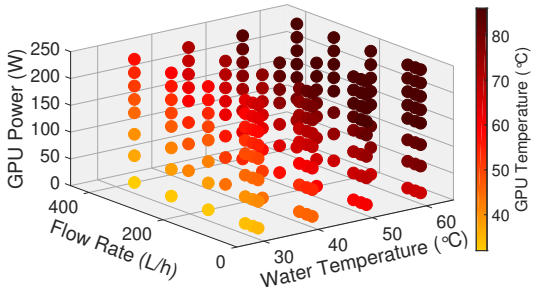


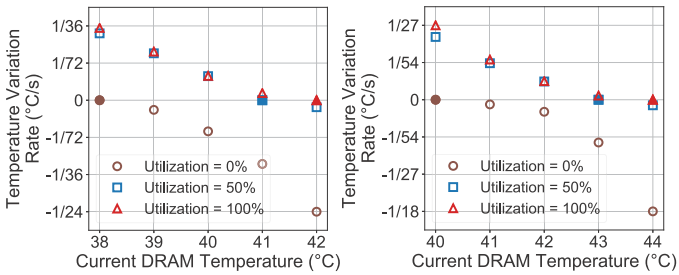

Fig. 8: Measurement results of GPU temperature with different inlet water temperature, flow rate, and GPU power.

According to our findings in Sec. 2.2, the inlet water temperature and flow rate directly impact the instantaneous temperature of computing hardware, so we only consider their real-time utilization/power to make cooling decisions. The measurement results of the CPU and GPU temperature are plotted in Figs. 7 and 8, respectively.

For DRAMs, since their power is excessively low as shown in Table 1, the generated heat can be timely taken away whatever the inlet water flow rate is, and the inlet water temperature directly influences the DRAM temperature variation rate instead of the instantaneous DRAM temperature. Here, we define the temperature variation rate as the reciprocal of the time spent to raise or reduce the hardware temperature by 1°C under specific inlet water temperature, hardware temperature, and hardware utilization. Next, we measure the relationships between the DRAM temperature variation rate and these three variables as shown in Fig. 9. The positive and negative values of the temperature variation rate mean an increase and decrease in the hardware temperature, respectively. The absolute value of the temperature variation rate grows exponentially with the temperature deviation from the stable temperature, making the temperature quickly reach near the stable temperature.

Based on the above hardware profiles and the received information from subControllers, the scheduler decides the best cooling setting for each heterogeneous component periodically with a custom-designed solution as follows:

- Computing hardware: For each hardware component, the scheduler first selects possible choices of the inlet water temperature and flow rate from the profiles (e.g., the measurement results in Fig. 7 and Fig. 8) which guarantee the hardware temperature lower than the safe operating temperature. Leveraging the HDO or CPO strategy, the scheduler can obtain the best warm water cooling setting by maximizing Eq. (3) or minimizing Eq. (4).
- Memory hardware: Since only the inlet water temperature is tunable for DRAMs, both HDO and CPO strategies will choose the same setting—maximizing the inlet water temperature. Thus, for each inlet water temperature, the scheduler calculates the DRAM temperature at the next adaptation period based on the measurement results in Fig. 9 and chooses the best cooling setting of the water temperature with the safe operating temperature guarantee.




(a) Water temperature = 34°C (b) Water temperature = 37°C

Fig. 9: DRAM temperature variation rate (the solid points represent the stable temperatures).

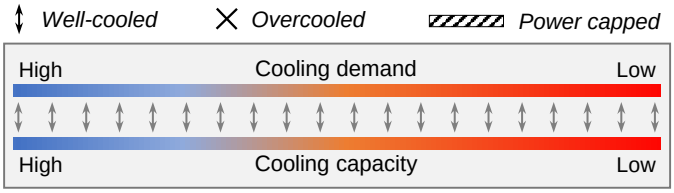
According to our single-threaded testing on a server with an Intel Xeon E5-2697 v4 CPU, it takes about 20 ms for the HDO or CPO strategy to obtain the best cooling setting for each hardware component. Leveraging parallel computing, we can easily scale up the computing speed when the number of hardware components increases greatly. After determining all the cooling settings, the scheduler will send cooling control commands to each subController. Then the subController will send the control signal to each valve at once to adjust the amounts of hot water and cold water sent to each hardware component accordingly.

## 5 Semi-Fine-Grained Cooling Solution

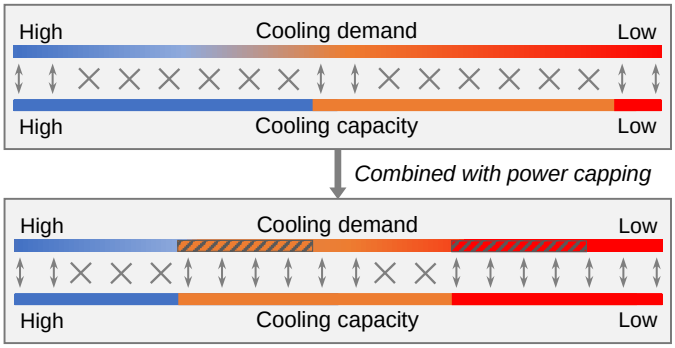
In this section, we propose a semi-fine-grained cooling solution CoolEdge$^+$ that reduces the cooling complexity of CoolEdge. We first introduce the main difference in the cooling control mechanism between CoolEdge and CoolEdge$^+$, and then present the details of the semi-fine-grained cooling control.

### 5.1 Cooling Control Mechanism

As presented in Sec. 3.2, although the proportional valves help achieve fine-grained cooling control by mixing any amounts of hot and cold water, they incur high capital expenditures. For some excessively underutilized datacenters, the energy savings might not offset the capital expenditures of valves as expected. Therefore, we design a semi-fine-grained cooling solution named CoolEdge$^+$ which replaces the original proportional valves with simpler on/off ones. Instead of customizing arbitrary cooling water temperatures as CoolEdge does, this solution can regulate three discrete water temperature values only, by allowing hot water only, cold water only, and the mix of both the hot and cold water




(a) CoolEdge always meets the cooling demand




(b) Combined with the power capping approach, CoolEdge$^+$ can largely meet the cooling demand though only three water temperature values are accessible

Fig. 10: The cooling control mechanism of CoolEdge vs. CoolEdge$^+$.

in a fixed ratio. To avoid potential efficiency drop because of over-cooling, we also integrate a service level objective-aware (SLO-aware) power capping approach into the design. By allowing limited performance degradation (e.g., 5%) through power capping, the cooling demand can be reduced slightly to match the cooling capacity provided by the cooling water under one of the three possible temperatures. Fig. 10 summarizes the difference in the cooling control mechanism between CoolEdge and CoolEdge$^+$. As shown in Fig. 10b, leveraging the well-managed power capping approach, CoolEdge$^+$ could avoid over-cooling significantly and thus achieves similar cooling efficiency as CoolEdge.

### 5.2 Semi-Fine-Grained Cooling Control

According to the above control mechanism, we present the control details of CoolEdge$^+$ here. In the offline phase, based on the measurement results in Figs. 7 and 8, we build a power model $P = M_P(T_{water}, T_{safe})$ for each hardware type to estimate the maximum allowed power consumption $P$ under its safe operating temperature $T_{safe}$ when cooled by the water at temperature $T_{water}$. Note that in the semi-fine-grained cooling system with on/off valves only, all components will share the same water flow rate, so we do not consider the flow rate in the power model and regard it as a fixed value. We define the ratio of the flow rates of the hot water to the cold water as $\alpha$, a hyperparameter that influences the warm water temperature when mixing the hot and cold water. We also build a latency model $L = M_L(P, i)$ to obtain the processing latency of the $i$-th task type (e.g., deep neural network inference) under the power limit of $P$.

In the online phase, for each incoming request of the $i$-th task type with the latency constraint of $L_{SLO}$ (Line 2), the Centralized Controller first records its metadata (e.g., the task type $i$ and the processing latency without power capping) and updates the temperature of the hot water in the water tank $T_{hot}$ (Lines 3-4). Then, for each of the three water temperature values in descending order, the Controller will estimate the maximum allowed hardware power $P$ and the processing latency of the $i$-th task type under the power limit of $P$ (Lines 5-7). Once the processing latency is within $L_{SLO}$, the request will be scheduled, and the Centralized Controller will send the cooling control commands to the corresponding subController (Lines 8-12). Finally, to avoid cooling failures when the hot water temperature rises as time goes by, every $C$ time period the Centralized Controller will perform a global adjustment to all valves by repeating the above cooling steps (Lines 14-26).

**Algorithm 1** Semi-fine-grained cooling control (CoolEdge$^+$)

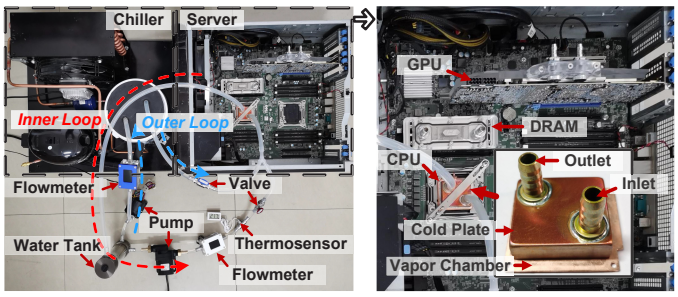
1: Initialize: the list $R$ recording all the running tasks, the temperature of the chiller water from the chiller $T_{cold}$, the temperature of the hot water directly from the water tank $T_{hot}$, the ratio of the flow rates of the hot water to the cold water $\alpha$, the power model $P = M_P(T_{water}, T_{safe})$, and the latency model $L = M_L(P, i)$.
2: **while** a request $r$ of the $i$-th task type with the latency constraint of $L_{SLO}$ arrives **do**
3: Record $r$ in $R$;
4: Update $T_{hot}$ according to the temperature reading;
5: **for** $T_{water} = T_{hot}, \frac{\alpha T_{hot}+T_{cold}}{\alpha+1}, T_{cold}$ **do**
6: Estimate $P = M_P(T_{water}, T_{safe})$;
7: Estimate $L = M_L(P, i)$;
8: **if** $L \leq L_{SLO}$ **then**
9: **break**;
10: **end if**
11: **end for**
12: Dispatch the request, and tune the valves and pumps based on $T_{water}$;
13: **end while**
14: **for** Every time period of length $C$ **do**
15: Update $T_{hot}$ according to the temperature reading;
16: **for** $r$ in $R$ **do**
17: **for** $T_{water} = T_{hot}, \frac{\alpha T_{hot}+T_{cold}}{\alpha+1}, T_{cold}$ **do**
18: Estimate $P = M_P(T_{water}, T_{safe})$;
19: Estimate $L = M_L(P, i)$;
20: **if** $L \leq L_{SLO}$ **then**
21: **break**;
22: **end if**
23: **end for**
24: Tune the valves and pumps based on all $T_{water}$;
25: **end for**
26: **end for**

## 6 Evaluation

In this section, we first introduce our well-established hardware prototype. Based on the collected hardware profiles, we then conduct extensive simulations with real-world traces to evaluate CoolEdge and CoolEdge$^+$ in terms of energy and cost savings. Finally, we present our further experiments on advanced vapor chamber-based cold plates.

### 6.1 Hardware Prototype

To verify the practicability of CoolEdge and CoolEdge$^+$ and collecting thermal profiles that are presented in Figs. 7, 8, and 9, we build up a hotspot-relievable warm water cooling prototype in a Dell Precision Tower 7910 Workstation [67],




(a) (b)

Fig. 11: Hardware prototype.

as illustrated in Fig. 11a. The cooling part contains Inner Loop and Outer Loop. Inner Loop is composed of a water tank, a pump, a flowmeter for monitoring the water flow rate, and a thermosensor for monitoring the temperature of inlet warm water from the water tank. Outer Loop consists of a pump, a flowmeter, and a chiller for providing chilled cold water. After mixing the hot water and cold water via the valves, customized warm water is sent to each hardware component. Fig. 11b shows the items in the server, including an Intel Xeon E5-2680 v4 CPU, an Nvidia GeForce RTX 2080 Ti GPU, and four G.Skill DDR4 DRAMs. Note that to make the illustration clear, we do not connect water pipes to all the hardware components, and take the CPU as a representative.

### 6.2 Evaluation Setup

We perform two experiments to evaluate the performance of CoolEdge (fine-grained, workload-agnostic) and CoolEdge$^+$ (semi-fine-grained, workload-aware), respectively, with different traces.

**Traces and workloads:** As real-world edge datacenters and edge traces are not accessible at the moment, to evaluate CoolEdge and CoolEdge$^+$ from the industrial perspective, we use the hardware utilization trace from SURFsara [68] and the workload trace from Alibaba PAI [69], respectively, to simulate the energy usage. The SURFsara trace includes the utilization or power information of 341 CPUs, 341 DRAMs, and 57 GPUs for about 3 months. To make the trace conform to the high-density and high-utilization characteristics of edge servers, we regard each server as a 2-way server and select the top-20 traces of CPUs, GPUs, and DRAMs separately based on their highest utilization or power within the selected 10 hours. That is, there are 10 servers considered in the evaluation, each equipped with 2 CPUs, 2 GPUs, and 2 DRAMs. Later Fig. 15 shows their utilization during the 10 hours. The Alibaba PAI trace contains high-level information of machine learning (ML) workloads during two months in a cluster with 6,500 GPUs, such as the task name, start time, and end time. We select the first seven days of the trace in the simulation. As the trace does not include the task type information, We divide the tasks into ten groups manually according to the remainder of its job name divided by 10, and assume that all the tasks in the same group are ML inference tasks on the same ML model, including ResNetV2-101, Inception, VGG16, EfficientNet-B3, EfficientNet-B5, EfficientNet-B7, YOLOv3, UNet, Pix2Pix, and XLNet. To build the power and latency models for each ML model type (i.e., task type) as mentioned in Sec. 5.2, we measure the average power consumption and inference latency of these models on one GPU in our hardware prototype under different power limits. The latency SLO is randomly set to 1.01$\sim$1.10$\times$ the inference latency without power capping. As the ML workloads are scheduled to GPUs only, we only consider cooling for GPUs when using this trace to evaluate CoolEdge$^+$.

**Simulation methodology:** In the datacenter-level simulation, all the thermal profiles of CPU, GPU, and DRAM required by the Centralized Controller are collected from the hardware prototype. This would not influence the feasibility and availability of the profiles considering the differences between servers and workstations are irrelevant to the water cooling efficiency [70]. We also take into account necessary physical infrastructure of edge datacenters in the simulation, including the length of pipes, and the sharing of one centralized chiller and two pumps in Inner Loop and Outer Loop. In addition to the cooling equipment, fans are considered to maintain the ambient temperature and improve the natural heat dissipation by increasing $h$. Considering that their energy consumption is much lower than the water cooling equipment, during the simulation, the total energy consumption $E_{total}$ is only the summation of the energy consumption of the centralized chiller $E_{chiller}$ and two pumps $E_{pump}$. The former can be calculated by $E_{chiller} = \frac{\rho ct(T_{hot}-T_{cold})\sum_i v_{cold,i}}{COP_c}$, and the latter can be approximately calculated by $E_{pump} = 1.3674vt$ according to our experimental results, where $v = \sum_i v_{warm,i}$ is the total flow rate of every water branch, and $t$ is the time slot. To validate the effectiveness of the fine-grained and semi-fine-grained cooling solutions, we integrate the vapor chambers in all the baselines in the simulation. Then, we will present a brief analysis of energy and cost savings from vapor chambers in the cost saving analysis.

**Baselines:** Since air cooling cannot meet the cooling demands of edge datacenters, we consider three water cooling baseline strategies as follows:

- Conventional coarse-grained water cooling system (*Coarse-grained*): For this baseline, we set the global water temperature and flow rate according to the highest cooling demand of all the components.
- State-of-the-art TEC-based water cooling system (*TEC*): Jiang et al. [13] equip each CPU with a TEC in a datacenter. Since this solution is infeasible in a heterogeneous edge datacenter as discussed in Sec. 2.3, we only consider cooling for CPUs in the comparison, and suppose there are two CPUs in each server. We use $E_{TEC}$ to indicate the energy consumption of TECs.
- Coarse-grained water cooling system with SLO-aware power capping (*ATAC*): ATAC [71] proposes a dynamic power capping solution to reduce cooling energy consumption by turning down power usage of hotspot components in an air-cooled datacenters. We apply this power capping solution to the water cooling system and control the number of tasks with a latency increase of more than 5% to be between 20% and 40% after capping the power. After that, we set the global cooling setting according to the highest cooling demand of all the components.

The first two baselines are workload-agnostic while the last one requires workload information. Thus, we compare CoolEdge with the *Coarse-grained* and *TEC* baselines only, and compare CoolEdge$^+$ with the *Coarse-grained* and *ATAC* baselines as well as CoolEdge.

**Parameter settings:** As prolonged operation at near MOT may degrade performance and shorten hardware lifespan [19], [72], for the CPU, we set the safe operating temperature as 90% of its MOT, while for the GPU, the value is set as 85% and 70% of its MOT when evaluating CoolEdge and CoolEdge$^+$, respectively, because of its higher temperature variation rate as shown in Fig. 3 and thus higher probability of overheating. As for the DRAM, the value is set at a lower level to maintain high reliability [35]. All the safe operating temperatures and other parameters are listed in Table 2 and Table 3, respectively. Note that the ambient temperature is set to 35°C when evaluating CoolEdge$^+$, as the Alibaba PAI trace is collected in summer [69].

TABLE 2: Safe operating temperature of hardware

| Hardware type | CPU | GPU | DRAM |
|---|---|---|---|
| Safe operating temperature (°C) | 77 | 76/62 | 43 |

TABLE 3: Other parameters

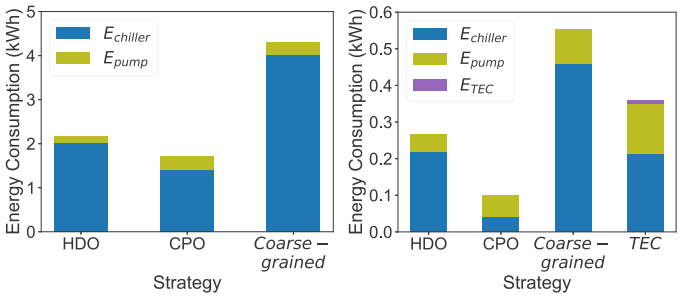
| Parameter | $h$ | $COP_c$ | Ambient temperature |
|---|---|---|---|
| Value | 10 W/m$^2$°C [73] | 3.6 [74] | 20°C/35°C |

### 6.3 Evaluation on CoolEdge with the Utilization Trace

We analyze the simulation results of CoolEdge (applying the HDO or CPO strategy) and two baseline strategies of *Coarse-grained* and *TEC* from several aspects as follows.

**Total cooling energy consumption:** As shown in Fig. 12a, the HDO and CPO strategies reduce the cooling energy by 49.57% and 60.08%, respectively, as compared to the *Coarse-grained* strategy. HDO and CPO strategies also reduce the partial power usage effectiveness (pPUE)[2] from 1.17 achieved by the *Coarse-grained* strategy to 1.09 and 1.07, respectively. It should be noted that although some recent cloud datacenters have achieved a low PUE thanks to the ideal climate and/or well-designed cooling techniques, the PUE of edge datacenters is typically close to 2 [75] owing to the critical challenges introduced in Sec. 2. For the results under the CPU trace, as plotted in Fig. 12b, the proposed strategies reduce up to 81.81% and 71.92% of the cooling energy, respectively, as compared with the *Coarse-grained* strategy and *TEC* strategy. The pPUE of the HDO, CPO, *TEC*, and *Coarse-grained* strategies are 1.05, 1.02, 1.10, and 1.06, respectively. In both figures, the CPO strategy consumes less cooling energy than the HDO strategy. On the one hand, the CPO strategy has a slightly higher $E_{pump}$ since it does not optimize $E_{pump}$. On the other hand, the CPO strategy reduces $E_{chiller}$ considerably as it aims at minimizing the amount of required chilled water. Note that in the following analysis of energy consumption patterns and hotspot elimination, we consider the heterogeneous system under all the traces and compare the proposed strategies with only the *Coarse-grained* strategy.

2. The pPUE is defined as (IT hardware energy + cooling energy) / IT hardware energy [13].




(a) Under all the traces (b) Under only the CPU trace

Fig. 12: Total cooling energy consumption.

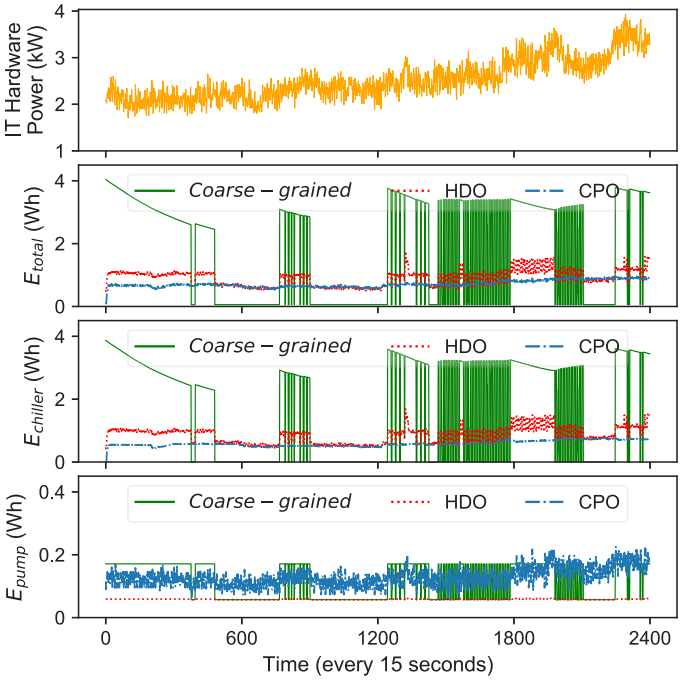



Fig. 13: Cooling energy consumption patterns.

**Cooling energy consumption patterns:** Fig. 13 depicts the total IT hardware power and the cooling energy consumption patterns of HDO, CPO, and *Coarse-grained* strategies. As we can see, both $E_{chiller}$ and $E_{pump}$ increase synchronously when the IT hardware power boosts, such as time = 1,800 and time = 2,200, because more chilled water has to be pumped to cool hardware components with higher power consumption and thus higher temperature. Compared with the HDC and CPO strategies, the *Coarse-grained* strategy incurs much more cooling energy fluctuation since it needs to consume substantial extra energy to eliminate even one hotspot in each period. The higher peak cooling demand usually means a higher capital expenditure of the chiller, which will be discussed in later cost saving analysis.

**Hotspot elimination for heterogeneous hardware:** The cumulative distribution functions (CDFs) of the maximum temperatures in each period of CPUs, GPUs, and DRAMs are plotted in Fig. 14 separately, and Fig. 15 plots their utilization patterns during the 10 hours. As we can see, the utilization of CPUs remains low almost all the time except for the No. 17 CPU, whose utilization exceeds 80% about half the time. By contrast, a large proportion of GPUs and DRAMs run close at their maximum utilization frequently and thus there are

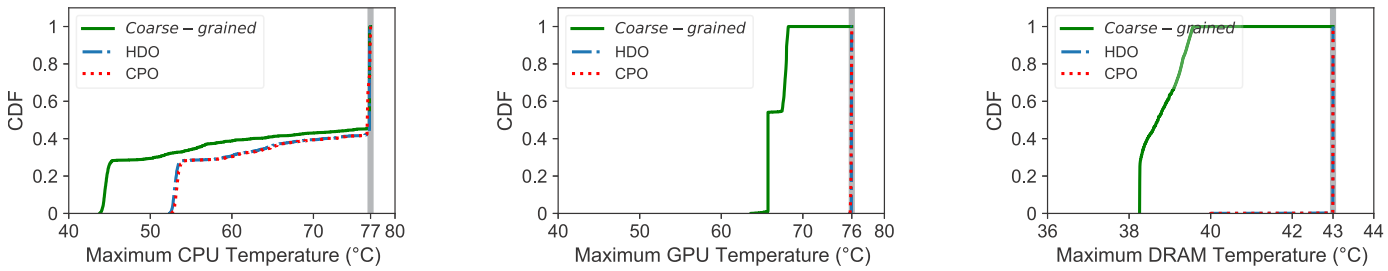



Fig. 14: CDFs of the maximum hardware temperature (the gray vertical lines represent safe operating temperatures).

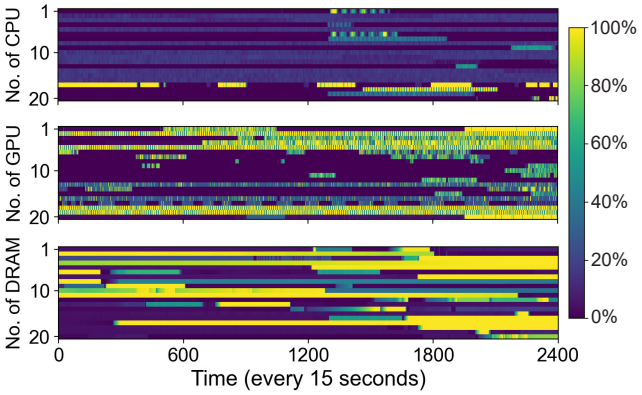



Fig. 15: Utilization patterns of CPUs, GPUs, and DRAMs.

TABLE 4: Cost saving calculation (unit: $/(server×year))

| Description | *TEC* | | *CoolEdge* | |
|---|---|---|---|---|
| ExCapEx | TEC | 0.4 | Valve | 18.0 |
| | Copper plate | 0.2 | Vapor chamber | 1.0 |
| | Additional cold plate | 2.1 | Copper baseplate | -0.2 |
| ChiSav | 1.25 | | 5.32 | |
| EnerSav | 2.68 | | 35.44 | |
| CoSav | 1.23 | | 21.96 | |

several hotspots in each period. That is why the maximum temperatures of GPUs and DRAMs remain high almost all the time. In conclusion, as plotted in Fig. 14, although the maximum temperatures get higher by applying the proposed strategies as compared with the *Coarse-grained* strategy, no hardware component is overheated.

**Cost saving analysis:** Here, we estimate the cost savings from CoolEdge and the *TEC* strategy as compared with the *Coarse-grained* strategy. We consider extra capital expenditures (ExCapEx), capital expenditure savings of the chiller (ChiSav), and cooling energy savings (EnerSav) in the analysis. ExCapEx mainly depends on the additions and can be calculated according to their purchase prices and lifespans [13], [48], [50], [55], [76], [77]. Note that the price of valves is irrelevant to hardware components, and the price of vapor chambers is mainly determined by the component size. Hence, these hardware prototype-based calculation results can reflect the actual ExCapEx for servers. Since the price of the chiller is mainly determined by its cooling capacity [16], ChiSav can be calculated by the peak cooling energy consumption as depicted in Fig. 13. As for EnerSav, it can be calculated from Fig. 12, where the electricity price for industrial consumers is about 15 cents/kWh [78]. Ultimately, Cost Savings (CoSav) can be calculated by ChiSav + EnerSav − ExCapEx. Based on the

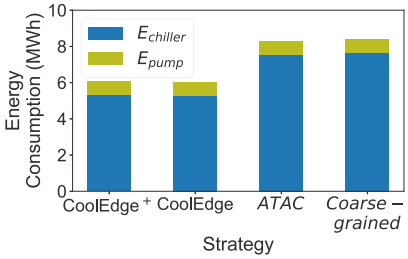



Fig. 16: Total cooling energy consumption.

experimental results shown in later Fig. 23 in Appendix A, as the CPU temperature is reduced by 4.36°C on average through integrating the vapor chambers, EnerSav and CoSav can be further improved by 4.5% and 3.5%, respectively [79]. All the calculation results are listed in Table 4. According to the estimation, for 2,000 small-scale edge datacenters (each equipped with 80 servers) in a city [80], [81], [82], the cost savings of the *TEC* strategy is about $196,800/year, while CoolEdge can save up to $3,513,600/year, 17.85× as much as the *TEC* strategy, showing the great potential to widely deploy CoolEdge. Besides, the *TEC* strategy is designed for homogeneous datacenters only, while CoolEdge can handle heterogeneity with good generalizability.

### 6.4 Evaluation on CoolEdge$^+$ with the Workload Trace

We analyze the simulation results of CoolEdge$^+$, CoolEdge, and two baseline strategies of *Coarse-grained* and *ATAC* from several aspects as follows.

**Energy and cost savings:** As shown in Fig. 16, CoolEdge$^+$ and CoolEdge reduce the cooling energy usage by 27.19% and 28.05%, respectively, as compared with the *Coarse-grained* strategy. The *ATAC* strategy lowers the cooling energy slightly by 1.30% than the *Coarse-grained* strategy at the expense of hardware performance. Similar to the discussion in Sec. 6.3, CoSav can be calculated by ChiSav + EnerSav − ExCapEx. Based on the purchase prices and lifespans of valves [48], [49], [77], the electricity price [78], and the demand on cooling capacity of the chiller [16], the calculation results are summarized in Table 5. As we can see, CoolEdge$^+$ further improves the cost savings by 35.24% than CoolEdge. For 2,000 small-scale edge datacenters (each equipped with 80 servers) in a city [80], [81], [82], the cost savings brought by CoolEdge$^+$ can reach $3,598,400/year.

**Computing performance:** Fig. 17 plots the inference latency increase of all tasks as compared to the inference latency without power capping, and Fig. 18 plots the CDF of the inference latency to its SLO constraint. Although CoolEdge$^+$ increases the inference latency by a ratio of

TABLE 5: Cost saving calculation (unit: $/(server×year))

| Description | *ATAC* | ***CoolEdge*** | | ***CoolEdge*$^+$** | |
|---|---|---|---|---|---|
| ExCapEx | 0 | Proportional valve | 12.00 | On/off valve | 5.79 |
| ChiSav | 0.68 | 17.33 | | 17.32 | |
| EnerSav | 0.53 | 11.30 | | 10.96 | |
| CoSav | 1.21 | 16.63 | | 22.49 | |

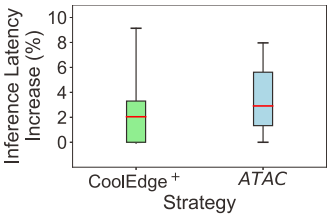



Fig. 17: The inference latency increase.

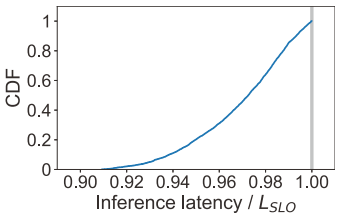


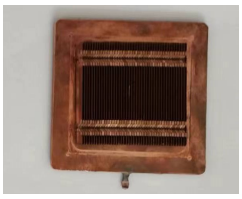

Fig. 18: The CDF of SLO satisfaction with CoolEdge$^+$.

1∼1.09 and by 1.02 on average, the latency is still within the SLO constraint. By comparison, the *ATAC* strategy increases the latency by 1.03 on average based on the cooling setup presented in Section 6.2. It is worth noting that CoolEdge$^+$ provides the ability to balance the computing performance and cooling energy usage by setting different SLOs, as illustrated by the bar filled with diagonal stripes in Fig. 10.

**Comparison with the results using the utilization trace:** As compared with the experimental results in Sec. 6.3, the energy and cost savings are lower here for the following reasons. First, we only consider one hardware type (i.e., GPU) as the workload trace only contains ML workloads that relies on GPUs, which results in smaller power consumption variation among components and a smaller number of hardware components per server considered in the cost saving calculation. Second, we do not consider the detailed power fluctuation while a task is running, since the workload trace does not include such information. Instead, we use the average power consumption value of an ML inference execution to determine the cooling demand. Both the above two reasons will diminish CoolEdge's and CoolEdge$^+$'s advantages in alleviating hotspots. Third, we do not consider adjusting the water flow rate for fair comparison between CoolEdge and CoolEdge$^+$ where the water flow rate of each branch should be equal when using on/off valves, which disables the component-level adjustment to the flow rate. Concerning the above reasons, we have revealed the lower bound of CoolEdge$^+$ in saving energy and costs.

**Comparison between CoolEdge and CoolEdge$^+$:** According to the aforementioned results, we can see that CoolEdge$^+$ achieves comparable energy savings as CoolEdge while reducing the CapEx of valves by over half, thus increasing the cost savings by near one million dollars every year for a city. However, the main concern of CoolEdge$^+$ is the degraded computing performance. Specifically, CoolEdge$^+$ is able to satisfy all SLO constraints on condition that the computing performance is allowed to degrade slightly. Otherwise, CoolEdge$^+$ will behave much worse than CoolEdge in avoiding over-cooling and improving cooling efficiency, as depicted in Fig. 10. Therefore, CoolEdge$^+$ is not suitable for edge datacenters with extreme performance requirements

(a) The appearance



(b) The fin structure

Fig. 19: Our newly developed cold plate.

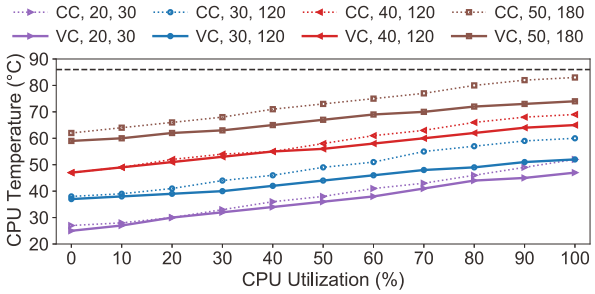



Fig. 20: CPU temperature under different cold plates, utilization, water temperature, and flow rate.

where the processing latency should be reduced as much as possible, and thus the power capping approach is not feasible. In conclusion, the selection between CoolEdge and CoolEdge$^+$ is highly dependent on the workloads supported by the edge datacenters.

### 6.5 Further Experiments on Advanced Vapor Chamber-Based Cold Plates

Attracted by the characteristics of vapor chambers introduced in Sec. 3.3, we further develop a customized, fully integrated vapor chamber-based cold plate with an internal fin structure, as shown in Fig. 19. We perform several experiments using the same setup as in Sec. 3.3 to compare the newly developed vapor chamber-based cold plate (VC) with the commonly used cold plate (CC), both of which include the internal fin structure. The results demonstrate the following three characteristics that are promising to edge datacenters.

**Reducing the overall hardware temperature:** Fig. 20 shows the overall CPU temperature, where the horizontal line indicates MOT (i.e., 86°C), and *CC, 20, 30* refers to using the commonly used cold plate under the inlet water temperature of 20°C and flow rate of 30 L/h. We can see that VC outperforms CC, especially when the CPU utilization and inlet water temperature get high. For example, when the CPU utilization and inlet water temperature are 100% and 50°C, respectively, the CPU temperature difference reaches 9°C. This characteristic helps narrow the temperature difference between hotspot components and others, especially for high-powered hardware components and in the scope of warm water cooling, saving the cooling energy for dispersing hotspots. As the TDP of modern server components continues to increase (e.g., 700 W of the Nvidia H100 GPU), we think that VC could play a key role in datacenter cooling.

**Smoothing the temperature distribution spatially:** Fig. 21 plots the core temperature distribution under various cooling conditions and utilization patterns. Across these eight settings, VC reduces the median and the maximum

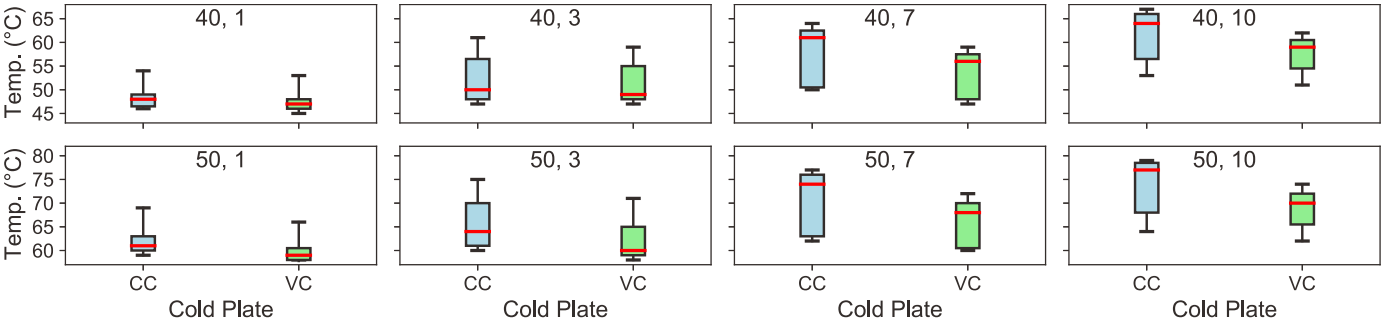
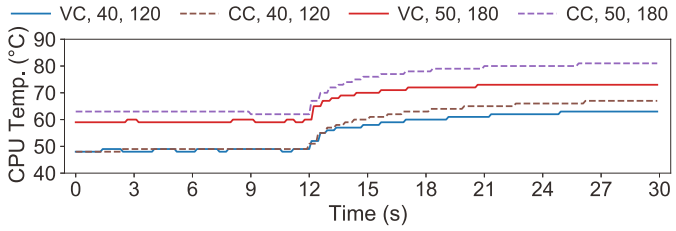
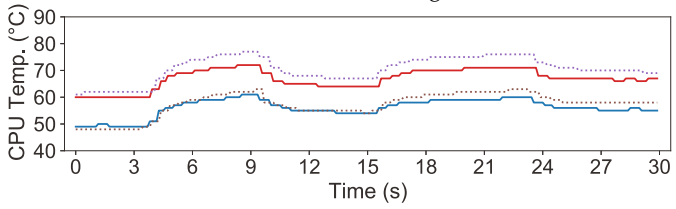



Fig. 21: The CPU core temperature distribution under various cooling conditions and utilization patterns. The first number in the title of each subfigure denotes the water temperature (°C), and the second number denotes the number of tested cores. The tested cores are kept at 100% utilization, and the rest remain 0%.




(a) At time = 11.6 s, the CPU utilization grows from 0% to 100%




(b) The CPU utilization grows from 0% to 100% at time = 3.6 s, to 20% at time = 9.3 s, to 80% at time = 15.0 s, and finally, to 40% at time = 23.2 s

Fig. 22: CPU temperature variation as utilization varies.

core temperature by 1°C∼7°C and 1°C∼5°C, respectively, as compared with CC. Also, the standard deviation drops from 2.38°C∼5.99°C to 2.08°C∼4.73°C after using VC. This characteristic brings two benefits. (1) Hardware safety: VC helps reduce the probability of local overheating inside a component automatically, especially when the component is partially utilized. This improves hardware safety and lifespans since there cannot be thermosensors everywhere inside the component to monitor local temperatures. (2) Cooling energy usage: VC helps reduce the maximum core temperature that usually determines the cooling demand. The cooling energy for dealing with hotspots can be reduced, especially in the existing coarse-grained cooling system.

**Smoothing the temperature variation temporally:** Fig. 22 plots the CPU temperature variation as the utilization varies. As we can see, the CPU temperature varies more smoothly when using VC rather than CC. For example, as shown in Fig. 22a, when the water temperature is 50°C, it takes 1.0 s and 2.9 s for the CPU temperature to reach 70°C with CC and VC, respectively. This characteristic helps slow down the instantaneous hardware temperature rise in face of the cooling lag and thus improves hardware safety, especially for high-powered hardware components running edge workloads with high utilization variation [9].

## 7 Related Work

**Warm water cooling:** Warm water cooling has been utilized by many works to reduce cooling costs [13], [14], [83], [84], [85], [86], [87]. Jiang et al. [13] propose a fine-grained warm water cooling solution that eliminates hotspots with TECs. However, as discussed in Sec. 2.3, there exist several limitations when dealing with high density and heterogeneity of edge datacenters. By contrast, CoolEdge provides general cooling supports for heterogeneous hardware components. Considering the limited space inside high-density servers, our newly-designed cold plates can directly replace original ones and all valves can be easily installed outside servers. Zhu et al. [14] illustrate another benefit of warm water cooling that the waste heat can be not only used for district heating, but also turned into electricity with thermoelectric generators. To the best of our knowledge, we are the first to theoretically analyze and formulate the profit of warm water cooling, and adopt it into edge datacenters with strict requirements and critical challenges.

**Immersion cooling:** Nowadays, immersion cooling has emerged as an efficient water cooling technique to hold a higher power density [88], [89]. Sugon has exhibited its newly designed two-phase immersion cooling server at Supercomputing 2018 [90]. As compared with direct-to-chip cooling, immersion cooling can reduce PUE further, and eliminate the use of cold plates and the corresponding engineering costs to design a new cold plate when the physical structure of new hardware components changes. However, there are still some technical problems of immersion cooling that may prevent its adoption in edge datacenters. Firstly, due to the strict requirements of fluid, such as high thermal conductivity, low electrical conductivity, fixed boiling point, etc., the fluid cost is much higher than direct-to-chip cooling [91]. Secondly, highly sealed sleds and stable gas pressure are necessary to ensure safety, which increases the tank cost. Thirdly, immersion cooling usually occupies more space since racks are usually placed horizontally rather than vertically [92], worsening the problem of land scarcity at the edge. Last but not least, immersion cooling may be over qualified in most cases at present [92], where the cheaper direct-to-chip cooling probably fits better.

**Vapor chamber solutions:** Several studies propose to use vapor chambers to cool heat sources like processors. Tsai et al. [93] and Liu et al. [94] focus on the structures (e.g., flow mechanism) and properties (e.g., thermal resistance) of vapor

chambers, and the influencing factors in heat conduction like installation orientations. Parhizi et al. [95] and Yuan et al. [96] evaluate the performance of vapor chambers by developing simulation models. Different from these studies, our work digs into thermal specifications of real hardware components used in datacenters. To our best knowledge, we are the first to integrate the vapor chamber into the cold plate which dissipates heat from server hardware components in a practical and cost-effective manner.

**Power and thermal management in datacenters:** Due to the rapid growth of cloud and edge computing, Internet of things, deep learning, etc., the power consumption of IT and cooling equipment in datacenters increases dramatically in recent years [97], [98]. To improve the overall efficiency, many works pay close attention to power and thermal management in datacenters, such as workload management [99], [100], [101], [102], [103], [104], [105], [106], [107], hotspot elimination [13], [35], [71], underprovisioning [28], [108], heat harvesting [14], [109], [110], and demand response [111], [112]. In this section, we summarize some closely related works on the hotspot issue and/or cooling efficiency. Liu et al. [35] notice that the DRAM temperature can rise to alarming 95°C in a high-performance computing IT system. To avoid throttling and maintain high performance, they propose three schemes to reduce peak temperature and temperature variation in an air-cooled datacenter. Zhou et al. [101] develop a power management framework to save CPU power with the dynamic voltage and frequency scaling technique. Instead of a single hardware type, other works focus on the whole datacenter infrastructure including the cooling system. Intel [103] devises a scheme of improving the ambient temperature in a low-power-density air-cooled datacenter to increase cooling efficiency. Ran et al. [100] propose a deep reinforcement learning based framework to schedule CPU jobs and adjust airflow for saving cooling energy while reducing hotspots. However, these works mainly focus on either power and thermal management of IT hardware components, or system-level cooling control in a homogeneous cloud datacenter, while our work enables component-level cooling control tailored to high-density and heterogeneous edge datacenters.

This work significantly extends the preliminary work [113]. First, to reduce the cooling complexity of CoolEdge, we further propose a semi-fine-grained cooling solution named CoolEdge$^+$. It leverages simpler but less flexible valves to achieve the component-level cooling control. Although only certain inlet water temperature values are accessible in this case, CoolEdge$^+$ could still reduce overcooling significantly by employing an SLO-aware power capping approach. Then, we further develop a customized, fully integrated vapor chamber-based cold plate to improve the effectiveness of dispersing heat and smoothing the temperature distribution. Finally, we add several experiments to evaluate the effectiveness of the extensions. In particular, we evaluate CoolEdge$^+$ with a new real-world trace and compare it with CoolEdge and an additional baseline.

## 8 Concluding Remarks

In this paper, we provide two water cooling solutions CoolEdge and CoolEdge$^+$ for owner-operated edge datacenters. For chip-level hotspots, both of them integrate the vapor chamber into the commonly used cold plate; for hardware-level hotspots, CoolEdge achieves fine-grained cooling control through the customized mix of hot and chilled water, while CoolEdge$^+$ reduces it to semi-fine-grained cooling control with low capital expenditures but incurs minor performance penalty. The evaluation results indicate that CoolEdge saves the cooling energy by 81.81% and 71.92%, respectively, compared with the conventional and state-of-the-art water cooling systems, and CoolEdge$^+$ achieves comparable cooling efficiency improvement as CoolEdge but saves 35.24% more costs. For a city with 2,000 edge datacenters, CoolEdge$^+$ can deliver $3,598,400 of cost savings yearly based on our estimation. It is worth noting that these cooling solutions are also applicable to cloud datacenters although their requirements are less stringent.

## Appendix A
## Measurement Results on the CPU Temperature with Different Cold Plates

To verify the effectiveness of our approach to the integration of vapor chambers, we conduct some experiments to reveal the impact on CPU temperature of the commonly used cold plate (CC) and two cold plates leveraging aforementioned integration approaches discussed in Sec. 3.3. One approach is attaching the vapor chamber to the commonly used cold plate directly (VC-CC) and the other is replacing its baseplate with the vapor chamber (VC, our approach). The results are plotted in Fig. 23[3], where the horizontal line indicates MOT (i.e., 86°C), and *CC, 30, 120* refers to using the commonly used cold plate under the inlet water temperature of 30°C and flow rate of 120 L/h. We can see that VC outperforms the others, especially when the CPU utilization and inlet water temperature get high. For selecting a cost-effective vapor chamber for the integration, we also test vapor chambers with different materials (copper and aluminum) and thicknesses under various load and cooling conditions (e.g., hardware utilization, water temperature, flow rate, etc.).

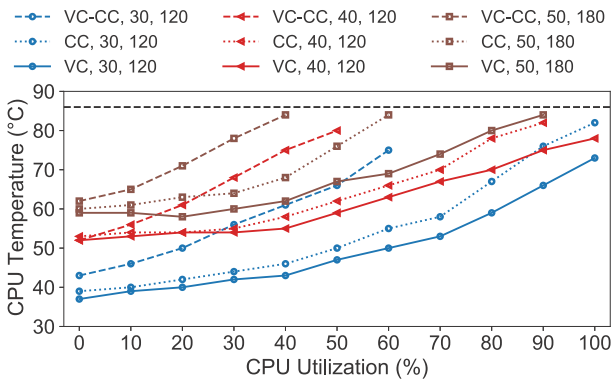



Fig. 23: CPU temperature under different cold plates, utilization, water temperature, and flow rate.

3. The details of the CPU are presented in Section 6.1.

## APPENDIX B
## DETAILED DISCUSSION ON THE NATURAL HEAT DISSIPATION

To derive Proposition 1, we consider a hollow-cylinder pipe of length $L$ and thermal conductivity $\lambda$. $r_i$, $r_o$, $T_i$, and $T_o$ denote the inner radius, outer radius, inner surface temperature, and outer surface temperature, respectively. For steady-state heat conduction, the problem can be formulated as $\frac{L}{r}\frac{d}{dr}(r\frac{dT}{dr}) = 0$ [114]. By applying the boundary conditions $T|_{r=r_i} = T_i$ and $T|_{r=r_o} = T_o$, temperature distribution along the radial direction is given by $T_r = T_i - \frac{T_i - T_o}{\ln(\frac{r_o}{r_i})}\ln(\frac{r}{r_i})$. Based on Fourier's law of heat conduction [59] and Newton's law of cooling [60], the dissipated heat $P$ (in Watts) through the pipe is expressed by:

$$P = (T_i - T_o)\Big/\left(\frac{\ln(\frac{r_o}{r_i})}{2\pi\lambda L} + \frac{1}{2\pi h r_o L}\right). \tag{5}$$

Eq. (5) assumes that the water temperature along the axis remains unchanged. To obtain a more accurate solution, we further adopt the infinitesimal calculus approach. Specifically, one water element with length $dl$ is considered, whose initial temperature is $T_i$ (i.e., the inlet water temperature of the pipe). Based on Eq. (5), the dissipated heat through the pipe in a time slot $dt$ can be obtained by $Q_{diss} = Pdt = (T_i - T_o)/(\frac{\ln\frac{r_o}{r_i}}{2\pi\lambda dl} + \frac{1}{2\pi h r_o dl})dt$. According to the law of conservation of energy [65], $Q_{diss}$ equals to the heat loss of the water $Q_{loss} = -c\pi {r_i}^2\rho dl \cdot dT$, where $\rho$ and $c$ are the density and specific heat capacity of the water, respectively, and $dT$ denotes the temperature reduction within $dt$. Hence, we have:

$$\begin{aligned}&(T - T_o)\Big/\left(\frac{\ln\frac{r_o}{r_i}}{2\pi\lambda dl} + \frac{1}{2\pi h r_o dl}\right)dt = -c\pi {r_i}^2\rho dl \cdot dT\\ \Leftrightarrow\ &\frac{dT}{dt} + \frac{1}{cr_i^2\rho(\frac{\ln(\frac{r_o}{r_i})}{2\lambda} + \frac{1}{2hr_o})}T = \frac{1}{cr_i^2\rho(\frac{\ln(\frac{r_o}{r_i})}{2\lambda} + \frac{1}{2hr_o})}T_o.\end{aligned}$$

By defining $\alpha = \pi/(\frac{\ln(\frac{r_o}{r_i})}{2\lambda} + \frac{1}{2hr_o})$, we have $T_{v,l} = (T_i - T_o)e^{-\frac{\alpha}{c\rho v}l} + T_o$, where $T_{v,l}$ refers to the water temperature at a distance of $l$ from the inlet when the flow rate is $v$. Hence, we can obtain the dissipated heat through the pipe with Eq. (5):

$$P = \int_l dP = c\rho v(T_i - T_o)(1 - \exp(-\frac{\alpha}{c\rho v}L)). \tag{6}$$

To verify the accuracy of Eq. (6), we use Fluent software [63] to simulate the process of heat dissipation through pipes. We choose a 1-meter copper pipe whose inner radius, outer radius, and thermal conductivity are $4\,\mathrm{mm}$, $5\,\mathrm{mm}$, and 401 W/m°C, respectively. The ambient temperature is assumed to be stable at 20°C. From the results in Fig. 6, we find that the estimation of the dissipated heat is generally accurate, while there is still an estimation error of at most 8.4% when $T_i$ and $h$ are high. The main reason is that we do not consider the water temperature distribution along the radius direction though it is generally uniform. Here, we use an attenuation factor $\beta = (1 - 0.0008h)\exp(-\frac{1}{26.5\sigma + 5.2})$ to represent this effect, where $h \leq 100$ W/m$^2$°C and the water velocity $\sigma \leq 1$ m/s.

## REFERENCES


[1] Gartner, "What edge computing means for infrastructure and operations leaders," 2018, https://www.gartner.com/smarterwithgartner/what-edge-computing-means-for-infrastructure-and-operations-leaders/.

[2] LF Edge, "State of the edge 2020," https://www.lfedge.org/wp-content/uploads/2020/04/SOTE2020.pdf.

[3] ——, "State of the edge 2021," https://www.lfedge.org/2021/03/12/state-of-the-edge-2021-report/.

[4] G. Kamiya, "Data centres and data transmission networks," 2021, https://www.iea.org/reports/data-centres-and-data-transmission-networks.

[5] Open Compute Project, "Open Rack/SpecsAndDesigns," https://www.opencompute.org/wiki/Open_Rack/SpecsAndDesigns.

[6] Vapor IO, "Vapor chamber technical specifications," https://www.vapor.io/wp-content/uploads/2018/05/Vapor-Chamber-Tech-Specs-V3.pdf.

[7] cnTechPost, "Tencent Cloud enables first 5G edge computing center," 2020, https://cntechpost.com/2020/10/15/tencent-cloud-enables-first-5g-edge-computing-center/.

[8] C. Byers, "Heterogeneous computing in the edge," 2021, https://www.iiconsortium.org/news/joi-articles/2021_June_JoI_Edge_Hetero_Compute_Byers_Final.pdf.

[9] M. Xu, Z. Fu, X. Ma, L. Zhang, Y. Li, F. Qian, S. Wang, K. Li, J. Yang, and X. Liu, "From cloud to edge: a first look at public edge platforms," in *Proceedings of the 21st ACM Internet Measurement Conference*, 2021.

[10] Asetek, "Liquid cooling for data centers," 2021, https://www.asetek.com/data-center/technology-for-data-centers.

[11] R. Ayoub, R. Nath, and T. Rosing, "JETC: Joint energy thermal and cooling management for memory and CPU subsystems in servers," in *Proceedings of the 18th International Symposium on High Performance Computer Architecture*, 2012.

[12] Vertiv, "Liquid cooling options for data centers," https://www.vertiv.com/en-us/solutions/learn-about/liquid-cooling-options-for-data-centers/.

[13] W. Jiang, Z. Jia, S. Feng, F. Liu, and H. Jin, "Fine-grained warm water cooling for improving datacenter economy," in *Proceedings of the 46th Annual International Symposium on Computer Architecture*, 2019.

[14] X. Zhu, W. Jiang, F. Liu, Q. Zhang, L. Pan, Q. Chen, and Z. Jia, "Heat to power: Thermal energy harvesting and recycling for warm water-cooled datacenters," in *Proceedings of the 47th Annual International Symposium on Computer Architecture*, 2020.

[15] M. Frizziero, "Rethinking chilled water temperatures can bring big savings in data center cooling," 2016, https://blog.se.com/datacenter/2016/08/17/water-temperatures-data-center-cooling/.

[16] FPL, "Air-cooled chillers," https://infpl.fpl.com/business/pdf/air-cooled-chillers-primer.pdf.

[17] J. Wan, X. Gui, S. Kasahara, Y. Zhang, and R. Zhang, "Air flow measurement and management for improving cooling and energy efficiency in raised-floor data centers: A survey," *IEEE Access*, vol. 6, pp. 48 867–48 901, 2018.

[18] CoolIT Systems, "Why centralized pumping outperforms distributed systems for rack-based direct liquid cooling," 2015, https://www.hpcwire.com/2015/10/26/why-centralized-pumping-outperforms-distributed-systems-for-rack-based-direct-liquid-cooling/.

[19] CPU World, "Minimum/maximum operating temperatures," 2018, https://www.cpu-world.com/Glossary/M/Minimum_Maximum_operating_temperatures.html.

[20] L. Ramos and R. Bianchini, "C-Oracle: Predictive thermal management for data centers," in *Proceedings of the 14th International Symposium on High Performance Computer Architecture*, 2008.

[21] R. Prashnani, "Task migration algorithm to reduce temperature imbalance amongst cores in linux based multi-core processor systems," in *Proceedings of the 3rd International Conference for Convergence in Technology*, 2018.

[22] Intel, "Intel Xeon Processor E5-2680 v4," https://ark.intel.com/content/www/us/en/ark/products/91754/intel-xeon-processor-e5-2680-v4-35m-cache-2-40-ghz.html.

[23] Safe Temp, "Nvidia GeForce RTX 2080 Ti max temp," 2019, https://safetemp.blogspot.com/2019/01/geforce-rtx-2080-ti-max-temp.html.

[24] Tech Power Up, "Nvidia GeForce RTX 2080 Ti," https://www.techpowerup.com/gpu-specs/geforce-rtx-2080-ti.c3305.

[25] Safe Temp, "Nvidia A100 PCIe 80GB max temp," 2021, https://safetemp.blogspot.com/2021/10/a100-pcie-80-gb-max-temp.html.

[26] NVIDIA, "NVIDIA A100 Tensor Core GPU," https://www.nvidia.com/en-us/data-center/a100/.

[27] Samsung, "983 DCT," https://www.samsung.com/semiconductor/minisite/ssd/product/data-center/983dct/.

[28] I. Manousakis, Í. Goiri, S. Sankar, T. D. Nguyen, and R. Bianchini, "CoolProvision: Underprovisioning datacenter cooling," in *Proceedings of the 6th ACM Symposium on Cloud Computing*, 2015.

[29] C. Fingar, "Playing it cool pays off for iceland," https://www.fdiintelligence.com/article/57142.

[30] J. Roach, "Microsoft finds underwater datacenters are reliable, practical and use energy sustainably," 2020, https://news.microsoft.com/innovation-stories/project-natick-underwater-datacenter/.

[31] Rittal, "Data center cooling: 4 effective types of liquid cooling," 2020, https://zutacore.com/wp-content/uploads/2020/06/Data_Center_Cooling_Liquid_Cooling_US528.pdf.

[32] Vapor IO, "The vapor chamber," https://www.vapor.io/chamber/.

[33] J. Musilli and P. Vaccaro, "Intel IT redefines the high-density data center: 1,100 Watts/Sq Ft," 2014, https://www.intel.com/content/dam/www/public/us/en/documents/white-papers/intel-it-redefines-high-density-data-center-1100-watts-per-sqft-paper.pdf.

[34] Schneider Electric, "Liquid cooling technologies for data centers and edge applications," https://download.schneider-electric.com/files?p_Doc_Ref=SPD_VAVR-AQKM3N_EN.

[35] S. Liu, B. Leung, A. Neckar, S. O. Memik, G. Memik, and N. Hardavellas, "Hardware/software techniques for DRAM thermal management," in *Proceedings of the 17th International Symposium on High Performance Computer Architecture*, 2011.

[36] Wikipedia, "Thermal design power," https://en.wikipedia.org/wiki/Thermal_design_power.

[37] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "HotSpot: A compact thermal modeling methodology for early-stage VLSI design," *IEEE Transactions on very large scale integration systems*, vol. 14, no. 5, pp. 501–513, 2006.

[38] Q. Zhu, X. Li, and Y. Wu, "Thermal managerment of high power memory module for server platforms," in *Proceedings of the 11th Intersociety Conference on Thermal and Thermomechanical Phenomena in misc Systems*, 2008.

[39] P. Zou, L. Ang, K. Barker, and R. Ge, "Indicator-directed dynamic power management for iterative workloads on GPU-accelerated systems," in *Proceedings of the 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing*, 2020.

[40] S. K. Khatamifard, L. Wang, W. Yu, S. Köse, and U. R. Karpuzcu, "ThermoGater: Thermally-aware on-chip voltage regulation," in *Proceedings of the 44th Annual International Symposium on Computer Architecture*, 2017.

[41] J. Lin, H. Zheng, Z. Zhu, H. David, and Z. Zhang, "Thermal modeling and management of DRAM memory systems," in *Proceedings of the 34th Annual International Symposium on Computer Architecture*, 2007.

[42] M. A. Adnan, R. Sugihara, and R. K. Gupta, "Energy efficient geographical load balancing via dynamic deferral of workload," in *Proceedings of the 5th IEEE International Conference on Cloud Computing*, 2012.

[43] T. Heath, A. P. Centeno, P. George, L. Ramos, Y. Jaluria, and R. Bianchini, "Mercury and Freon: Temperature emulation and management for server systems," in *Proceedings of the 12th International Conference on Architectural Support for Programming Languages and Operating Systems*, 2006.

[44] I. n. Goiri, T. D. Nguyen, and R. Bianchini, "CoolAir: Temperature- and variation-aware management for freecooled datacenters," in *Proceedings of the 20th International Conference on Architectural Support for Programming Languages and Operating Systems*, 2015.

[45] S. Shurpali, "Role of edge computing in connected and autonomous vehicles," https://www.einfochips.com/blog/role-of-edge-computing-in-connected-and-autonomous-vehicles/.

[46] "Catalogue of our peltier modules," https://peltiermodules.com/?p=product.

[47] Tameson, "Proportional solenoid valve – how they work," https://tameson.com/proportional-solenoid-control-valve.html.

[48] Alibaba, "Price of the proportional solenoid valve," 2021, https://www.alibaba.com/product-detail/proportional-solenoid-valve_60730339977.html.

[49] ——, "Price of the 2-way solenoid valve," 2023, https://www.alibaba.com/product-detail/2-2-Way-NC-2W160-15_1600473913429.html.

[50] Radian, "What is a vapor chamber heatsink?" https://www.radianheatsinks.com/vapor-chamber-heatsink/.

[51] T. Tharayil, L. G. Asirvatham, V. Ravindran, and S. Wongwises, "Effect of filling ratio on the performance of a novel miniature loop heat pipe having different diameter transport lines," *Applied Thermal Engineering*, vol. 106, pp. 588–600, 2016.

[52] S. Wiriyasart and P. Naphon, "Fill ratio effects on vapor chamber thermal resistance with different configuration structures," *International Journal of Heat and Mass Transfer*, vol. 127, pp. 164–171, 2018.

[53] R.-T. Wang, J.-C. Wang, and T.-L. Chang, "Experimental analysis for thermal performance of a vapor chamber applied to high-performance servers," *Journal of Marine Science and Technology*, vol. 19, no. 4, pp. 353–360, 2011.

[54] Y.-S. Chen, K.-H. Chien, T.-C. Hung, C.-C. Wang, Y.-M. Ferng, and B.-S. Pei, "Numerical simulation of a heat sink embedded with a vapor chamber and calculation of effective thermal conductivity of a vapor chamber," *Applied Thermal Engineering*, vol. 29, no. 13, pp. 2655–2664, 2009.

[55] Alibaba, "Price of the vapor chamber," 2021, https://www.alibaba.com/product-detail/China-custom-copper-cu1100-VC-vapor_62410320227.html.

[56] TLX Technologies, "PWM solenoid theory," https://www.tlxtech.com/understanding-solenoids/theory-operation/pwm-solenoid-theory.

[57] Tameson, "Solenoid valve response time," https://tameson.com/solenoid-valve-response-time.html.

[58] Parker, "Miniature proportional valve," https://ph.parker.com/us/12051/en/hf-pro-miniature-proportional-valve/920-000047-001.

[59] N. Connor, "What is Fourier's law of thermal conduction — definition," https://www.thermal-engineering.org/what-is-fouriers-law-of-thermal-conduction-definition/.

[60] M. Yoshizawa, "Newton's law of cooling or heating," http://web.math.ucsb.edu/~myoshi/cooling.pdf.

[61] Engineers Edge, "Convective heat transfer convection equation and calculator," https://www.engineersedge.com/heat_transfer/convection.htm.

[62] P. Kosky, R. Balmer, W. Keat, and G. Wise, "Chapter 14 - mechanical engineering," in *Exploring Engineering (Fifth Edition)*, fifth edition ed., P. Kosky, R. Balmer, W. Keat, and G. Wise, Eds. Academic Press, 2021, pp. 317 – 340.

[63] Ansys, "Ansys Fluent fluid simulation software," 2021, https://www.ansys.com/products/fluids/ansys-fluent.

[64] S. Mishra, H. Chandra, and A. Arora, "Effects on heat transfer and radial temperature profile of non-isoviscous vibrational flow with varying Reynolds number," *Journal of Applied Fluid Mechanics*, vol. 12, no. 1, pp. 135–144, 2019.

[65] E. Education, "Law of conservation of energy," https://energyeducation.ca/encyclopedia/Law_of_conservation_of_energy.

[66] Wikipedia, "Coefficient of performance," https://en.wikipedia.org/wiki/Coefficient_of_performance.

[67] Dell, "Dell precision tower 7000 series (7910)," https://i.dell.com/sites/doccontent/shared-content/data-sheets/en/Documents/Dell-Precision-Tower-7000-Series-7910-Spec-Sheet.pdf.

[68] K. V. Laursen Olason, A. Uta, A. Iosup, P. Melis, D. Podareanu, and V. Codreanu, "Beneath the SURFace: An MRI-like view into the life of a 21st century datacenter," 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3878143

[69] Q. Weng, W. Xiao, Y. Yu, W. Wang, C. Wang, J. He, Y. Li, L. Zhang, W. Lin, and Y. Ding, "MLaaS in the wild: Workload analysis and scheduling in large-scale heterogeneous GPU clusters," in *Proceedings of the 19th USENIX Symposium on Networked Systems Design and Implementation*, 2022.

[70] N-able, "Servers vs. workstations overview," 2019, https://www.n-able.com/blog/servers-vs-workstations.

[71] S. Yeo, M. M. Hossain, J.-C. Huang, and H.-H. S. Lee, "ATAC: Ambient temperature-aware capping for power efficient datacenters," in *Proceedings of the 5th ACM Symposium on Cloud Computing*, 2014, pp. 1–14.

[72] Brian Stone, "Why you should prolong GPU lifespan (and ways to do it right)," https://www.computer.org/publications/tech-news/trends/why-you-should-prolong-gpu-lifespan.

[73] H. Khurshid, K. Silaipillayarputhur, and T. Al Mughanam, "Design of a heat sink for an misc component in ABB drive using different types of fins," in *MATEC Web of Conferences*, vol. 249, 2018, p. 03009.

[74] McQuay, "Air-cooled screw chiller," http://www.mcquay.com.hk/my-product/air-cooled-screw-chiller-awsmcs/.

[75] V. A. Vincent Petit, Steven Carlini, "Digital economy and climate impact," https://download.schneider-electric.com/files?p_File_Name=998-21202519.pdf.

[76] Alibaba, "Price of the cold plate," 2021, https://www.alibaba.com/product-detail/Standard-40-40-mm-copper-liquid_62462733833.html.

[77] Thorne & Derrick, "ASCO 220 solenoid valves," https://www.heatingandprocess.com/product/product-category/asco-220-solenoid-valves-steam/.

[78] Statista, "Electricity prices for industries in the European Union in 2019, by country," 2020, https://www.statista.com/statistics/1046605/industry-electricity-prices-european-union-country/.

[79] E. E. Manual, "Keep the chilled water supply temperature as high as possible," 2015, http://energybooks.com/wp-content/uploads/2015/07/264266.pdf.

[80] B. Waldhauser, "The edge & edge data centers: Gaining clarity," 2019, https://www.dotmagazine.online/issues/on-the-edge-building-the-foundations-for-the-future/edge-data-centers.

[81] VIAVI Solutions, "What is 5G technology?" https://www.viavisolutions.com/en-us/5g-technology.

[82] City Mayors, "The largest cities in the world by land area, population and density," 2007, http://www.citymayors.com/statistics/largest-cities-area-125.html.

[83] M.-H. Kim, S.-W. Ham, J.-S. Park, and J.-W. Jeong, "Impact of integrated hot water cooling and desiccant-assisted evaporative cooling systems on energy savings in a data center," *Energy*, vol. 78, pp. 384–396, 2014.

[84] 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 *Proceedings of the 28th Annual IEEE Semiconductor Thermal Measurement and Management Symposium*, 2012.

[85] M. Sahini, C. Kshirsagar, M. Kumar, D. Agonafer, J. Fernandes, J. Na, V. Mulay, P. McGinn, and M. Soares, "Rack-level study of hybrid cooled servers using warm water cooling for distributed vs. centralized pumping systems," in *Proceedings of the 33rd Thermal Measurement, Modeling & Management Symposium*, 2017.

[86] A. Addagatla, J. Fernandes, D. Mani, D. Agonafer, and V. Mulay, "Effect of warm water cooling for an isolated hybrid liquid cooled server," in *Proceedings of the 31st Thermal Measurement, Modeling & Management Symposium*, 2015.

[87] R. Januszewski, N. Meyer, and J. Nowicka, "Evaluation of the impact of direct warm-water cooling of the HPC servers on the data center ecosystem," in *Proceedings of the International Supercomputing Conference*, 2014.

[88] M. Jalili, I. Manousakis, Í. Goiri, P. A. Misra, A. Raniwala, H. Alissa, B. Ramakrishnan, P. Tuma, C. Belady, M. Fontoura, and R. Bianchini, "Cost-efficient overclocking in immersion-cooled datacenters," in *Proceedings of the 48th Annual International Symposium on Computer Architecture*, 2021.

[89] 3M, "Immersion cooling for data centers," 2021, https://www.3m.com/3M/en_US/data-center-us/applications/immersion-cooling/.

[90] P. Kennedy, "Sugon nebula phase change immersion cooling a unique platform," 2018, https://www.servethehome.com/sugon-nebula-phase-change-immersion-cooling-a-unique-platform/.

[91] Pol, "What is immersion cooling," https://submer.com/blog/what-is-immersion-cooling/.

[92] H. Villa, "Liquid cooling vs. immersion cooling deployment," 2020, https://blog.rittal.us/liquid-cooling-vs-immersion-cooling-deployment.

[93] M.-C. Tsai, S.-W. Kang, and K. V. de Paiva, "Experimental studies of thermal resistance in a vapor chamber heat spreader," *Applied Thermal Engineering*, vol. 56, no. 1-2, pp. 38–44, 2013.

[94] W. Liu, J. Gou, Y. Luo, and M. Zhang, "The experimental investigation of a vapor chamber with compound columns under the influence of gravity," *Applied Thermal Engineering*, vol. 140, pp. 131–138, 2018.

[95] M. Parhizi, A. A. Merrikh, and A. Jain, "Investigation of two-phase, vapor chamber based thermal management of multiple microserver chips," in *Proceedings of the International Mechanical Engineering Congress and Exposition*, 2014.

[96] Z. Yuan, G. Vaartstra, P. Shukla, S. Reda, E. Wang, and A. K. Coskun, "Modeling and optimization of chip cooling with two-phase vapor chambers," in *Proceedings of the IEEE/ACM International Symposium on Low Power miscs and Design*, 2019.

[97] L. Ismail and H. Materwala, "Computing server power modeling in a data center: Survey, taxonomy, and performance evaluation," *ACM Computing Surveys*, vol. 53, no. 3, 2020.

[98] W. Deng, F. Liu, H. Jin, B. Li, and D. Li, "Harnessing renewable energy in cloud datacenters: opportunities and challenges," *IEEE Network*, vol. 28, no. 1, pp. 48–55, 2014.

[99] 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 *Proceedings of the 12th ACM SIGMETRICS/PERFORMANCE joint international conference on Measurement and Modeling of Computer Systems*, 2012.

[100] Y. Ran, H. Hu, X. Zhou, and Y. Wen, "DeepEE: Joint optimization of job scheduling and cooling control for data center energy efficiency using deep reinforcement learning," in *Proceedings of the 39th International Conference on Distributed Computing Systems*, 2019.

[101] L. Zhou, L. N. Bhuyan, and K. Ramakrishnan, "Gemini: Learning to manage CPU power for latency-critical search engines," in *Proceedings of the 53rd Annual International Symposium on Microarchitecture*, 2020.

[102] K. Ji, C. Chi, A. Marahatta, F. Zhang, and Z. Liu, "Energy efficient scheduling based on marginal cost and task grouping in data centers," in *Proceedings of the 11th ACM International Conference on Future Energy Systems*, 2020.

[103] Intel, "The efficient datacenter," https://www.intel.co.jp/content/dam/doc/technology-brief/efficient-datacenter-high-ambient-temperature-operation-brief.pdf.

[104] S. Lee, K.-D. Kang, H. Lee, H. Park, Y. Son, N. S. Kim, and D. Kim, "GreenDIMM: OS-assisted DRAM power management for DRAM with a sub-array granularity power-down state," in *Proceedings of the 54th Annual International Symposium on Microarchitecture*, 2021.

[105] F. Liu, Z. Zhou, H. Jin, B. Li, B. Li, and H. Jiang, "On arbitrating the power-performance tradeoff in SaaS clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 10, pp. 2648–2658, 2014.

[106] Z. Zhou, F. Liu, B. Li, B. Li, H. Jin, R. Zou, and Z. Liu, "Fuel cell generation in geo-distributed cloud services: A quantitative study," in *Proceedings of the 34th International Conference on Distributed Computing Systems*, 2014.

[107] Z. Zhou, F. Liu, R. Zou, J. Liu, H. Xu, and H. Jin, "Carbon-aware online control of geo-distributed cloud services," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2506–2519, 2016.

[108] C. Zhang, A. G. Kumbhare, I. Manousakis, D. Zhang, P. A. Misra, R. Assis, K. Woolcock, N. Mahalingam, B. Warrier, D. Gauthier, L. Kunnath, S. Solomon, O. Morales, M. Fontoura, and R. Bianchini, "Flex: High-availability datacenters with zero reserved power," in *Proceedings of the 48th Annual International Symposium on Computer Architecture*, 2021.

[109] J. Liu, M. Goraczko, S. James, C. Belady, J. Lu, and K. Whitehouse, "The data furnace: Heating up with cloud computing," in *Proceedings of the 3rd USENIX Workshop on Hot Topics in Cloud Computing*, 2011.

[110] S. Chen, Z. Zhou, F. Liu, Z. Li, and S. Ren, "CloudHeat: An efficient online market mechanism for datacenter heat harvesting," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, vol. 3, no. 3, p. 31, 2018.

[111] S. Chen, L. Jiao, F. Liu, and L. Wang, "EdgeDR: An online mechanism design for demand response in edge clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 2, pp. 343–358, 2022.

[112] Z. Zhou, F. Liu, S. Chen, and Z. Li, "A truthful and efficient incentive mechanism for demand response in green datacenters," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 1, pp. 1–15, 2020.

[113] Q. Pei, S. Chen, Q. Zhang, X. Zhu, F. Liu, Z. Jia, Y. Wang, and Y. Yuan, "CoolEdge: hotspot-relievable warm water cooling

for energy-efficient edge datacenters," in *Proceedings of the 27th International Conference on Architectural Support for Programming Languages and Operating Systems*, 2022, pp. 814–829.
[114] Dr. Scott K. Thomas, "Heat conduction equation," http://cecs.wright.edu/~sthomas/htchapter02.pdf.

**Qiangyu Pei** received the BS degree in physics from the Huazhong University of Science and Technology, China, in 2019. He is currently working toward the PhD degree in the School of Computer Science and Technology, Huazhong University of Science and Technology. His research interests include edge computing, green computing, and deep learning.

**Shutong Chen** received her B.Sc. degree in the College of Mathematics and Econometrics, Hunan University, China. She is currently a Ph.D. student in the School of Computer Science and Technology, Huazhong University of Science and Technology, China. Her research interests include edge computing, green computing, and datacenter energy management.

**Yongjie Yuan** received the B.Eng. degree from the School of Computer Science and Technology, Huazhong University of Science and Technology, China, in 2021, where he is currently pursuing the M.Eng. degree. His research interests include edge computing and serverless computing.

**Qixia Zhang** received his Ph.D. degree in 2021 and B.Eng. degree in 2016 from School of Computer Science and Technology, Huazhong University of Science and Technology, China. His research interests include network function virtualization, cloud computing and edge computing, datacenter and green computing, 5G network and network slicing. He is a recipient of the Best Paper Award of IEEE/ACM IWQoS 2019.

**Xinhui Zhu** received her M.S. degree in Computer Science and Technology from Huazhong University of Science and Technology, Wuhan, China, in 2021. She received her B.Eng. degree in Software Engineering from Hunan University, Changsha, China, in 2018. Her research interests include green computing and datacenter energy.

**Ziyang Jia** received his B.E. degree from Huazhong University of Science and Technology, Wuhan, China, in 2021. He is a Ph.D. student in University of California, Riverside now. His research interest includes computer architecture, high-performance computing, and GPU architecture.

**Fangming Liu** (S'08, M'11, SM'16) received the B.Eng. degree from the Tsinghua University, Beijing, and the Ph.D. degree from the Hong Kong University of Science and Technology, Hong Kong. He is currently a Full Professor with the Huazhong University of Science and Technology, Wuhan, China. His research interests include cloud computing and edge computing, datacenter and green computing, SDN/NFV/5G and applied ML/AI. He received the National Natural Science Fund (NSFC) for Excellent Young Scholars, and the National Program Special Support for Top-Notch Young Professionals. He is a recipient of the Best Paper Award of IEEE/ACM IWQoS 2019, ACM e-Energy 2018 and IEEE GLOBECOM 2011, the First Class Prize of Natural Science of Ministry of Education in China, as well as the Second Class Prize of National Natural Science Award in China.