





# Carbon-Aware Computing for Datacenters

Ana Radovanović, Ross Koningstein , Ian Schneider, Bokan Chen , Alexandre Duarte , Binz Roy, Diyue Xiao, Maya Haridasan, Patrick Hung, Nick Care, Saurav Talukdar, Eric Mullen, Kendal Smith, MariEllen Cottman , and Walfredo Cirne

**Abstract**—The amount of CO<sub>2</sub> emitted per kilowatt-hour on an electricity grid varies by time of day and substantially varies by location due to the types of generation. Networked collections of warehouse scale computers, sometimes called Hyperscale Computing, emit more carbon than needed if operated without regard to these variations in carbon intensity. This paper introduces Google’s system for global Carbon-Intelligent Compute Management, which actively minimizes electricity-based carbon footprint and power infrastructure costs by delaying temporally flexible workloads. The core component of the system is a suite of analytical pipelines used to gather the next day’s carbon intensity forecasts, train day-ahead demand prediction models, and use risk-aware optimization to generate the next day’s carbon-aware Virtual Capacity Curves (VCCs) for all datacenter clusters across Google’s fleet. VCCs impose hourly limits on resources available to temporally flexible workloads while preserving overall daily capacity, enabling all such workloads to complete within a day with high probability. Data from Google’s in-production operation shows that VCCs effectively limit hourly capacity when the grid’s energy supply mix is carbon intensive and delay the execution of temporally flexible workloads to “greener” times.

**Index Terms**—Carbon- and efficiency-aware compute management, datacenter computing, power management.

## I. INTRODUCTION

**D**EMAND for computing resources and datacenter power worldwide has been continuously growing, now accounting for approximately 1% of total electricity usage [1]. Between 2010 and 2018, global datacenter workloads and compute instances increased more than sixfold [1]. In response, new methodologies for increasing datacenter power and energy efficiency are required to limit their growing environmental, economic and performance impacts [2], [3].

Greenhouse gas emissions from electricity production vary substantially by time and location [4]–[7]. This wide variation in carbon intensity (average greenhouse gas emissions per unit of energy consumption) or marginal emissions (the

additional greenhouse gas emissions per additional unit of electricity consumption) imply that the time and location of electricity consumption has a large effect on its associated global warming impact. Assessing and forecasting system conditions to improve the scheduling of flexible demand have been identified as high-priority areas for machine learning efforts to combat climate change [8].

The datacenter industry has the potential to facilitate carbon emissions reductions in electricity grids. A considerable fraction of compute workloads have flexibility in both when and where they run. Given that emissions from electricity production vary substantially by time and location [4]–[7], we can exploit load flexibility to consume power where and when the grid is less carbon intensive. By effectively managing its load, the datacenter industry can contribute to a more robust, resilient, and cost-efficient energy system, facilitating grid decarbonization. Electric grid operators, in turn, can possibly benefit by as much as EUR 1B/year [9]. Furthermore, shifting execution of flexible workloads in time and space can decrease peak demand for resources and power. Since datacenters are planned based on peak power and resource usage, smaller peaks reduce the need for more capacity. Not only does this save money, it also reduces environmental impact.

Carbon-aware datacenter load management is not a novel concept. In addition to a wide range of work focusing on reducing energy consumption of datacenter hardware equipment, including IT, cooling, and supply systems (see [3], and information on the standardized actuators for energy-driven management of IT equipment in [10]), it has been recognized that sustainable datacenters require intelligent and unifying solutions for energy-aware management of both datacenter hardware and software architectures [11]. Previous treatments of compute load management have mainly focused on self-managed datacenters, i.e. where a single company manages all the infrastructure necessary to support their computing needs. However, there are also some recent proposals of stylized, decentralized optimization models used to incentivize colocation tenants to carbon- and cost-effectively manage their power demand [12], [13].

Carbon- and cost-aware compute management was previously addressed via theoretical treatments, small-scale prototypes and simulation-based studies. These research investigations discuss two types of workload shifting: (i) across datacenter locations, and (ii) in time, by delaying jobs’ execution.

Investigations related to shifting computing across datacenter locations studied the impact of real-time rebalancing of serving requests, cloud VM migration and workload placement

Manuscript received 11 August 2021; revised 27 January 2022 and 26 April 2022; accepted 30 April 2022. Date of publication 6 May 2022; date of current version 27 February 2023. This work was supported by Google. Paper no. TPWRS-01286-2021. (Corresponding author: Bokan Chen.)

The authors are with the Google Inc., Mountain View, CA 94043 USA (e-mail: anaradovanovic@google.com; ross@google.com; ischneid@google.com; bokan.pro@gmail.com; alexandredu@google.com; binzroy@google.com; diyuexiao@google.com; haridasan@google.com; phfhung@google.com; ncare@google.com; stalukdar@google.com; ericmullen@google.com; kendal-smith@google.com; meacottman@google.com; walfredo@google.com).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TPWRS.2022.3173250>.

Digital Object Identifier 10.1109/TPWRS.2022.3173250

that co-optimizes datacenter portfolio cost and environmental objectives [14]–[22]. In addition, to increase power efficiency and environmental benefits, some of the research explored powering down redundant machines. Furthermore, it has been commonly assumed that datacenters are colocated with renewable assets, and the studied optimization models often incorporated forecasts of the onsite renewable supply. The implications of power rebalancing on energy markets, renewable portfolio, customer latency, and avoided carbon dioxide emissions were discussed in [21]. In addition to the real-time carbon-aware datacenter power management, [16], [23] investigated long-term datacenter planning that cost-effectively “follows” renewables. Finally, to the best of our knowledge, the first-at-scale in-production system that uses real-time marginal carbon intensity at cloud clusters’ grid locations to decide where to place cloud VMs was recently announced in [24].

Research investigations for temporal shifting of flexible compute jobs proposed an optimization-based framework with stylized models for job-level resource demand modeling [25]–[27], and server-level power consumption. Here, the goal was to delay jobs to “greener” hours of the day, where the delay decisions are the result of trade-offs between environmental, cost, and modeled performance objectives. These theoretical investigations suggest optimization-based, job-level scheduling decisions, which is infeasible for large-scale, real-life, hyper-scale datacenters with large number of flexible job classes and available machines with different hardware configurations. Also, the previous research explorations typically assume that the scheduled jobs’ real-time resource and power usage are controllable while, in real datacenter operations, they can be highly variable within the allocated resource constraints. Furthermore, rather than previously proposed stylized models of demand uncertainty and its translation to power consumption, a load shaping framework for Google-scale datacenters is required to capture large diversity in workloads and hardware configurations. No previously discussed methodology incorporates risk associated with application and infrastructure performance expectations. Finally, none of the proposed solutions has ever been implemented in-production and at Google scale.

This paper describes the methodology and principles behind Google’s global, in-production system for Carbon-Intelligent Compute management, which reduces grid carbon emissions from Google’s datacenter electricity use, as well as its operating costs via increased resource and power efficiency. To accomplish this goal, the system harnesses the temporal flexibility of a significant fraction of Google’s internal workloads that tolerate delays as long as their work gets completed within 24 hours. Typical examples of such workloads are data compaction, machine learning, simulation, and data processing (e.g., video processing) pipelines – many of the tasks that make information found through Google products more accessible and useful. Note that other loads include user-facing services (Search, Maps and YouTube) that people rely on around the clock, and our cloud customers’ workloads running in allocated Virtual Machines (VMs), which are not temporally flexible and therefore not affected by the new system.

Methodology for effective carbon-aware shifting Google datacenter workloads in time and across locations faces several

challenges. Workloads, comprised of compute jobs, are characterized by their arrival patterns, resource usage, dependencies and placement consequences, which generally have high uncertainty and are hard to predict (i.e., we do not know in advance what jobs will run over the course of the next day). Also, Google’s load management is required to meet the established reliability principles, which guarantees timely detection and mitigation of system vulnerabilities. Furthermore, at all times, workload performance expectations need to be met, while respecting hard infrastructure limits, such as cluster-level machine capacity limits, circuit breaker limits, etc. Importantly, workload management is required to keep the workload scheduler’s complexity as little as possible so that it can cope with the high volume (hundreds of thousands) of job scheduling requests every second [28]. The opportunity to effectively manage Google’s flexible workloads lies in the fact that their resource usage and daily consumption at a cluster-level and beyond are quite predictable within a day-ahead forecasting horizon. As a consequence, the aggregate outcome of job scheduling ultimately affects global costs, carbon footprint, and future resource utilization.

The core of Google’s in-production carbon-aware load shaping mechanism is a set of cluster-level [29] Virtual Capacity Curves (VCCs), which are hourly resource usage limits that serve to shape each cluster resource and power usage profile over the following day. These limits are computed using an optimization process that takes account of aggregate flexible and inflexible demand predictions and their uncertainty, hourly carbon intensity forecasts [30], explicit characterization of business and environmental targets, infrastructure and workload performance expectations, and usage limits set by energy providers for different datacenters across Google’s fleet. The cluster-level VCCs are pushed to all of Google’s datacenter clusters prior to the start of the next day, where they set hourly limits for total flexible compute usage. These limits directly impact real-time admission of flexible workloads. As a consequence, at times of day when the local grid’s carbon intensity is expected to be high, the corresponding clusters’ VCC values tend to be smaller, which reduces their total compute and power usage [31]. The reduction of usage is achieved via delaying scheduling and execution of flexible computing tasks to later times of the day.

The scope of the presented Carbon-Intelligent Computing System (CICS) is global in that the VCC curves shift load to lower overall carbon impact wherever Google locates its datacenters, regardless of the generation source of local lower-carbon energy (Fig. 1). This helps Google achieve its environmental, efficiency, and performance targets across the world.

To the best of our knowledge, the CICS encompasses the first methodology with the accompanying Google-scale and in-production implementation of carbon-aware algorithms that shift datacenter computing in time to realize global environmental and efficiency objectives [32] using automated adjustments based on current and forecasted grid conditions. The design and implementation of such a system requires novel theory, engineering and data. This paper discusses a new methodology behind the CICS. Its core comprises of a suite of analytical pipelines that result in the computed VCCs. The paper discusses the analytical

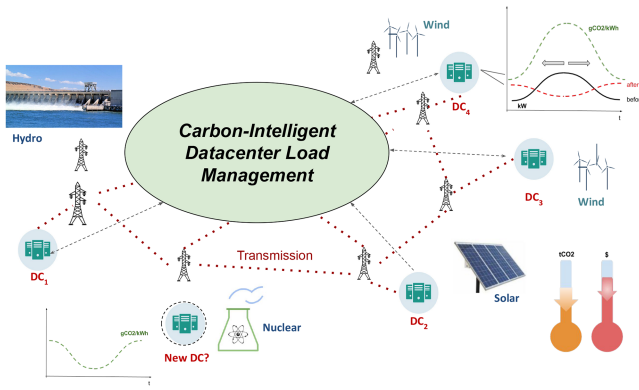


Fig. 1. Carbon-Intelligent management of Google's datacenter portfolio. The green dashed curves associated with specific datacenter locations are intraday carbon intensities of the local electricity generation. The impact of CICS on intraday datacenter power consumption is depicted using black (original) and dashed red (shaped) curves. With CICS, datacenter power consumption during peak carbon intensity hours is intended to be lower than without CICS.

methods, implemented in production, and demonstrated in real life to effectively and in a risk-aware manner shape all Google clusters' power consumption globally. Google's CICS impacts load in more than 20 datacenters on 4 continents [33], consuming more than 15.5 terawatt hours of electricity [34]. Specifically, the paper contributes with:

- *The overall CICS design*, including its integration with the existing Google's load management system, as well as the system monitoring, and a feedback loop that evaluates the recent application-level impact and adjusts load shaping accordingly. This will be discussed in more detail throughout the paper.
- *A detailed overview of CICS's analytical pipelines*, used for (i) cluster-level compute load characterization, (ii) power modeling [31], and (iii) risk-aware optimization that incorporates the treatment of workload and infrastructure performance expectations, as well as datacenter power-contract information.
- *Performance evaluation* using data from real-life datacenter operations.
- *Discussion on extensibility* of the proposed approach to incorporate different business objectives and operational constraints, including shifting compute load across locations.

The CICS complies with the established engineering principles in mind, such as

- *Reliable*: The system is built with reliability principles in mind, with an established monitoring and feedback loop which ensures that the system operates as intended and that it adapts to changes in resource usage trends and system upgrades.
- *Modular*: Due to the nature of Google's workload, the day-ahead, load-shaping optimization runs independently from real-time job-level scheduling, long-term resource planning, and user-level interfaces with the compute infrastructure.

- *Scheduler-agnostic*: CICS operation is independent from the real-time job scheduler. It interfaces with the scheduler only via the aggregate capacity value, used as a constraint beyond which no more jobs can be scheduled. To that end, the real-time job scheduler can evolve on its own, irrespective of the CICS.

The paper is organized as follows. Section I is followed by a basic overview on Google's datacenter power architecture, as well as the key concepts behind the real-time management of compute workload in Section II. The end of Section II provides a high level introduction and the key design principles behind the mechanism implemented to shape Google's datacenter compute load and the related power usage in a carbon- and efficiency-aware manner. Section III contains the details of the analytic pipelines that enable the load shaping mechanism. Section IV demonstrates the effectiveness of the deployed load shaping mechanism. The paper is concluded in Section V.

## II. DATACENTER ARCHITECTURE AND CLOUD COMPUTING OVERVIEW

Most of Google's compute resources reside in Google-designed datacenters with proprietary power distribution, cooling, networking and compute hardware [35], [36]. The premise of carbon-aware computing based on day-ahead planning is that a known amount of computing, translated into power usage and optimized based on expected grid carbon intensity, yields the best placement of work. Furthermore, managing peak power requires a good understanding of how workload resource usage translates to power. Therefore, it is important to have a good model of how resource usage and power inter-relate, and this requires a from-the-basics model of the complete datacenter power architecture.

### A. Power Architecture

Every datacenter is connected to the electricity grid via several medium voltage feeders. Each medium voltage distribution line is transformed to supply low voltage Power Distribution Units, also known as Power Domains (PDs). PDs are connected to bus ducts. The bus ducts supply power to the IT and cooling equipment. More information on a typical Google datacenter power architecture can be found [36], [31].

The IT equipment on the datacenter floor comprises compute, storage, and networking racks. A PD typically has a few thousand machines, and a handful of PDs comprise a cluster. Each PD is metered at a single PDU. The PDs in each cluster belong to a single job-scheduling domain, i.e., a common real-time scheduler that assigns computing tasks to its feasible machines. Generator backup is available to keep the datacenter running in the event of a grid power outage.

### B. Google's Real-Time Resource Management and Its Reliability Principles

Machines at Google are set up to run any application, and connected via high bandwidth switches within a campus, and via a global backbone of network connecting datacenters. Datacenter

hardware is controlled and administered by specialized software that can handle massive scale. To the extent possible, hardware controls, job scheduling, etc., are abstracted away from users.

Compute jobs at Google are managed by a distributed cluster-level operating system (known as Borg [28], [29]). These jobs can be roughly split into two categories: (i) indefinitely running servers, and (ii) batch processes (e.g., data processing pipelines using MapReduce or Flume [37], [38]). Jobs can consist of several tasks (sometimes thousands), both for reasons of reliability and because a single process can't usually handle all traffic. The cluster operating system is responsible for task allocation across machines within a cluster, which includes starting a job, finding machines for its tasks (i.e., task scheduling), allocating requested resources (CPU/RAM/disk) on machines, and instructing the machines to start executing the tasks. Since the scheduler needs to make hundreds of thousands of placement decisions per second [28], it is important that the scheduling algorithm is computationally inexpensive and, therefore, fast, generally allowing jobs to flow into available compute resources like fluid into containers. The cluster operating system continually monitors and, potentially, reschedules tasks in case of problems. At Google, all available machines are typically turned on within a datacenter unless they are broken.

In this paper, we divide jobs into 2 categories: temporally inflexible and flexible (i.e., batch jobs that tolerate delays). We assume that the flexible workload assigned to a cluster can be subject to delays, as long as the amount of computation they perform during a day is preserved. Scheduling and the associated resource allocation is managed using the estimated upper bound (referred to reservations) of a task's actual usage across all resource dimensions (CPU, memory, disk, etc.). If resources are not available to run a job's tasks, they are queued. The admission controller visits this queue periodically, trying to enable jobs that pass a series of checks, one of which is cluster-level resource availability.

The following subsection describes how the temporal flexibility of the flexible workload can be exploited to reduce carbon footprint and peak power usage of Google's datacenter portfolio.

### C. Mechanism for Carbon- and Cost-Aware Load Shaping

This section discusses the framework we deployed for shaping intraday resource usage using time shifting of flexible workload in Google's compute clusters, with the goal to decrease both its electricity-based carbon footprint and improve overall resource efficiency (and, therefore, its long-term build costs).

The aggregate cluster-level load shaping is achieved using a Virtual Capacity Curve (VCC), which artificially limits the cluster's hourly compute usage and, in view of the power modeling advancements in [31], its hourly power usage as well. Borg uses the VCC values to compute the real-time CPU availability for incoming flexible jobs. Reducing cluster-level CPU capacity (i.e., total allowed CPU usage limits) at times of day when the corresponding grid carbon intensity is high or when it is cost-effective to do so, prevents the scheduler from starting as many jobs then, to reap the carbon or economic benefit. Flexible jobs get queued until resources become available. CPU

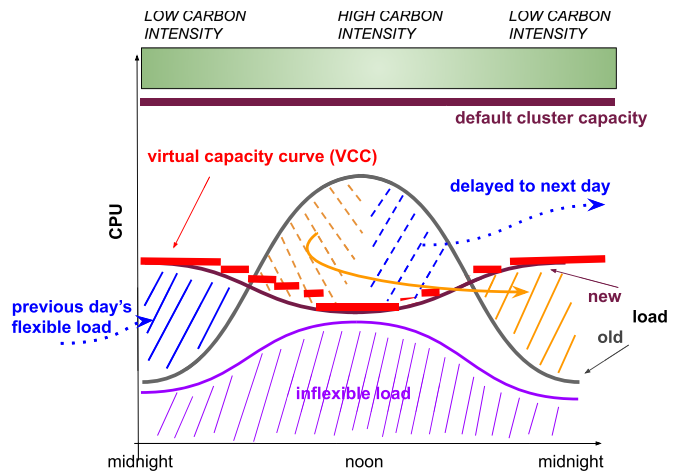


Fig. 2. Effect of using the VCC mechanism for load shaping in a cluster.

load shaping affects resource usage across other datacenter resource dimensions as well (e.g., memory, disk, spindles, etc.). In aggregate, resource consumption is highly correlated to CPU consumption. Consequently, reduction of CPU consumption leads to predictable reductions of power consumption across all resources [31]. The entire system can be viewed as a two-step optimization framework. In the first step, VCC assigns capacity into different clusters based on the carbon intensity and demand. In the second step, Borg schedules the jobs based on the assigned capacity and job characteristics. Most data center operators should already have a scheduler analogous to Borg in place. Since VCC simply changes one of the input parameters (available capacity in each hour), it can be adopted easily by any data center operators for carbon footprint reduction.

An example of the effect that the VCC curve has on cluster CPU load shape is demonstrated in Fig. 2. VCC (in red) has lower values in the middle of the day when the intraday carbon intensity values are the highest. Flexible usage (orange and blue shaded areas) is pushed from midday to evenings and early mornings when the carbon intensity is lowest. In addition to carbon-aware load shifting in time, the proposed shaping mechanism reduces daily peak CPU and, consequently, power consumption. Note that the shifting should only impact flexible workloads, without affecting inflexible jobs in any way. Also, when delaying the execution of the flexible jobs, their users should be impacted in an unbiased way.

## III. LOAD SHAPING ANALYTICS

In this section, we describe the methodology for computing cluster-level VCCs. The implemented system comprises several components that: (i) predict the next day's load, (ii) train models that map CPU usage to power consumption, (iii) retrieve the next day's predictions for average carbon intensities on electrical grids where Google's datacenters reside, (iv) run day-ahead, risk-aware, optimization to compute VCCs, and (v) check for flexible workload SLO violations and trigger a feedback mechanism. As described earlier, the computed curves are used to reshape the corresponding cluster's intraday CPU usage and,

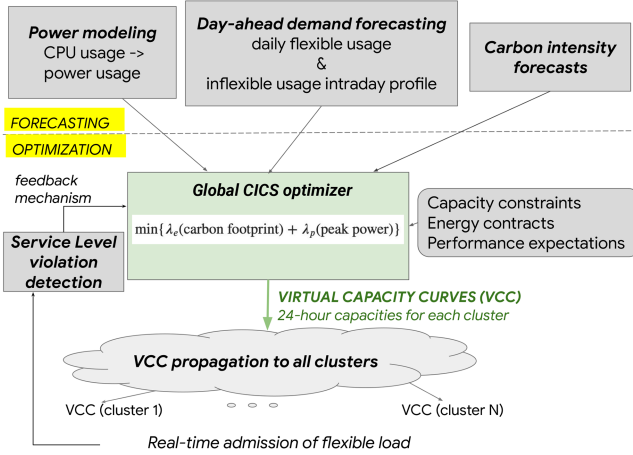


Fig. 3. Architecture of the Carbon-Intelligent Computing system (CICS).

therefore, power consumption (see more details below) by affecting only flexible (e.g., batch) workload.

In view of the above, the suite of analytics pipelines (see Fig. 3) encompasses:

- Carbon fetching pipeline, which reads hourly average carbon intensity forecasts from Tomorrow [30] for each electricity grid zone where Google’s datacenters reside.
- Power models pipeline, which trains statistical models to map CPU usage to power consumption for each power domain (PD) across Google’s datacenter fleet. The models are retrained and evaluated daily in a parallelized manner, and are demonstrated to be highly accurate irrespective of the underlying power architecture and machine platforms installed within a PD [31]. The ability to accurately translate changes in power domains and the corresponding clusters CPU usage to changes in power consumption enables carbon-aware load shaping, and is the core component of the CICS.
- Load forecasting pipeline, which generates the next day’s cluster-level forecasts for flexible and inflexible workload demand. Since inflexible load cannot be shaped, we predict its next day’s hourly usage profile and the associated uncertainty. On the other hand, flexible load is considered shapeable as long as its total daily compute (CPU) demand is preserved. To that end, we predict the next day’s flexible load compute usage, which turns out to be far more predictable than its typical hourly usage profile. In addition, the forecasting pipelines include accurate models of the relationship between the total cluster workload usage and reservations. Note that the forecasting errors are embedded within the proposed risk-aware optimization framework to make sure that Google infrastructure and applications’ SLOs are preserved (see Section III-B1 for more details).
- Optimization pipeline, which runs daily to co-optimize the next day’s fleetwide expected carbon footprint and appropriately-scaled power peaks, subject to infrastructure and application SLO constraints, contractual and resource capacity limits, as discussed in Section III-C below. The results of the optimization are cluster-level virtual capacity

curves setting optimal capacity values for each hour of the next day. We use  $VCC^{(c)}(h)$  to denote the optimal virtual capacity for cluster  $c$  at hour  $h$  of the next day.

- SLO violation detection flags when a cluster’s daily flexible demand is not met. The violation can happen due to the unpredicted growth in flexible or inflexible usage (which, consequently, reduces resource available for flexible, lowest tier, workload). When the measured cluster flexible daily demand starts to persistently exceed the computed violation threshold (see Section III-B2 for details), the CICS stops shaping these clusters for a week, to allow load forecasting models to adapt to changes in demand.

Daily analytics pipelines are scheduled at different times of a day. Data collection and modeling pipelines generate the next day’s predictions (in particular, all usage data at Google is timestamped using Pacific Standard Time (PST)). These are then used by the central optimizer once per day to compute optimal next day VCCs for all clusters fleetwide. The generated VCCs span 24 hours of the next day tracked in Pacific Time (PT).

In the rest of the section, we discuss the details of the implemented analytical approaches within the CICS. We use capital letters to denote stochastic processes, and lowercase letters to denote their realizations. Also, the operator  $\hat{\cdot}$  is used to denote the predicted value of a forecasted variable.

#### A. Power Modeling

There have been many studies that suggest simple models to capture the relationship between machine/PD/cluster CPU usage and its power consumption [31], [39]–[41]. The recent study [31], evaluated across all of Google’s power domains, demonstrated that a PD power consumption can be accurately estimated using only its CPU usage as a resource usage feature. Moreover, this general conclusion was shown to hold irrespective of the hardware heterogeneity across power domains (e.g., diversity in machine types, such as compute, storage, accelerators, etc.)

More specifically, as discussed in [31], a piecewise linear model accurately captures the relationship between CPU usage and dynamic power consumption for a given PD. Using a rigorous evaluation methodology and usage data at 5-minute time granularity, it was demonstrated that the daily Mean Absolute Percent Error (MAPE) of the proposed model is less than 5% for more than 95% of PDs. In our testing, we also find that the impact of a change in a PD CPU usage on power can be accurately locally approximated as

$$\begin{aligned} & Pow^{(PD)}(u_{CPU}^{(PD)} + \Delta u_{CPU}^{(PD)}) - Pow^{(PD)}(u_{CPU}^{(PD)}) \\ & \approx \pi^{(PD)}(u_{CPU}^{(PD)}) \Delta u_{CPU}^{(PD)} \end{aligned} \quad (1)$$

where  $Pow^{(PD)}(u_{CPU}^{(PD)})$  is used to denote PD power consumption when its CPU usage equals  $u_{CPU}^{(PD)}$ ,  $\pi^{(PD)}(u_{CPU}^{(PD)})$  is power model’s slope at  $u_{CPU}^{(PD)}$  for PD, and  $\Delta u_{CPU}^{(PD)}$  denotes a change in CPU usage of power domain PD.

While the power models are developed per PD, the workload scheduler works on a per cluster basis. It assigns a large number of computing tasks in real time to randomly selected

TABLE I  
NOTATION

Variables	Definition
$U_{IF}^{(c)}(h), U_F^{(c)}(h)$	Inflexible and flexible CPU usage of cluster $c$ at hour $h$ , $h \in d$ .
$R_{IF}^{(c)}(h), R_F^{(c)}(h)$	Inflexible and flexible CPU reservations of cluster $c$ at hour $h$ , $h \in d$ .
$T_{U,F}^{(c)}(d)$	Daily amount of flexible compute usage (in CPU-hour) of cluster $c$ on day $d$ , defined as integral of cluster $c$ flexible CPU usage over day $d$ .
$T_R^{(c)}(d)$	Daily amount of CPU reservations of cluster $c$ on day $d$ , defined as integral of all cluster $c$ reservations over day $d$ .
$\Theta^{(c)}(d)$	cluster-level, SLO-based, capacity requirement, i.e. $\sum_{h \in d} VCC^{(c)}(h) = \Theta^{(c)}(d)$ .
$\mathcal{R}^{(c)}(h)$	CPU reservations-to-usage ratio ( $\geq 1$ ) for cluster $c$ at hour $h$ , $h \in d$ . It is a function of total cluster CPU usage.
$\tau_U^{(c)}(d)$	Risk-aware daily flexible compute usage for cluster $c$ on day $d$ , $\tau_U^{(c)}(d) = \alpha^{(c)}(d)T_{U,F}^{(c)}(d)$ , where $\alpha^{(c)}(d)$ is a risk-aware inflation factor computed to ensure that Service Level Objectives are met (Subsection III-B2).

feasible machines in a given cluster. As a consequence, it is observed that the resulting CPU usage fractions across PDs within the same cluster varies insignificantly across time, and we denote them using  $\lambda^{(PD)}$ . Thus, (1) and  $Pow^{(c)}(u_{CPU}^{(c)}) = \sum_{PD \in c} Pow^{(PD)}(u_{CPU}^{(PD)})$ , we can obtain the cluster level power slope with the PD level power slopes with the following approximation:

$$\begin{aligned}
& Pow^{(c)}(u_{CPU}^{(c)} + \Delta u_{CPU}^{(c)}) - Pow^{(c)}(u_{CPU}^{(c)}) \\
&= \sum_{PD \in c} \left( Pow^{(PD)}(u_{CPU}^{(PD)} + \Delta u_{CPU}^{(PD)}) - Pow^{(PD)}(u_{CPU}^{(PD)}) \right) \\
&\approx \sum_{PD \in c} \pi^{(PD)}(u_{CPU}^{(PD)}) \Delta u_{CPU}^{(PD)} \\
&\approx \left( \sum_{PD \in c} \pi^{(PD)}(u_{CPU}^{(PD)}) \lambda^{(PD)} \right) \Delta u_{CPU}^{(c)} \\
&\equiv \pi^{(c)}(u_{CPU}^{(c)}) \Delta u_{CPU}^{(c)},
\end{aligned} \tag{2}$$

where  $\pi^{(c)}(u_{CPU}^{(c)}) \equiv \sum_{PD \in c} \pi^{(PD)}(u_{CPU}^{(PD)}) \lambda^{(PD)}$  represents cluster  $c$  power usage sensitivity with respect to its CPU usage.

## B. Day-Ahead Forecasting

The proposed risk-aware optimization framework for computing VCCs requires a forward looking view of the next day's compute demand and carbon intensities. It also requires a method for detecting when flexible workload SLOs are not met, and mechanisms to respond to these events. The effectiveness of the proposed shaping in this paper is mainly due to the high prediction accuracy of the aggregated flexible and inflexible demands, as well as that of the next day's carbon intensities, provided by [30].

1) *Load Forecasting*: The day-ahead forecasting pipeline of Google's CICS predicts next day cluster-level: (i) hourly inflexible compute (CPU) usage,  $U_{IF}^{(c)}(h)$ , (ii) daily flexible compute usage,  $T_{U,F}^{(c)}(d) = \sum_{h \in d} U_F^{(c)}(h)$ , (iii) daily total compute

reservations,  $T_R^{(c)}(d) = \sum_{h \in d} (R_{IF}^{(c)}(h) + R_F^{(c)}(h))$ , and (iv) ratio between total workload reservations and usage, denoted as  $\mathcal{R}^{(c)}(h)$ .

The next day  $d$  inflexible hourly CPU usage,  $U_{IF}^{(c)}(h)$ ,  $h \in d$ , daily flexible compute usage,  $T_{U,F}^{(c)}(d)$ , and daily total compute reservations,  $T_R^{(c)}(d)$ , are forecasted using a two-step approach. First, we compute the week-ahead hourly/daily predictions. Then, to adapt to day-to-day deviations from weekly patterns, we augment the week-ahead forecasts based on the deviation of the previous day's measurements from the corresponding week-ahead predictions using a linear regression model.

The week-ahead hourly/daily forecasts are obtained as a product of the next week's predictions for weekly mean value and hourly/daily factors. The detailed steps are described as follows:

- Forecast weekly mean value using Exponential Weighted Moving Average (EWMA), with a half-life of 0.5 (i.e., decay rate equal to 0.45).
- Calculate historical intra-week hourly factors by dividing historical hourly/daily usage (the raw data, which includes 24/1 data points each day of the 90-day history per cluster) by the corresponding weekly mean value.
- Forecast future hourly/daily factors using EWMA with a half-life of 4 (i.e., decay rate equal to 0.07).
- Compute the week-ahead forecasts by multiplying the weekly mean value forecast with the intra-week hourly/daily factors forecasts.

The EWMA parameters are selected via exploration over a given range so that out-of-sample Mean Absolute Percent Error (MAPE) is minimized.

Once the week-ahead forecasts are available, we train the following linear regression model to augment them:

$$\begin{aligned}
& (\text{augmented hourly/daily forecast}) \approx \\
& a + b \times (\text{week-ahead forecast}) + \\
& c \times (\text{yesterday's deviation from week-ahead forecast});
\end{aligned}$$

Yesterday's deviation from week-ahead forecast is calculated as actual measurements subtracted by the week-ahead forecasts.  $a, b, c$  are the constants obtained from training the linear regression model that minimizes out-of-sample MAPE. To this end, simple linear regression model has proven effective in predicting the adapted next day predictions (see Section IV for more details), and is applied to forecasting  $U_{IF}^{(c)}(h)$ ,  $T_{U,F}^{(c)}(d)$  and  $T_R^{(c)}(d)$ . Once computed these forecasts are used by the optimizer as discussed in Section III-C below.

It is observed that the ratio between the all load CPU reservations and usage,  $\mathcal{R}^{(c)}(h)$ , is primarily driven by the compute (CPU) usage. In particular, the larger the CPU usage of a cluster is, the smaller the ratio. To capture the observed trend, we built a linear regression model with  $\log(\text{usage})$  as the feature to predict the reservation to usage ratio. The computed ratios are used by the optimizer to translate the computed next day optimal usage profiles into VCCs.

2) *Service Level Objective Awareness*: The main constraint within the implemented load shaping framework corresponds

to workload SLO expectations, i.e., when a temporally flexible workload is shifted in time, its cluster-level daily compute usage must be preserved. The daily flexible compute usage is a stochastic process and, as previously discussed, is fairly predictable. In view of that, we set the SLO for its violation, which we explicitly embed into the proposed framework, and monitor for validation and safety. We define the target: the cluster-level daily amount of flexible compute (i.e., flexible load's daily capacity requirement) cannot be violated more often than one day per month when averaged across a given time horizon. This translates to roughly 3 days within a 100-day time horizon, which is equivalent to roughly 0.03 upper bound for violation probability. To meet this SLO, each cluster's VCC must ensure that the total amount of compute reservation demand satisfies

$$\sum_{h \in d} VCC^{(c)}(h) = (T_R^{(c)}(d))._{97}, \quad (3)$$

since flexible workload throughput gets violated when there is not enough capacity to meet the daily demand of flexible and inflexible demand altogether.

We compute the corresponding daily resource requirement using the methodology discussed in Subsection III-B1. A violation of the introduced SLO happens either due to the unpredicted growth in flexible demand, or due to the unpredicted growth in inflexible workload. The 97%-ile of the total daily capacity requirement (i.e., all-load daily compute reservations) is computed using the previous 90-day relative errors of the day-ahead predictions:

$$\begin{aligned} \Theta^{(c)}(d) &= (T_R^{(c)}(d))._{97} \\ &= \left( \hat{T}_R^{(c)}(d) \right) \left( 1 + (\{\epsilon^{(c)}(n)\}_{n=d-90, \dots, d-1})._{97} \right) \end{aligned} \quad (4)$$

where  $\epsilon^{(c)}(n)$  is the next day evaluation of the relative prediction error for day  $n$ , and  $\hat{T}_R^{(c)}(d)$  is day  $d$  prediction for next day's all-load compute reservations for cluster  $c$ .

If the actual daily reservations demand in cluster  $c$  gets close to  $\Theta^{(c)}(d)$  limit for two days in a row, the system considers it a sign of usage violation, and triggers the feedback mechanism. While there are different ways to cope with the unpredicted demand growth, one option is to stop load shaping for some time (e.g., a week) to allow load forecasting models to adapt to changes in load demand.

To ensure that the CPU capacity limit in (3) is met in the optimal daily planning process (described in detail in Subsection III-C below), we attribute all the "extra" capacity to the daily amount of flexible usage by inflating its forecasted value with factor  $\alpha^{(c)}(d)$  computed to satisfy

$$\sum_{h \in d} \left( \hat{U}_{IF}^{(c)}(h) + \alpha^{(c)}(d) \frac{\hat{T}_{U,F}^{(c)}(d)}{24} \right) \hat{\mathcal{R}}^{(c)}(h) = \Theta^{(c)}(d) \quad (5)$$

where  $\hat{\mathcal{R}}^{(c)}(h)$  is the predicted reservations-to-usage ratio corresponding to the nominal expected CPU usage at hour  $h$ , i.e.  $\hat{U}_{IF}^{(c)}(h) + \frac{\hat{T}_{U,F}^{(c)}(d)}{24}$ , on day  $d$ . In the rest of the paper, we use  $\tau_U^{(c)}(d) = \alpha^{(c)}(d) \hat{T}_{U,F}^{(c)}(d)$  to denote the inflated, risk-aware, daily flexible usage.

3) *Carbon Intensity Forecasting*: The optimization methodology embedded into the CICS retrieves the near-term (48-hour) forecasts for average carbon intensities from Tomorrow (electricityMap.org), for each data center location for which we run the optimization. Tomorrow's approach accounts for demand, generation, and imports, to estimate the average carbon intensity of grid consumption in each particular region [30]. We manually map each datacenter to one of Tomorrow's grid regions. Since a datacenter often contains many colocated clusters, the forecasted and actual carbon intensities are identical for clusters located in the same physical datacenter. To compute the optimal capacity plan for the next day, the optimizer uses the carbon intensity forecast for each location and hour of the next day,  $\eta^{(c)}(h)$ , obtained in kgCO<sub>2</sub>e / kWh (here e means equivalent). The optimizer runs at 6pm PT, using forecasted hourly carbon data for each location and for each hour of the following day, from the hour starting at 12am PT to the hour starting at 11pm PT. Thus, the forecast horizon ranges from 6-32 hours. The evaluated error of Tomorrow's hourly carbon intensity forecasts depend on weather forecasts and the forecast horizon. The carbon intensity forecast MAPE, computed for different grid locations where Google datacenters reside, ranges from 0.4% - 26% over the range of forecast horizons (6-32 hours) for the day-ahead forecast.

### C. Optimization Framework

Our risk-, cost- and carbon-aware load shaping approach uses day-ahead forecasts to compute the next day's optimal capacity values,  $VCC^{(c)}(h)$ , for each hour and cluster fleetwide. The uncertainty of the next day's predictions strongly impact the effectiveness of the proposed approach, and the proposed optimization methodology is carefully designed to harness predictable workload, environmental and infrastructure parameters.

The optimizer's objective is to derive next day's hourly reservation capacities that minimize the weighted sum of expected carbon footprint and daily power peak values summed over all clusters fleetwide, i.e.,

$$\begin{aligned} &\lambda_e \sum_{c,h} \eta^{(c)}(h) (Pow^{(c)}(\hat{U}_{nom}^{(c)}(h))) \\ &+ \pi^{(c)}(\hat{U}_{nom}^{(c)}(h)) \delta(c, h) \frac{\tau_U^{(c)}(d)}{24} + \lambda_p \sum_c y^{(c)}(d) \end{aligned} \quad (6)$$

where  $\lambda_e$  is the cost of 1 kg/CO<sub>2</sub>e generated carbon footprint (\$/ kg CO<sub>2</sub>e) and  $\lambda_p$  is the cost associated with power infrastructure costs driven by clusters peak power consumption (\$/ MW / day). These costs are set internally and subject to change. Currently, we have  $\lambda_e = 40$  and  $\lambda_p = 10$ .  $\hat{U}_{nom}^{(c)}(h) \equiv \tau_U^{(c)}(d)/24 + \hat{U}_{IF}^{(c)}(h)$  represents cluster  $c$  nominal, risk-aware, CPU usage at hour  $h$  of next day obtained by adding hourly prediction for inflexible CPU usage and the average hourly risk-aware flexible compute usage as defined in (5). Matrix  $\delta$  (n x 24 matrix) is used to denote hourly deviations of flexible CPU usage from its average hourly target,  $\tau_U^{(c)}(d)/24$ ; for example, a value of  $\delta(c, h) = -0.1$  would arise when the VCC allows for 10% less than the nominal flexible CPU usage in cell  $c$  in

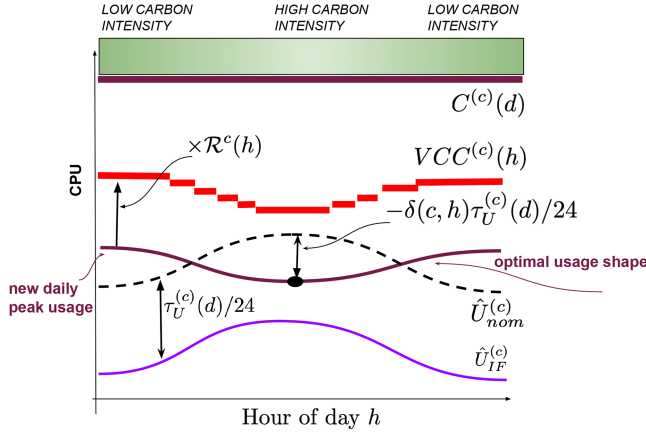


Fig. 4. An example of using  $\delta(c, \cdot)$  to control cluster-level CPU (and power usage) shape so that both its carbon footprint and daily usage peaks are reduced.

hour  $h$ . The variable  $y^{(c)}(d)$  is cluster  $c$  upper bound for its daily peak power consumption. The goal is to compute optimal values for  $\delta$  and  $y^{(c)}(d)$ . An example on how  $\delta(c, \cdot)$  is used to control cluster-level CPU usage shape so that its carbon footprint and daily peak power usage are reduced, and how the optimal usage shape is translated into the corresponding VCC, is included in Fig. 4. Note that (6) does not seek to minimize electricity generation costs; this is not part of our objective, but could be relevant for other users, especially those that see high variance in hourly electricity prices.

The optimization needs to ensure that application and infrastructure SLO constraints are honored, as well as contractual constraints that define the maximum datacenter power demand.

Daily usage conservation constraint for temporally flexible workloads translates into

$$\sum_h \delta(c, h) = 0, \forall c. \quad (7)$$

Power capping constraint is a power infrastructure SLO constraint, which sets a threshold for cluster-level CPU usage,  $\bar{U}_{pow}^{(c)}$ , to prevent power domains' circuit breakers from tripping ([31], [39]). In particular, cluster  $c$  compute usage can exceed the given threshold with probability less than or equal to a given parameter  $0 < \gamma \ll 1$ . Therefore, to meet the SLO,  $\mathbb{P}[U_{IF}^{(c)}(h) + (1 + \delta(c, h))\tau_U^{(c)}(d)/24 \geq \bar{U}_{pow}^{(c)}] \leq \gamma$ . Since the amount of flexible load in each hour can be upper-bounded by the VCC value,  $(1 + \delta(c, h))\frac{\tau_U^{(c)}(d)}{24}$ , this is equivalent to setting an upper bound on the maximum allowable flexible load in each hour:

$$(1 + \delta(c, h))\frac{\tau_U^{(c)}(d)}{24} \leq \bar{U}_{pow}^{(c)} - (U_{IF}^{(c)}(h))_{1-\gamma}, \forall c, h. \quad (8)$$

We use  $(\cdot)_{1-\gamma}$  in the previous expression to denote  $(1 - \gamma)$ th quantile evaluated using historical day-ahead predictions and actual, measured inflexible CPU usage, as discussed in Subsection III-B1.

Campus-level energy contracts set power usage limits for some Google datacenters,  $L_{cont}^{(dc)}$ , which the optimizer enforces

by bounding the sum of cluster level peak power usage as

$$\sum_{c \in dc} y^{(c)} \leq L_{cont}^{(dc)}, \forall dc. \quad (9)$$

*Cluster-level total machine capacity:* the next day's optimal cluster-level CPU reservations profile cannot exceed its total machine capacity,  $C^{(c)}(d)$ . Therefore, the virtual capacity curve values are computed as

$$\begin{aligned} VCC^{(c)}(h) &= \left( \hat{U}_{IF}^{(c)}(h) + (1 + \delta(c, h))\frac{\tau_U^{(c)}(d)}{24} \right) \hat{\mathcal{R}}^{(c)}(h), \forall c, h, \end{aligned} \quad (10)$$

where

$$VCC^{(c)}(h) \leq C^{(c)}(d). \quad (11)$$

*Other constraints:* Note that there are other constraints that could be incorporated into the optimization. For example, a constraint could be added to bound the allowed drop in intraday flexible usage, or to bound hour-to-hour changes in  $VCC^{(c)}(\cdot)$  values. Also, the listed constraints can be incorporated inside the objective terms (as soft constraints) using an appropriately large penalty and function form (e.g., hinge, quadratic, etc.).

*Carbon vs peak power consumption cost:* Note that by using an objective function that incorporates both carbon footprint and cluster-level power consumption peaks as in expression (6), Google decreases its load's expected global carbon footprint while also reducing demand for future infrastructure builds required to support its workload.

In summary, our optimizer solves the following optimization model:

$$\begin{aligned} \min_{\delta, y} & \text{Equation (6)} \\ \text{s.t.} & \text{Equation (7) – Equation (11)} \end{aligned}$$

#### IV. DEMONSTRATION AND IMPACT

The impact of the carbon-aware computing approach can be observed across Google's fleet, spanning different electricity grids. The magnitude of these benefits, however, varies significantly from location to location. This section evaluates the impact of the proposed shaping mechanism by analysing operational data, showing how it is affected by (i) the amount of flexible usage, (ii) the high uncertainty range in the computed demand forecast, (iii) the intraday variability and magnitude of grid carbon intensity.

The load forecasting models are trained and evaluated daily for all clusters across Google's datacenter fleet. For each cluster, we compute absolute percent error (APE) of all day-ahead predictions across a chosen 3-month-long time horizon. Then, we compute their median, 75%-ile, and 90%-ile, and we plot the distribution of their values for all clusters fleetwide. The results are shown in Fig. 5 for hourly inflexible CPU usage ( $U_{IF}^{(c)}(h)$ ), daily flexible compute usage ( $T_{U,F}^{(c)}(d)$ ), daily total compute reservations ( $T_R^{(c)}(d)$ ), and hourly reservations-to-usage ratio ( $\mathcal{R}^{(c)}(h)$ ) predictions.



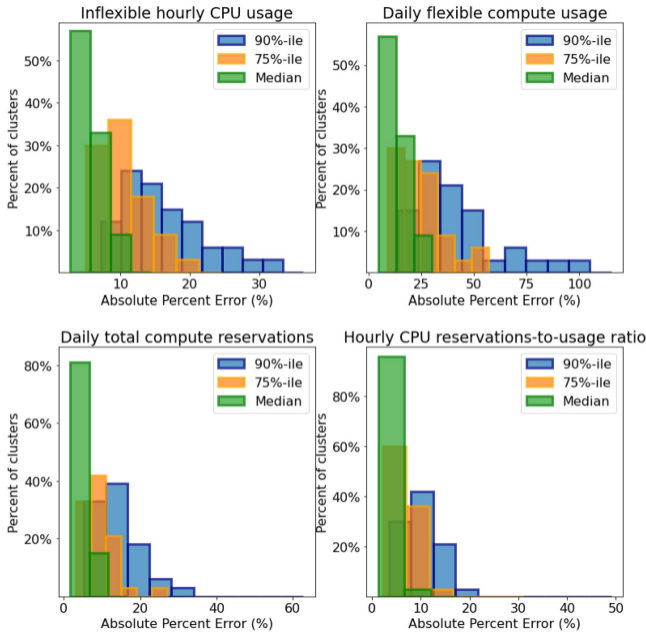


Fig. 5. Percent of clusters (axis y) with median, 75%-ile and 90%-ile Absolute Percent Errors (rounded to the nearest 3% increment) within a given range set by x axis. The x axis shows absolute percent error, and while outliers are in the 50-100% range, the majority have far smaller errors. Consequently, these results suggest that the load is predictable.

We can see that the median APEs of inflexible usage, total load reservations, and reservations-to-usage ratio predictions are smaller than 10% for more than 90% of the clusters. Daily flexible compute usage forecasts have larger APEs, which is not surprising given that flexible demand is typically more variable. The rare, high, APEs (greater than 50%) are sometimes observed for clusters that have small flexible usage, or that are going through a typically sudden, transient increase in flexible demand especially in new clusters. In order to comply with Google's SLOs, our optimizer makes decisions based on the 97th percentile of the forecasts. Large prediction error would mean that the 97th percentile forecast could be much larger than the actual load. As a result, the optimizer would assign large capacities to the clusters, and the capacity curves would be unable to suppress demand. Thus these rare scenarios result in either inactive or ineffective shaping of the specific clusters on the particular days (see Subsection III-C), and are omitted when Fig. 5 was created. We also compared the results of the proposed forecasting model with that of a structural time-series model, and find that their performance are comparable in metrics including MAPE, and confidence interval coverage.

To show these impacts, we present shaping results from two clusters within a large Google campus on a selected day in Figs. 6, 7. The clusters and date were chosen because they help illustrate how the predictability of compute demand impacts the effectiveness of load shaping. In each figure, the top graph captures real-time compute reservations (blue) constrained by a VCC (red). The bottom graph shows the normalized power used by the data center (orange), with the power's carbon-intensity (black) which influenced the VCC.

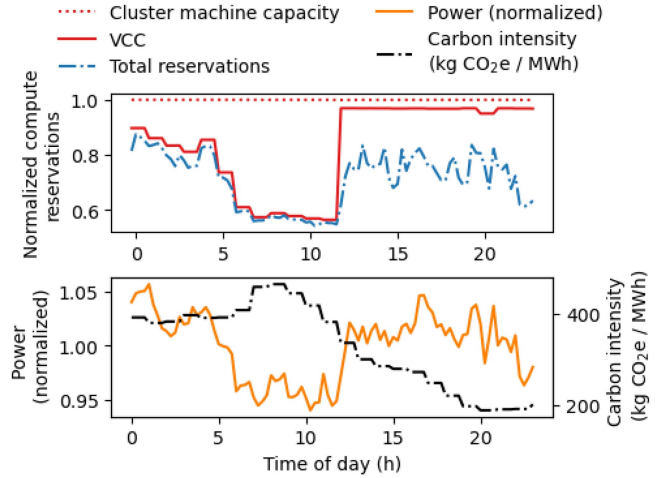


Fig. 6. Hourly VCC, compute reservations, cluster power, and carbon intensity in cluster  $X$  on the selected day.

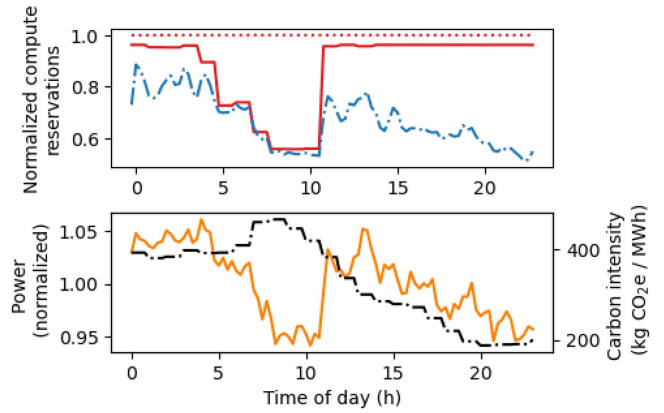


Fig. 7. Hourly VCC, compute reservations, cluster power, and carbon intensity in cluster  $Y$  on the selected day.

In the first cluster,  $X$ , the average value of the VCC is about 18% higher than the average load demand. This difference is due to the uncertainty in the load forecast, because our forecast value for the 97th quantile for load demand is 18% higher than the actual daily load. In this location, the VCC is able to drop flexible load by roughly 50% during peak carbon intensity hours driving an 8% drop in power during the hours with peak carbon intensity (see Fig. 6).

In another location,  $Y$ , uncertainty in the forecasts for inflexible and flexible load drives the VCC higher (see Fig. 7). Here, the average value of the VCC is about 33% higher than the average load demand. The VCC is still able to drop by almost 50% during peak carbon intensity hours, but the drop is not as sustained (its duration is shorter). This drives a roughly 8% decrease in power during the hours with peak carbon intensity, but the duration is only 3 hours versus 6 hours in cluster  $X$ , reducing the carbon impact of shaping. The higher predictability of load in cluster  $X$  allows for more effective shaping and higher carbon reductions in cluster  $X$  than in cluster  $Y$ .

Note that a VCC is set to a cluster total machine capacity when a cluster is too full to allow for shaping, for example when the

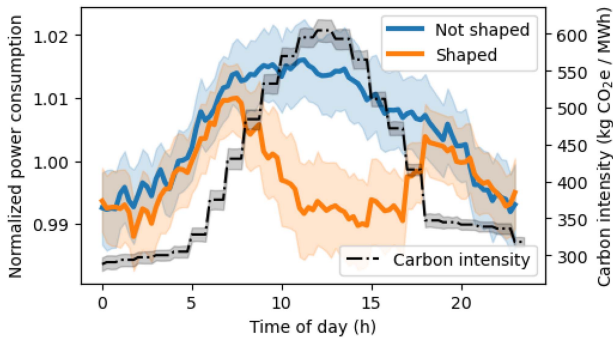


Fig. 8. The effect of load shaping optimization on cluster power consumption, averaged across all clusters and test days in a single Google datacenter campus.

risk-aware total daily compute reservations is larger than cluster machine capacity. The same happens when there is insufficient data for forecasting or estimating power models.

The VCCs have disparate impacts across datacenter locations, and their impacts may change over time as demand patterns and daily carbon intensity patterns change. Therefore, it is helpful to evaluate aggregate impacts on campuses and Google’s fleet. To that end, we ran a controlled experiment to evaluate the impact of day-to-day shaping on power consumption. Fig. 8 shows the normalized power curves, averaged across all datacenter clusters in a campus, on randomly treated (optimized) and non-treated (not optimized) days for two months beginning February 12th, 2021. On each day, each cluster is randomly assigned to receive the carbon-aware optimal shaping or not, with 50% probability of being in each group on any given day. The solid lines plot normalized power, over the course of the day, averaged over all data center clusters within a selected campus for 1 mo, for shaped (orange) and not shaped (blue) clusters. The (black) dashed line displays the average (actual) carbon intensity in each hour over the course of the day for the grid that it is powered by. In each line, the uncertainty band reflects the 95% confidence interval in the mean value for the hour, averaged across days and clusters.

In this example, when CICS is active, average cluster power drops by 1-2% during the highest carbon hours, compared to power in the same clusters on dates that the cluster is placed in the control group and not shaped by the carbon-aware computing mechanism. We also observe that, in load shaping regimes where we allow for *larger and longer drops* in capacity (obtained by increasing the cost associated with the carbon footprint,  $\lambda_e$ , in (6) and by relaxing conditions for  $\delta$ ), total daily flexible compute and, consequently, power usage tends to slightly decrease in the shaped clusters. This happens because some flexible jobs “choose” to move to different clusters in response to lower and durable virtual capacity limits imposed by the carbon-aware computing mechanism, as a result of which, the flexible compute conservation condition fails to hold. The “spontaneous” load shifting to other locations may increase or decrease carbon emissions. To harness the spatial flexibility to carbon- and cost-effectively redistribute flexible load across both time and locations, future models will explicitly characterize spatially flexible demand and extend the proposed optimization framework to take it into consideration.

## V. CONCLUDING REMARKS

The growing climate impact of increased Greenhouse Gas Emissions and CO<sub>2</sub> levels in Earth’s atmosphere highlights the value and importance of technologies that reduce such impact. Electricity generation is one of the larger contributors to global CO<sub>2</sub> emissions [42]. The datacenter industry accounts for an expanding electricity demand, expected to reach anywhere from 3 to 13% of global electricity demand by 2030 [43]. Yet, it has the potential to facilitate grid decarbonization in a manner different from isolated power loads.

This paper introduces Google’s Carbon-Intelligent Computing System, which shifts datacenter computing in time and will soon also shift computing in space. These together will help realize the company’s global environmental [44] and efficiency objectives. The system proactively makes automated adjustments based on current and forecasted grid conditions to reliably and effectively shape Google’s compute load in a carbon- and efficiency-aware manner. The core of the carbon-aware load shaping mechanism are cluster-level [29] Virtual Capacity Curves, which are hourly resource usage limits that serve to shape the cluster resource and power usage profile over the following day. These limits are computed using an optimization process that takes account of aggregate flexible and inflexible demand predictions and their uncertainty, hourly carbon intensity forecasts [30], explicit characterization of business and environmental targets, infrastructure and workload performance expectations, and usage limits set by energy providers for different datacenters across Google’s fleet.

Using actual measurements from Google datacenter clusters, we demonstrate a power consumption drop of 1-2% at times with the highest carbon intensity. Ongoing system enhancements, which include shifting flexible workloads across locations, are expected to increase the benefits of this system. A valuable topic for future work would be to compare results between this proposed strategy and other existing data center load shifting methods, if they are implemented in large-scale datacenters. The framework and principles embedded in Google’s Carbon Intelligent Computing system align with its compute management systems and workload properties. While other compute providers’ approaches to carbon-aware computing will necessarily vary, we hope that the initial results demonstrated in this paper inspire academia and industry to pursue diverse approaches to individual cluster or hyperscale computing system management.

## REFERENCES

- [1] E. Masanet, A. Shehabi, N. Lei, S. Smith, and J. Koomey, “Recalibrating global data center energy-use estimates,” *Science*, vol. 367, no. 6481, pp. 984–986, 2020.
- [2] M. Dayarathna, Y. Wen, and R. Fan, “Data center energy consumption modeling: A survey,” *IEEE Commun. Surv. Tut.*, vol. 18, no. 1, pp. 732–794, Jan.–Mar. 2016.
- [3] U. Hölzle, “Data centers are more energy efficient than ever,” Google Blog, Feb. 27, 2020. [Online]. Available: <https://blog.google/outreach-initiatives/sustainability/data-centers-energy-efficient>
- [4] M. Li, T. M. Smith, Y. Yang, and E. J. Wilson, “Marginal emission factors considering renewables: A case study of the US midcontinent independent system operator (MISO) system,” *Environ. Sci. Technol.*, vol. 51, no. 19, pp. 11215–11223, 2017.

- [5] M. P. Thind, E. J. Wilson, I. L. Azevedo, and J. D. Marshall, "Marginal emissions factors for electricity generation in the midcontinent ISO," *Environ. Sci. Technol.*, vol. 51, no. 24, pp. 14445–14452, 2017.
- [6] D. S. Callaway, M. Fowlie, and G. McCormick, "Location, location, location: The variable value of renewable energy and demand-side efficiency resources," *J. Assoc. Environ. Resour. Economists*, vol. 5, no. 1, pp. 39–75, 2018.
- [7] I. Khan, "Temporal carbon intensity analysis: Renewable versus fossil fuel dominated electricity systems," *Energy Sources, Part A: Recovery, Utilization, Environ. Effects*, vol. 41, no. 3, pp. 309–323, 2019.
- [8] D. Rolnick *et al.*, "Tackling climate change with machine learning," *ACM Comput. Surv. (CSUR)*, vol. 55, no. 2, pp. 1–96, Feb. 8, 2022.
- [9] P. Vassilopoulos and M. Gandhi, "Sustainable markets: Activating flexibility through integrated energy markets," EPEX SPOT and SDIA Report, Jul. 2020. [Online]. Available: [https://www.epexspot.com/sites/default/files/download\\_center\\_files/20-07\\_EPEX%20SPOT%20-%20SDIA\\_Whitepaper%20Data%20Centers%20in%20Energy%20markets\\_clean.pdf](https://www.epexspot.com/sites/default/files/download_center_files/20-07_EPEX%20SPOT%20-%20SDIA_Whitepaper%20Data%20Centers%20in%20Energy%20markets_clean.pdf)
- [10] "Advanced configuration and power interface specification revision 6.3," Unified Extensible Firmware Interface Forum, Jan. 2019. [Online]. Available: [https://uefi.org/sites/default/files/resources/ACPI\\_6\\_3\\_final\\_Jan30.pdf](https://uefi.org/sites/default/files/resources/ACPI_6_3_final_Jan30.pdf)
- [11] J. Guitart, "Toward sustainable data centers: A comprehensive energy management strategy," *Computing*, vol. 99, pp. 597–615, 2017.
- [12] A. H. Mahmud and S. S. Iyengar, "A distributed framework for carbon and cost aware geographical job scheduling in a hybrid data center infrastructure," in *Proc. IEEE Int. Conf. Autonomic Comput. (ICAC)*, 2016, pp. 75–84.
- [13] M. A. Islam, S. Ren, and X. Wang, "Greencolo: A novel incentive mechanism for minimizing carbon footprint in colocation data center," in *Proc. Int. Green Comput. Conf.*, 2014, pp. 1–8.
- [14] K. Le, R. Bianchini, T. D. Nguyen, O. Bilgir, and M. Martonosi, "Capping the brown energy consumption of internet services at low cost," in *Proc. Int. Conf. Green Comput.*, 2010, pp. 3–14.
- [15] X. Deng, D. Wu, J. Shen, and J. He, "Eco-aware online power management and load scheduling for green cloud datacenters," *IEEE Syst. J.*, vol. 10, no. 1, pp. 78–87, Mar. 2016.
- [16] J. Berral, Í. Goiri, T. Nguyen, R. Gavalda, J. Torres, and R. Bianchini, "Building green cloud services at low cost," in *Proc. IEEE 34th Int. Conf. Distrib. Comput. Syst.*, 2014, pp. 449–460.
- [17] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew, "Geographical load balancing with renewables," in *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 39, pp. 62–66, 2011.
- [18] Z. Liu, M. Lin, A. Wierman, S. Low, and L. Andrew, "Greening geographical load balancing," *IEEE/ACM Trans. Netw.*, vol. 23, no. 2, pp. 657–671, Apr. 2015.
- [19] J. Zheng, A. Chien, and S. Suh, "Mitigating curtailment and carbon emissions through load migration between data centers," *Joule*, vol. 4, no. 10, pp. 2208–2222, 2020.
- [20] A. Rahman, X. Liu, and F. Kong, "A survey on geographic load balancing based data center power management in the smart grid environment," *IEEE Commun. Surv. Tut.*, vol. 16, no. 1, pp. 214–233, Jan.–Mar. 2014.
- [21] C. Kelly, J. Ging, A. Kansal, and M. Walsh, "Balancing power systems with datacenters using a virtual interconnector," *IEEE Power Energy Technol. Syst. J.*, vol. 3, no. 2, pp. 51–59, Jun. 2016.
- [22] A. James and D. Schien, "A low carbon kubernetes scheduler," in *Proc. ICT4S*, Jan. 2019. [Online]. Available: [http://ceur-ws.org/Vol-2382/ICT4S2019\\_paper\\_28.pdf](http://ceur-ws.org/Vol-2382/ICT4S2019_paper_28.pdf)
- [23] C. Ren, D. Wang, B. Urgaonkar, and A. Sivasubramaniam, "Carbon-aware energy capacity planning for datacenters," in *Proc. IEEE 20th Int. Symp. Model., Anal. Simul. Comput. Telecommun. Syst.*, 2012, pp. 391–400.
- [24] B. Johnson, "Carbon-aware kubernetes," Microsoft DevBlogs, Oct. 5, 2020. [Online]. Available: <https://devblogs.microsoft.com/sustainable-software/carbon-aware-kubernetes/>
- [25] Z. Liu *et al.*, "Renewable and cooling aware workload management for sustainable data centers," in *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 40, pp. 175–186, 2012.
- [26] Í. Goiri *et al.*, "Greenslot: Scheduling energy consumption in green datacenters," in *Proc. Int. Conf. High Perform. Comput. Netw. Storage Anal.*, 2011, pp. 1–11.
- [27] Í. Goiri *et al.*, "GreenHadoop: Leveraging green energy in data-processing frameworks," in *Proc. 7th ACM Eur. Conf. Comput. Syst.*, 2012, pp. 57–70.
- [28] M. Tirmazi *et al.*, "Borg: The next generation," in *Proc. EuroSys' 20*, Heraklion, Crete, 2020, pp. 1–14.
- [29] A. Verma, L. Pedrosa, M. R. Korupolu, D. Oppenheimer, E. Tune, and J. Wilkes, "Large-scale cluster management at google with borg," in *Proc. Eur. Conf. Comput. Syst. (EuroSys)*, Bordeaux, France, 2015, pp. 1–17.
- [30] Electricity Map, "Tomorrow," [Online]. Available: <https://www.tmrw.com/>
- [31] A. Radovanović, B. Chen, S. Talukdar, B. Roy, A. Duarte, and M. Shahbazi, "Power modeling for effective datacenter planning and compute management," *IEEE Trans. Smart Grid*, vol. 13, no. 2, pp. 1611–1621, Mar. 2022.
- [32] A. Radovanovic, "Our data centers now work harder when the sun shines and wind blows," The keyword, Apr. 22, 2020. [Online]. Available: <https://blog.google/inside-google/infrastructure/data-centers-work-harder-sun-shines-wind-blows>
- [33] Google, "Discover our data center locations," Data centers. [Online]. Available: <https://www.google.com/about/datacenters/locations/>
- [34] C. Clifford, "How google plans to use 100% carbon-free energy in its data centers by 2030," CNBC, Apr. 13, 2022. [Online]. Available: <https://www.cnbc.com/2022/04/13/google-data-center-goal-100percent-green-energy-by-2030.html>
- [35] B. Beyer, C. Jones, J. Petoff, and N. R. Murphy, *Site Reliability Engineering: How Google Runs Production Systems*. Sebastopol, CA, USA: O'Reilly Media, Inc., 2016.
- [36] L. A. Barroso, U. Hölzle, and P. Ranganathan, "The datacenter as a computer: Designing warehouse-scale machines," *Synth. Lectures Comput. Architecture*, vol. 13, no. 3, pp. 1–189, 2018.
- [37] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," in *Proc. 6th Symp. Operating Syst. Des. Implementation*, 2004, pp. 137–150.
- [38] S. McVeety and R. Lippert, "Dataflow under the hood: The origin story," Google Cloud, Aug. 20, 2020. [Online]. Available: <https://cloud.google.com/blog/products/data-analytics/how-cloud-batch-and-stream-data-processing-works>
- [39] X. Fan, W.-D. Weber, and L. A. Barroso, "Power provisioning for a warehouse-sized computer," *ACM SIGARCH Comput. Architecture News*, vol. 35, no. 2, pp. 13–23, 2007.
- [40] J. D. Davis, S. Rivoire, M. Goldszmidt, and E. K. Ardestani, "No hardware required: Building and validating composable highly accurate os-based power models," Microsoft Technical Report, Jul. 2011.
- [41] S. Rivoire, P. Ranganathan, and C. Kozyrakis, "A comparison of high-level full-system power models," *HotPower*, vol. 8, no. 2, pp. 32–39, 2008.
- [42] P. Hawken, *Drawdown: The Most Comprehensive Plan Ever Proposed to Reverse Global Warming*. Old Saybrook, CT, USA: Tantor Audio, 2017.
- [43] A. S. G. Andrae and T. Edler, "On global electricity usage of communication technology: Trends to 2030," *Challenges*, vol. 6, no. 1, pp. 117–157, 2015.
- [44] Google, "24/7 by 2030: Realizing a carbon-free future," 2020. [Online]. Available: <https://www.gstatic.com/gumdrop/sustainability/247-carbon-free-energy.pdf>