Towards Environmentally Equitable AI via Geographical Load Balancing

Pengfei Li
University of California, Riverside
pli081@ucr.edu

Jianyi Yang
University of California, Riverside
jyang239@ucr.edu

Adam Wierman
California Institute of Technology
adamw@caltech.edu

Shaolei Ren
University of California, Riverside
shaolei@ucr.edu

ABSTRACT

Fueled by the soaring popularity of foundation models, the accelerated growth of artificial intelligence (AI) models' enormous environmental footprint has come under increased scrutiny. While many approaches have been proposed to make AI more energy-efficient and environmentally friendly, environmental inequity — the fact that AI's environmental footprint can be disproportionately higher in certain regions than in others — has emerged, raising social-ecological justice concerns. This paper takes a first step toward addressing AI's environmental inequity by fairly balancing its regional environmental impact. Concretely, we focus on the carbon and water footprints of AI model inference and propose equity-aware geographical load balancing (eGLB) to explicitly minimize AI's highest environmental cost across all the regions. The consideration of environmental equity creates substantial algorithmic challenges as the optimal GLB decisions require complete offline information that is lacking practice. To address the challenges, we introduce auxiliary variables and optimize GLB decisions online based on dual mirror descent. In addition to analyzing the performance of eGLB theoretically, we run trace-based empirical simulations by considering a set of geographically distributed data centers that serve inference requests for a large language AI model. The results demonstrate that existing GLB approaches may amplify environmental inequity while eGLB can significantly reduce the regional disparity in terms of carbon and water footprints.

ACM Reference Format:
Pengfei Li, Jianyi Yang, Adam Wierman, and Shaolei Ren. 2024. Towards Environmentally Equitable AI via Geographical Load Balancing. In *The 15th ACM International Conference on Future and Sustainable Energy Systems (E-Energy '24), June 04–07, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3632775.3661938

1 INTRODUCTION

The success of artificial intelligence (AI) relies heavily on computationally intensive calculations to learn useful information from data during training and provide insightful predictions during inference. As such, AI models, especially large generative models like GPT-3 [9], are typically trained on large clusters of power-hungry servers that may each have multiple graphic processing units (GPUs) and are housed in warehouse-scale data centers. Moreover, for inference, AI models are often deployed in geographically distributed data centers to serve users with low transmission latency.

Consequently, the exponentially growing demand for AI has created an enormous appetite for energy as well as a negative impact on the environment [9, 29, 46, 59, 71, 75]. For example, putting aside the environmental toll of chip manufacturing (e.g., raw material extraction and toxic chemicals) [23, 57, 76] and the noise pollution of running AI servers [55], training a large language model like GPT-3 and LaMDA can easily consume hundreds of megawatt-hour of electricity, generate many tonnes of carbon emissions, and evaporate hundreds of thousands of liters of clean freshwater for cooling [43, 46, 79]. Crucially, in addition to their impacts on the global climate, AI's environmental footprint also has significant local and regional impacts. Elevated carbon emissions have localized social costs [13] and may increase local ozone, particulate matter, and premature mortality [34]; electricity generation, especially when burning fuels, produces *local* air pollutants, discharges pollution such as thermal pollution into water bodies, and generates solid wastes (possibly including hazardous wastes) [85]; and staggering water consumption, both directly for on-site cooling and indirectly for off-site electricity generation, can further stress the already-limited local freshwater resources and even worsen extended megadroughts in regions like Arizona [43, 80].

Fueled by the soaring popularity of large language and foundation models, the accelerated growth of AI's environmental footprint has come under increased scrutiny recently [25, 86]. To make AI more energy-efficient and environmentally friendly, research studies have pursued a variety of approaches, including computationally efficient training and inference [11, 65], energy-efficient GPU and accelerator designs [22, 59, 87], carbon-aware task scheduling [29, 86], green cloud infrastructures [1, 4, 18], sustainable AI policy recommendations [25, 57], among others. As supply-side solutions, data center operators have also increasingly adopted carbon-free energy such as solar and wind power, (partially) powering AI servers and lowering carbon emissions [20, 50, 86]. Additionally, to reduce on-site water consumption and mitigate the stress on already-limited freshwater resources, climate-conscious cooling system designs (e.g., using air-side economizers if the climate condition permits) have recently seen an uptick in the data center industry [21, 51].

E-Energy '24, June 04–07, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0480-2/24/06
https://doi.org/10.1145/3632775.3661938

While existing efforts are encouraging, a worrisome outcome — environmental inequity — has unfortunately emerged. That is, minimizing the total environmental cost of AI across multiple regions does not necessarily mean each region is treated equitably. In fact, AI's environmental footprint is often disproportionately higher in certain regions than in others, potentially exacerbating other unintended social-ecological consequences [83]. For example, a data center's on-site cooling water usage effectiveness (WUE, the ratio of water consumption to IT energy consumption) highly depends on the outside temperature [32] — while it can stay well below 1.0 L/kWh for data centers located in cooler climates, the monthly average WUE can be as high as 9.0 L/kWh in the summer in drought-stricken Arizona [37]. Likewise, there exists a significant regional difference in terms of the carbon efficiency — as of 2020, only 4% of the energy for Google's data center in Singapore is carbon-free, whereas this number goes up to 94% in Finland [59], creating a 23× disparity. Thus, as a result of such regional differences, certain data center locations are severely disadvantaged and more negatively impacted by the environmental toll of AI. Further compounded by enduring socioeconomic disparities and even potentially amplified by existing data center scheduling algorithms, environmental inequity of AI can pose critical business risks and hence needs to be properly reconciled.

Indeed, addressing its environmental inequity is increasingly important and becoming integral to responsible AI and computing [6, 58]. For example, in the first-ever global agreement to ensure healthy development of AI, the United Nations Educational, Scientific and Cultural Organization (UNESCO) recommends that "AI should not be used" if it creates "disproportionate negative impacts on the environment" [82]. The AI Now Institute even compares the uneven regional distribution of AI's environmental costs to "historical practices of settler colonialism and racial capitalism" in its 2023 Landscape report [35]. Among all the environmental-related topics, Meta ranks environmental *justice* as the most critical one with the greatest impact on its business risks and opportunities [50]. More recently, studies have also emerged to suggest new regulations pertinent to AI's growing environmental footprint [25], and holistic assessment of AI as social-ecological-technological systems using available tools from environmental justice [5, 66].

In this paper, we take a first step to address the emerging environmental inequity of AI by balancing its negative environmental impact across geographically distributed data centers. More concretely, we focus on the carbon and water footprints of AI model inference and dynamically schedule users' inference requests (also referred to as *workloads* in this paper) using equity-aware geographical load balancing (GLB) to fairly distribute AI's environmental cost to each region. To mitigate environmental inequity, our key novelty is to augment the traditional cost-saving objective by explicitly including minimization of the most significant negative environmental impacts among all the data centers.

Nonetheless, the consideration of environmental equity in GLB decisions creates substantial algorithmic challenges. Specifically, due to their dependency on the long-term carbon and water footprints, the equity-related costs couple all the GLB decisions over T time slots (see (4a)–(4d)). This means that the optimal GLB decisions require all the offline information (e.g., future workload arrivals and water efficiency) in advance, while we must make GLB decisions online without knowing all the future. To address this challenge, we propose a new online equity-aware GLB algorithm, called `eGLB`, which leverages online information to optimize the GLB decisions based on dual mirror descent. We also bound the cost performance of `eGLB` compared to the offline optimal equity-aware GLB algorithm (`eGLB-Off`).

To empirically evaluate our proposed equity-aware GLB, we run trace-based simulations by considering a set of 10 geographically distributed data centers that serve inference requests for a large language AI model. Our results demonstrate that the proposed equity-aware GLB can significantly reduce the carbon and water footprints in the most disadvantaged region. In stark contrast, existing carbon- and water-saving GLB approaches may even amplify environmental inequity, showing that minimizing the total environmental footprint does not necessarily treat each region fairly.

In summary, our work is the first study to advance AI's environmental equity via GLB, connecting research across data center scheduling, sustainable AI, and equitable AI. It highlights the need and great potential of equity-aware GLB to fairly distribute AI's environmental cost across different regions for environmental equity.

2 PROBLEM FORMULATION

While AI model training is energy-intensive, the environmental footprint of its inference phase is also enormous and can even be several times higher than the training process [12]. As such, we consider a pre-trained AI model (e.g., large language model) and focus on the inference phase. The AI model inference service is deployed over a set $\mathcal{N} = \{1, \cdots, N\}$ of geographically distributed data centers to serve users in different regions. There are a set $\mathcal{J} = \{1, \cdots, J\}$ of front-end traffic gateways that aggregate users' requests from their respective surrounding areas and assign the requests to data centers, which is also referred to as geographical load balancing (GLB) in the literature [12, 45]. The GLB decisions are made in a time-slotted manner over a total of T time slots. Each time slot can range from a few minutes to about an hour, depending on how frequently the decisions are updated. We also interchangeably use "workloads" and "requests" when referring to users' demand for the AI model inference service. Our model is consistent with those used in the literature such as [32, 45, 64].

Each data center houses a cluster of servers (typically each equipped with multiple GPUs) to host AI models for inference. For the ease of presentation, we assume a homogeneous AI model on all the servers, while the extension to heterogeneous AI models with different model sizes is considered in Appendix C. During each time slot, the maximum service capacity for the AI model inference is M_i for data center i. We use $\lambda_{j,t}$ to denote the total amount of workloads arriving at gateway j at time t, and $x_{ij}(t) \geq 0$ to represent the GLB decision (i.e., the load assigned to data center i from gateway j). For the convenience of presentation, we also use $x(t) = \{x_{i,j}(t) \mid i \in \mathcal{N}, j \in \mathcal{J}\}$ as the collection of all the GLB decisions at time t.

The total load assigned to data center i is $\sum_{j\in\mathcal{J}} x_{ij}(t) \leq M_i$ at time t, thus resulting in a total server energy consumption of $e_i(x(t))$ which is an increasing function of $\sum_{j\in\mathcal{J}} x_{ij}(t)$. For example, a common model [45] is $e_i(x(t)) = \rho_{i,t}\bar{E}_{i,s} + \frac{\sum_{j\in\mathcal{J}} x_{ij}(t)}{M_i} \cdot \bar{E}_{i,d}$ where $\bar{E}_{i,s}$ is the server cluster's static/idle energy even when no

workload is processed in data center i, $\bar{E}_{i,d}$ is the cluster's dynamic energy consumed when only processing workloads, $\frac{\sum_{j\in\mathcal{J}} x_{ij}(t)}{M_i}$ is the cluster-level utilization, and $\frac{\sum_{j\in\mathcal{J}} x_{ij}(t)}{M_i} \le \rho_{i,t} \le 1$ indicates how well the cluster is right-sized in proportion to the workloads (i.e., $\rho_{i,t} = \frac{\sum_{j\in\mathcal{J}} x_{ij}(t)}{M_i}$ means the cluster is perfectly sized to the workloads by turning off unused servers, while $\rho_{i,t} = 1$ means the servers are always kept on regardless of the assigned workloads).

Next, we model the energy cost, carbon footprint, and water footprint in terms of the GLB decisions. Here, we explicitly model carbon and water footprints separately, as they are two complementary and *non-substitutable* measures for ecological impacts [16].

Energy cost. Suppose that the electricity price and power usage effectiveness (PUE, which accounts for non-IT energy consumption such as cooling systems and power distribution losses) are $p_{i,t}$ and $\gamma_{i,t}$ for data center i at time t, respectively. Then, the total energy cost at time t can be written as

$$g_t(x(t)) = \sum_{i\in\mathcal{N}} p_{i,t}\gamma_{i,t}e_i(x(t)). \tag{1}$$

Note that, if the AI model inference service is run on virtual machine (VM) instances rented from public cloud providers, the electricity price $p_{i,t}$ becomes the VM price subject to the VM instance type and $g_t(x(t)) = \sum_{i\in\mathcal{N}} p_{i,t}e_i(x(t))$ is the total VM rental cost at time t where $e_i(x(t))$ represents the number of VM instances rented to process the assigned workloads in location i.

Carbon footprint. The carbon footprint of AI model inference is embedded in the generation of electricity using carbon-intensive fuels such as coal [19, 23, 46]. The carbon footprint for data center i at time t can be denoted as follows:

$$c_{i,t}(x(t)) = \alpha_{i,t}\gamma_{i,t}e_i(x(t)) \tag{2}$$

The carbon intensity $\alpha_{i,t}$ can be obtained by querying the local utility or averaging the carbon intensity of the grid's fuel mix [19].

Water footprint. In parallel with the carbon footprint, data centers' staggering water footprint has recently become a new focused area for sustainability (see, e.g., the pledge of "Water Positive by 2030" by big techs [21, 53]). To serve AI model inference, data centers consume clean freshwater both directly and indirectly [21, 33, 43, 74]. The direct water consumption comes from the cooling system to keep servers from overheating. AI servers use either air or closed-loop liquid to transfer the heat to the facility level (e.g., the facility cooling water loop or heat exchanger [73]). Then, to further reject the heat into the outside environment, data centers commonly use cooling towers due to their energy efficiency and applicability to a wide range of weather conditions. Nonetheless, a large amount of water is evaporated into the outside environment (i.e., not discharged or returned to the source) and hence considered "consumed" [21]. For example, depending on the outside wet-bulb temperature, a cooling tower typically consume 1~4 liters of water (up to 9 liters of water in the summer) for each kWh server energy [37]. Importantly, the vast majority of the cooling water supply is drinking-grade (e.g., nearly 90% for Google's U.S. data centers in 2021 [21]). Alternatively, air-side economizers (i.e., directly using outside air to cool down servers) can be used to save water if the climate condition is suitable, but water is still needed when the outside temperature is high and/or the humidity is low — Meta's state-of-the-art cooling systems use an average of 0.26 liters of water for each kWh server energy across its global data center fleet in 2021 [50]. AI systems are also accountable for water consumption embedded in the electricity generation process. For example, thermal and nuclear power plants require a large volume of water consumption for cooling, while hydropower consumes water by expediting evaporation downstream [68, 72].

Thus, the total water footprint for data center i at time t is

$$w_{i,t}\left(x(t)\right) = \left[\epsilon_{i,t} + \beta_{i,t}\gamma_{i,t}\right] \cdot e_i\left(x(t)\right), \tag{3}$$

where $\epsilon_{i,t}$ is the direct water usage effectiveness (WUE) for on-site cooling, $\beta_{i,t}$ is the indirect WUE for off-site electricity generation, and $\gamma_{i,t}$ is the PUE. The direct WUE is defined as the ratio of water consumption to IT server energy consumption [78], and hence we do not need to multiply $\gamma_{i,t}$ when calculating the direct water consumption. In practice, the direct WUE $\epsilon_{i,t}$ heavily depends on the outside temperature [37, 43]. Like the carbon intensity, the indirect WUE $\beta_{i,t}$ measures the water consumption per kWh electricity generation and can be calculated by averaging over the water intensity of different energy fuels [32]. The monetary price for on-site water consumption is typically much smaller compared to the energy cost and can be factored into the price $p_{i,t}$ for modeling purposes.

3 ENVIRONMENTALLY EQUITABLE GLB

To make AI environmentally equitable, we propose a novel online equity-aware GLB algorithm, called eGLB, to distribute AI's environmental cost across different data centers in a fair manner.

3.1 Objective

Our goal is not to blindly *equalize* its regional environmental footprint, which, as similarly observed in the context of mitigating AI's algorithmic unfairness [14], may artificially elevate the environmental footprints in those otherwise advantaged regions and provide a false sense of fairness. Instead, we adopt the notion of *minimax* fairness [14, 49, 77] and exploit the power of GLB as a software-based approach to explicitly minimize AI's environmental impact on the most disadvantaged region.

Mathematically, we augment the traditional cost-saving objective by including the minimization of the greatest environmental cost among all the data centers. By normalizing the energy cost and environmental footprints over T, our equity-aware GLB problem is formulated as follows

$$\min_{x(t)} \frac{1}{T}\sum_{t=1}^{T} g_t(x(t)) + \mu_c \cdot \max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,c}\left(\frac{1}{T}\sum_{t=1}^{T} c_{i,t}(x(t))\right)\right] + \mu_w \cdot \max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,w}\left(\frac{1}{T}\sum_{t=1}^{T} w_{i,t}\left(x(t)\right)\right)\right], \tag{4a}$$

$$s.t., \quad x_{i,j}(t) = 0, \ \text{ if } B_{i,j} = 0, \quad \forall\, i \in \mathcal{N}, j \in \mathcal{J}, t = 1, \cdots, T, \tag{4b}$$

$$\sum_{j\in\mathcal{J}} x_{i,j}(t) \le M_i, \quad \forall\, i \in \mathcal{N}, t = 1, \cdots, T, \tag{4c}$$

$$\sum_{i\in\mathcal{N}} x_{i,j}(t) = \lambda_{j,t}, \quad \forall\, j \in \mathcal{J}, t = 1, \cdots, T, \tag{4d}$$

where the assignment condition $B_{i,j} = 0$ indicates that the workloads cannot be assigned from gateway j to data center i (due to, e.g., latency constraints or data sovereignty regulations) and hence enforces $x_{i,j} = 0$ in (4b), the constraint (4c) means that the total workloads assigned to a data center cannot exceed its processing capacity, and the constraint (4d) requires that all workloads arriving at a gateway be assigned to data centers. In the optimization objective (4a), the monotonically-increasing convex functions $\mathcal{H}_{i,c}()$ and $\mathcal{H}_{i,w}()$ quantify the environmental impacts of AI on data center i due to its long-term carbon footprint and water footprint, respectively, and can be specified based on the local environment assessment. Note that the carbon footprint is also a good indicator of the amount of local air/thermal pollution caused by our GLB decisions. For example, coal-based energy sources are carbon-intensive and also proportionally create air and thermal pollution for local communities [85].

Using $\mathcal{H}_{i,w}\left(\frac{1}{T}\sum_{t=1}^{T} w_{i,t}\left(x(t)\right)\right) = \frac{\theta_i}{T}\cdot\sum_{t=1}^{T} w_{i,t}\left(x(t)\right)$ as an illustrative example, we can set a higher $\theta_i \geq 0$ if data center i is located in a severely water-stressed and drought-prone region. In line with the principle of proportionality, the carbon footprint $\sum_{t=1}^{T} c_{i,t}(x(t))$ in $\mathcal{H}_{i,c}()$ and water footprint $\sum_{t=1}^{T} w_{i,t}(x(t))$ in $\mathcal{H}_{i,w}()$ for data center i can also be normalized by the maximum processing capacity M_i to achieve proportional fair distribution of AI's environmental cost.

The two functions $\mathcal{H}_{i,c}()$ and $\mathcal{H}_{i,w}()$ are general enough and can also capture the effects of additional sustainability practices that data center operators may adopt (e.g., installing solar for carbon mitigation and restoring watersheds for local water supply [20, 50]). The term $\sum_{t=1}^{T} g_t(x(t))$ in (4a) is the total energy cost. The hyperparameters $\mu_c \geq 0$ and $\mu_w \geq 0$ indicate the relative importance weights of carbon footprint equity and water footprint equity, respectively, and can be flexibly tuned to balance the impact of carbon and water footprints. For example, by setting $\mu_c = 0$, we focus solely on the negative environmental impact of AI's water footprint. In addition, we can also include into (4a) AI's other environmental impacts such as concerns with the servers' noise pollution if applicable [55].

Importantly, the cost terms $\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,c}\left(\frac{1}{T}\sum_{t=1}^{T} c_{i,t}(x(t))\right)\right]$ and $\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,w}\left(\sum 1T\sum_{t=1}^{T} w_{i,t}\left(x(t)\right)\right)\right]$ improve environmental equity by explicitly penalizing the greatest environmental impacts that AI model inference creates on different regions. This is fundamentally different from the existing sustainable GLB techniques that have predominantly focused on minimizing the weighted *sum* of energy costs, carbon footprint and/or water footprint [1, 19, 32, 39, 45]. As shown in our experiments (Section 4), minimizing the total environmental footprint does not necessarily treat each individual region fairly and can even potentially exacerbate environmental inequity due to aggressive exploitation of certain regions.

3.2 An Online Algorithm

The addition of two equity-related costs in (4a) explicitly mitigates the greatest long-term environmental costs across all the data centers. Thus, they couple all the GLB decisions over T time slots. Consequently, the optimal GLB decisions require complete offline information (including future workload arrivals and carbon/water efficiencies) in advance, which is lacking in practice. Next, we propose an online algorithm, called eGLB, to solve (4a)–(4d) and optimize equity-aware GLB decisions in an online manner.

A crucial step in eGLB is to construct a new optimization problem that can be solved based on available online information by removing the dependency of the optimization objective on future information. To this end, we first transform the original problem (4a)–(4d) into an equivalent new problem that can be solved using dual mirror descent (DMD). Specifically, for every time step $t \in [1, T]$, we introduce a set of auxiliary variables $\{z_c(t), z_w(t)\}$ and consider the following new transformed problem:

$$\min_{x(t),z_c(t),z_w(t)} \frac{1}{T}\sum_{t=1}^{T} g_t(x(t)) + \frac{\mu_c}{T}\sum_{t=1}^{T}\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,c}(z_{i,c}(t))\right] + \frac{\mu_w}{T}\sum_{t=1}^{T}\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,w}(z_{i,w}(t))\right] \tag{5a}$$

$$s.t., \quad \text{constraints (4b)(4c)(4d)} \tag{5b}$$

$$\frac{1}{T}\sum_{t=1}^{T} z_{i,c}(t) \geq \frac{1}{T}\sum_{t=1}^{T} c_{i,t}(x(t)), \quad \forall i \in \mathcal{N} \tag{5c}$$

$$\frac{1}{T}\sum_{t=1}^{T} z_{i,w}(t) \geq \frac{1}{T}\sum_{t=1}^{T} w_{i,t}\left(x(t)\right), \quad \forall i \in \mathcal{N} \tag{5d}$$

where the auxiliary variables $z_c(t) = (z_{1,c}(t)\cdots, z_{N,c}(t))$ and $z_w(t) = (z_{1,w}(t)\cdots, z_{N,w}(t))$ are chosen from a fixed feasible set $\mathcal{Z}_c$ and $\mathcal{Z}_w$, respectively. Here, we set $\mathcal{Z}_c = \{z_c | 0 \leq z_{i,c} \leq \bar{z}_{i,c}, \forall i = 1, \cdots, N\}$ and $\mathcal{Z}_w = \{z_w | 0 \leq z_{i,w} \leq \bar{z}_{i,w}, \forall i = 1, \cdots, N\}$ to guarantee a feasible solution for any $x_t \in \mathcal{X}_t$. Specifically, we can choose $\bar{z}_{i,c}$ and $\bar{z}_{i,w}$ to be the maximum possible per-time carbon footprint and water footprint in data center i, respectively.

Next, we prove the equivalence of the new transformed problem to the original problem.

Lemma 1. *The transformed problem* (5a)–(5d) *and the original problem* (4a)–(4d) *have the same optimal GLB decisions.*

Proof. To prove this, we first define the optimal GLB decisions as $x^*_{1:T} = (x(1)^*, \cdots, x(T)^*)$ for the original problem (4a)–(4d). Then, we can construct a feasible solution $z_{i,c}(t) = \frac{1}{T}\sum_{t=1}^{T} c_{i,t}(x^*(t))$ and $z_{i,w}(t) = \frac{1}{T}\sum_{t=1}^{T} w_{i,t}\left(x^*(t)\right)$, $\forall t \in [1, T]$ for the transformed problem (5a)–(5d), which results in an equivalent objective function value as (4a) in the original problem. Therefore, the optimal value of the transformed objective in (5a) is less than or equal to that in the original problem.

On the other hand, suppose that there exists another solution, denoted as $\{x'(t), z_c(t)', z_w(t)', t \in [1, T]\}$, which minimizes the transformed problem and makes the transformed objective in (5a) strictly smaller than the original one in (4a). By the convexity assumption of $\mathcal{H}_{i,w}(\cdot)$ and $\mathcal{H}_{i,c}(\cdot)$ and Jensen's inequality, we have

$$\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,c}\left(\frac{1}{T}\sum_{t=1}^{T} z'_{i,c}(t)\right)\right] \leq \frac{1}{T}\sum_{t=1}^{T}\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,c}\left(z'_{i,c}(t)\right)\right], \tag{6}$$

$$\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,w}\left(\frac{1}{T}\sum_{t=1}^{T} z'_{i,w}(t)\right)\right] \leq \frac{1}{T}\sum_{t=1}^{T}\max_{i\in\mathcal{N}}\left[\mathcal{H}_{i,w}\left(z'_{i,w}(t)\right)\right]. \tag{7}$$

Based on the monotonically increasing assumption on $\mathcal{H}_{i,c}(\cdot)$ and $\mathcal{H}_{i,w}(\cdot)$ and by substituting $x'_{1:T} = (x'(t), \cdots, x'(T))$ back to (4a), we see that the objective value in (4a) with $x'(t)$ as the solution is even smaller, which is in contradiction to the assumption that $x^*_{1:T}$ is optimal. Therefore, for the transformed problem, the optimal objective value has to be the same as the original one, and the action $x^*_{1:T} = x'_{1:T}$ is the optimal solution. □

Based on the equivalence of the new transformed problem to the original problem, we now focus on solving the transformed problem (5a)–(5d). The two added constraints (5c) and (5d) still involve all the decisions over T time slots. To remove the temporal coupling, we consider the Lagrangian form of the transformed problem (5a)–(5d). For the convenience of notation, we first define

$$\mathcal{H}_c(z_c(t)) = [\mathcal{H}_{1,c}(z_c(t)), \cdots, \mathcal{H}_{N,c}(z_c(t))], \tag{8}$$

$$\mathcal{H}_w(z_w(t)) = [\mathcal{H}_{1,w}(z_w(t)), \cdots, \mathcal{H}_{N,w}(z_w(t))], \tag{9}$$

$$C(t) = [c_{1,t}(x(t)), \cdots, c_{N,t}(x(t))], \tag{10}$$

$$\mathcal{W}(t) = [w_{1,t}(x(t)), \cdots, w_{N,t}(x(t))]. \tag{11}$$

Then, subject to the constraints (4b)(4c)(4d), we write the Lagrangian as follows:

$$\begin{aligned}&\mathcal{L}(x_{1:T}, z_{c,1:T}, z_{w,1:T}, \kappa)\\ &= \frac{1}{T}\left(\sum_{t=1}^{T} g_t(x(t)) + \mu_c\|\mathcal{H}_c(z_c(t))\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t))\|_\infty\right)\\ &\quad + \langle \kappa, \begin{bmatrix} \frac{1}{T}\cdot\left(\sum_{t=1}^{T} C_t(x(t)) - \sum_{t=1}^{T} z_c(t)\right) \\ \frac{1}{T}\cdot\left(\sum_{t=1}^{T} w_t(x(t)) - \sum_{t=1}^{T} z_w(t)\right)\end{bmatrix}\rangle\end{aligned} \tag{12}$$

where κ is the Lagrangian multipliers associated with the constraints (5c) and (5d), and $\langle a, b\rangle$ denotes the inner product of two vectors a and b.

By solving the problem (12) online subject to the constraints (4b)(4c)(4d), we would obtain the optimal GLB decisions if the optimal Lagrangian multiplier κ were provided. Nonetheless, κ can only be estimated with online information. Based on this insight, we sequentially update κ using dual mirror descent (DMD) [27] based on online information and obtain GLB decisions $x(t)$ for $t = 1, \cdots, T$.

We describe the algorithm in Algorithm 1. More specifically, at time t, we receive the cost functions and optimize the action $x(t)$ and auxiliary variable $z(t) = (z_c(t), z_w(t))$ according to the current estimate of dual variable κ_t. These variables are optimized in Line 4 and Line 5, respectively. The insight is that the estimated dual variable κ_t controls the adjusted penalty for the GLB action $x(t)$ based on how much the cumulative actual carbon and water footprints have deviated from the targets $z(t) = (z_c(t), z_w(t))$.

We update the dual variable κ_t using DMD. More concretely, by taking the subgradient of κ with respect to the Lagrange function and using the online information at time t, we obtain a stochastic gradient estimate of κ_t. In Line 7, the vector d_t is set as the opposite direction to the gradient of κ_t in order to minimize the Lagrange function. Finally, the updated dual variable κ_{t+1} is obtained with Bregman projection using a reference function $h(\cdot)$ which is differentiable and strongly convex. For example, a common choice of the reference function is $h(a) = \frac{1}{2}\|a\|^2$, which results in additive updates of the dual variable estimate κ_t [3].

Algorithm 1: Online GLB for Environmentally Equitable AI (eGLB)

1 **Input:** Initial Lagrange multiplier $\kappa_1 \in \mathbb{R}^{2N}_{\geq 0}$, reference function $h(\cdot) : \mathbb{R}^{2N} \to \mathbb{R}$, total length of horizon T and learning rate η

2 **for** $t = 1, \ldots, T$ **do**

3 Receive the cost function of energy, carbon and water as $g_t(\cdot)$, $c_t(\cdot)$ and $w_t(\cdot)$, the action constraint $\mathcal{X}_t = \{x | x \text{ satisfies (4b)(4c)(4d)}\}$.

4 Make the primal decision

$$x(t) = \arg\min_{x(t)\in\mathcal{X}_t} \{g_t(x(t)) + \kappa_t^\top \cdot \begin{bmatrix} C_t(x(t)) \\ \mathcal{W}_t(x(t))\end{bmatrix}\},$$

5 Determine the auxiliary variable:

$$\begin{aligned}\{z_c(t), z_w(t)\} = \arg\min_{z_c\in\mathcal{Z}_c, z_w\in\mathcal{Z}_w} &\{\mu_c\|\mathcal{H}_c(z_c)\|_\infty\\ &+ \mu_w\|\mathcal{H}_w(z_w)\|_\infty - \kappa_t^\top \begin{bmatrix} z_c \\ z_w\end{bmatrix}\},\end{aligned}$$

6 Obtain a stochastic subgradient of κ_t:

$$d_t = \begin{bmatrix} z_c(t) \\ z_w(t)\end{bmatrix} - \begin{bmatrix} C_t(x(t)) \\ \mathcal{W}_t(x(t))\end{bmatrix}.$$

7 Update the dual variable by mirror descent:

$$\kappa_{t+1} = \arg\min_{\kappa\in\mathcal{R}^{2N}_{\geq 0}} \langle d_t, \kappa\rangle + \frac{1}{\eta} V_h(\kappa, \kappa_t),$$

where $V_h(x, y) = h(x) - h(y) - \nabla h(y)^\top (x - y)$ is the Bregman divergence.

Next, we analyze eGLB in terms of the cost objective in (4a).

Theorem 1. *By initializing $\kappa_1 \in \mathbb{R}^{2N}_{\geq 0}$ as a zero vector, considering the reference function $h(a) = \frac{1}{2}\|a\|^2$, and denoting the GLB actions as $x_{1:T} = (x(1), \cdots, x(T))$ and the overall cost defined in* (4a) *as $cost(x_{1:T})$, we have the following:*

$$cost(x_{1:T}) \leq cost(x^*_{1:T}) + \eta BT + C\sqrt{\frac{2}{T}(B + \frac{M}{\eta}D)} \tag{13}$$

where $cost(x^*_{1:T})$ is the minimum cost given by optimal offline algorithm, $\eta > 0$ is the learning rate, c_m and w_m are the maximum possible gradients of carbon and water footprints in (2) *and* (3)*, $M = \max_{i\in\mathcal{N}} M_i$ is the maximum processing capacity of all data centers, θ_m is the maximum gradient of $H_{i,c}(\cdot)$ and $H_{i,w}(\cdot)$, $B = \frac{N}{2}\left[\max_{i\in\mathcal{N}} \bar{z}^2_{i,c}\right] + \frac{N}{2}\left[\max_{i\in\mathcal{N}} \bar{z}^2_{i,w}\right]$ (in which $\bar{z}_{i,c}$ and $\bar{z}_{i,w}$ are the maximum possible per-time carbon and water footprints in data center i, respectively), $C = \theta_m(\mu_c + \mu_w)$ and $D = \theta_m(\mu_c c_m + \mu_w w_m)$, respectively. Moreover, by setting the learning rate $\eta = O(1/T)$, we have*

$$cost(x_{1:T}) \leq cost(x^*_{1:T}) + O(1). \tag{14}$$

Theorem 1 bounds the gap between eGLB and the optimal offline algorithm in terms of the overall cost defined in (4a). The constants B and D are problem-specific and naturally increase as the input range is larger. In addition, the gap depends on the choice of the learning rate η. Specifically, by increasing η, eGLB updates the dual

variable κ by more aggressively following the stochastic gradient (Line 6 in Algorithm 1). This can introduce greater drifts due to the "forgetting" of the past time slots and hence increases the term ηBT. On the other hand, a larger η can reduce the time steps needed for updating the dual variable to track the optimal dual variable, and hence reduce the term $D\sqrt{\frac{2}{T}(B+\frac{M}{\eta}D)}$. Thus, by setting the learning rate $\eta = O(1/T)$ to balance the two terms, we can have an $O(1)$ cost gap. Note that, without further stochastic assumptions (e.g., all the inputs follow an independent and identical distribution), eliminating the $O(1)$ cost gap between eGLB and the optimal offline algorithm remains an open challenge in the literature [3, 56]. For example, in a relevant context of online budget allocation, having a zero cost gap is impossible in general adversarial settings that we consider [3]. Importantly, as is shown in our experimental results (Section 4.2), eGLB demonstrates a strong empirical performance even compared to the optimal offline algorithm.

4 EXPERIMENTS

In this section, we report on experiments of different GLB algorithms using trace-based simulations. Our results demonstrate that eGLB has a great potential to effectively address AI's environmental inequity that would otherwise be potentially amplified by other GLB algorithms. Importantly, the empirical cost performance of eGLB is close to the optimal offline equity-aware GLB, complementing our theoretical analysis of eGLB in Theorem 1.

4.1 Methodology

As detailed information about AI system and workload settings is typically proprietary, we run simulations by scaling up workload traces collected from public sources and considering synthetic data center settings that approximate realistic scenarios. This is in line with the prior GLB literature [19, 32, 45, 67]. Next, we describe the default setup of our experiments, which will later be varied for sensitivity studies.

4.1.1 Workload Trace. To obtain the workload trace, we extract the GPU power usage data from [46] for the server cluster hosting the large language model BLOOM over an 18-day period (between September 23 and October 11 in 2022). Because there is only a single workload trace provided for BLOOM in [46], we follow the data augmentation method in [12] and distribute the workload trace to the 10 gateways (plus a small perturbation to account for different time zones). As in [46], we directly quantify the amount of workload using power demand. We also scale up the workload trace to let the maximum workload match our data center power capacity as introduced below. The 18-day workload trace will be also be extended using data augmentation techniques to evaluate different GLB algorithms over a longer timescale (Section 4.2.4).

4.1.2 Data Centers. We consider a set of 10 geo-distributed data centers, including four in the U.S. (Virginia, Georgia, Texas, and Nevada), four in Europe (Belgium, the Netherlands, Germany, and Denmark), and two in Asia (Singapore and Japan). These locations are all a large presence of data centers, including Google's data centers [59]. The details of data center locations are available in the appendix.

Assuming that there are 10 gateways corresponding to the 10 data center locations, we consider two scenarios: (1) **full GLB flexibility**: the workloads can be flexibly dispatched from any gateway to any data center; and (2) **partial GLB flexibility**: the workloads arriving at a gateway can only be dispatched to a certain subset of data centers. As shown in recent studies [12], even cross-continent AI workload placement only marginally increases the end-to-end latency without degrading service quality. Thus, the "full GLB flexibility" scenario is already feasible in practice, whereas the "partial GLB flexibility" scenario accounts for various other constraints such as strict latency and bandwidth.

For processing AI inference workloads, we assume that each data center houses a cluster of 500 homogeneous servers. Each server is equipped with four NVIDIA A100 GPUs and has a maximum total power of 2 kW. Thus, excluding the network switches and servers for other services beyond the scope of our study, each data center has a maximum server power of 1 MW for AI inference.

We set the data center PUE as 1.1, which is consistent with the state-of-the-art PUE value with efficient operation [20, 59]. For simplicity, we use the actual carbon footprint and water footprint to measure the regional environmental impact (i.e., $\mathcal{H}_{i,c}(x) = x$ and $\mathcal{H}_{i,w}(x) = x$ in (4a)).

4.1.3 Energy Price, Carbon Intensity, and WUE. We collect hourly energy prices for the 10 data centers over the same 18-day period as our workload trace. Specifically, for each data center in Europe and Asia, we collect the hourly country-level energy prices from [30]. For the U.S. data centers, we collect the hourly energy prices from their respective ISOs [84].

For each of the U.S. data centers, we collect the state-level hourly energy fuel mix data [84] and calculate the indirect WUE based on the fuel mix by following [19] and [32], respectively. The carbon intensity and energy water intensity factor (EWIF) for each fuel mix are chosen based on [19] and [43]. We do not have free access to the hourly energy fuel mix data for our data center locations in Europe and Asia [30]. Thus, we generate synthetic hourly fuel mixes for these locations based on the U.S. data. Besides, the hourly carbon intensity of each datacenter is obtained from [48], where the US locations are ISO level and the Europe and Asia locations are country-level carbon intensity. The details are available in the appendix.

To model the on-site WUE, we assume that the data centers use cooling towers for heat rejection, which are common in the industry (even in water-stressed regions like Arizona [37]). We collect the hourly weather data from [31] for the airports closest to each of our data center locations, and then obtain the wet bulb temperature from the dry bulb temperature and relative humidity based on [52]. Next, we calculate the on-site WUE using the empirical formula in terms of the wet-bulb temperature presented in [32]. While assuming cooling towers for rejecting heat into the outside environment, our study can be easily adapted to air-side economizers, which use water for humidity control or when the outside dry bulb temperature is high [51].

4.1.4 Offline and Online Optimization. Assuming complete knowledge of future information, we first use offline optimization in order to quantify the maximal potential of equity-aware GLB to address

Table 1: Comparison of different GLB algorithms. The metric ratio is the maximum water or carbon footprint divided by the average. The results of eGLB with the learning rate $\eta = 1.7 \times 10^{-4}$ are bolded.

GLB Flexibility	Metric		Algorithm								
			GLB-Energy	GLB-Carbon	GLB-Water	GLB-C2	GLB-All	GLB-Nearest	eGLB-Off	eGLB-MPC	**eGLB**
Full	**Energy** (US$)	avg	29170	47708	56184	33735	32466	47038	36106	36199	**37643**
	Water (m^3)	avg	1525.1	1396.2	1243.9	1486.3	1426.7	1446.7	1431.8	1444.4	**1448.7**
		max	2607.5	2671.6	2010.4	2675.7	2358.0	2090.9	1705.2	1898.3	**1928.0**
		max/avg	1.71	1.91	1.62	1.80	1.65	1.45	1.19	1.31	**1.33**
	Carbon (ton)	avg	118.17	90.50	103.17	100.75	105.83	111.80	102.59	105.06	**105.70**
		max	205.10	166.79	224.49	166.82	171.50	143.46	128.79	134.68	**139.39**
		max/avg	1.74	1.84	2.18	1.66	1.62	1.28	1.26	1.28	**1.32**
Partial	**Energy** (US$)	avg	29659	47694	53976	33729	32822	47038	36013	37210	**37768**
	Water (m^3)	avg	1524.1	1415.5	1249.9	1490.4	1420.1	1446.7	1440.6	1446.6	**1450.1**
		max	2616.1	2700.3	2028.4	2668.8	2344.7	2090.9	1777.3	1891.4	**1929.0**
		max/avg	1.72	1.91	1.62	1.79	1.65	1.45	1.23	1.31	**1.33**
	Carbon (ton)	avg	117.52	92.22	106.07	102.13	106.53	111.80	103.31	105.11	**105.65**
		max	205.77	168.42	224.49	166.44	177.43	143.46	133.45	135.42	**139.89**
		max/avg	1.75	1.83	2.12	1.63	1.67	1.28	1.29	1.29	**1.32**

AI's environmental inequity. We then use our online algorithm eGLB to demonstrate how much of that potential can be realized.

In the offline case, we consider hourly GLB decisions and use *cvxpy* to solve (4a)–(4d) offline based on the complete information about all the future workload arrivals, energy prices, carbon intensity, and WUE values. We refer to this offline algorithm as **eGLB-Off**. It takes about 3 minutes on a desktop with Intel i7-9700K CPU and 16GB RAM to solve the problem for an 18-day simulation in our experiments. The weight hyperparameters in (4a) are set as $\mu_c = 1500$ \$/ton and $\mu_w = 60$ \$/m^3. Note that these hyperparameters are only used to adjust the relative importance of different cost terms in the optimization process and do not reflect the true monetary costs of carbon or water footprints.

In the online case, we use eGLB to optimize the GLB decisions according to the sequentially revealed workload arrivals, energy price, carbon intensity, and water efficiency information. It takes around 30 seconds on the same machine to calculate GLB decisions for the 18-day simulation. Besides, we compare eGLB with another online policy, eGLB-MPC, which leverages model predictive control (MPC). Specifically, eGLB-MPC optimizes the objective in Eqn (4a) over a receding 24-hour horizon, utilizing predictions of future workloads, energy prices, carbon intensities, and water efficiencies. For a fair comparison, the hyperparameters μ_c and μ_w by eGLB and eGLB-MPC are chosen as the same values as the offline optimizer eGLB-Off.

4.1.5 Metrics. We evaluate our equity-aware GLB algorithms using the following metrics: *average energy cost*, the total energy cost throughout the 18-day period divided by 10 data center locations; *average carbon/water footprint*, the total carbon/water footprint throughout the 18-day period divided by 10 data center locations; and the *maximum regional carbon/water footprint* over the 18-day period among the 10 data center locations. If scaled up by a factor of 10, the average value is equivalent to the total value. We also include the maximum regional carbon/water footprint to the average value to reflect the level of environmental equity, i.e., the smaller max/avg, the more equitable, and $max/avg = 1$ means all the regions have the same environmental cost in terms of the carbon/water footprint.

4.1.6 Baseline Algorithms. We consider the following GLB-related algorithms for comparison.

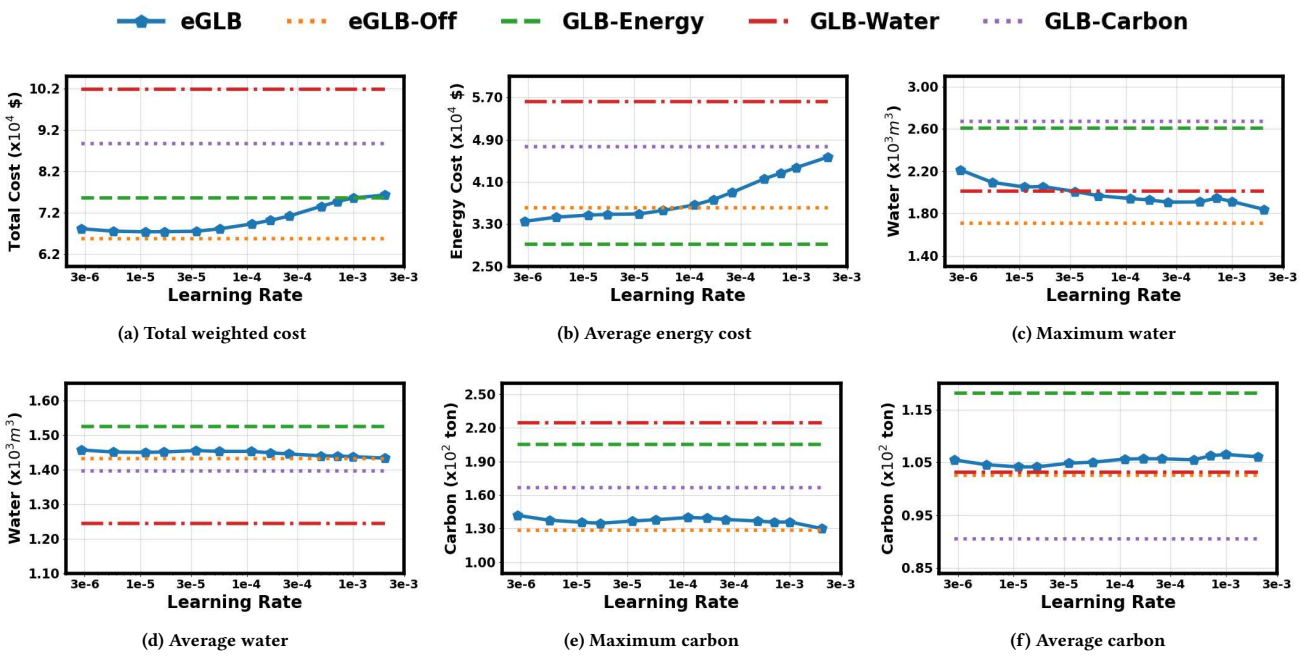
- GLB-Energy: This algorithm is based on [45, 63, 67] and only minimizes the total energy cost. It is a special case of eGLB by setting $\mu_c = 0$ and $\mu_w = 0$ in (4a).
- GLB-Carbon: Minimization of the total carbon footprint.
- GLB-Water: Minimization of the total water footprint.
- GLB-C2: This algorithm is based on [19] and minimizes the weighted sum of the total energy cost and carbon footprint.
- GLB-All: This algorithm is based on [32] and minimizes the weighted sum of the total energy cost, carbon footprint, and water footprint.
- GLB-Nearest: This algorithm is a special case of GLB and directly routes workloads from each gateway to its nearest data center. It is commonly used in practice as a default baseline algorithm [19, 63].

Without considering equity-related costs, the GLB decisions in these algorithms are not coupled over time and hence can be optimally obtained online. The weights for carbon and water (if applicable) in GLB-C2 and GLB-All are set such that their respective total carbon and water footprints are smaller than those of eGLB-Off.

4.2 Results

We show our results in Table 1 by considering two different scenarios: GLB with full or partial flexibility. Our results highlight that eGLB can improve AI's environmental equity by reducing the environmental impact on the most disadvantaged region while still keeping the average environmental footprint and energy cost close to or even lower than those of alternative GLB algorithms. In addition, the empirical performance of eGLB is close to the optimal offline algorithm eGLB-Off. Next, we discuss our results in detail.

4.2.1 GLB with Full Flexibility. We first consider the full-flexibility scenario in which the workloads can be dispatched to any data center. Among all the algorithms, eGLB-Off has the lowest carbon and water footprints for the most disadvantaged regions with complete information of workload, carbon and water footprints. Meanwhile, the average energy cost, carbon footprint, and water footprint of eGLB-Off are comparable to or even lower than the other GLB algorithms. Thus, eGLB-Off has almost the lowest "*maximum to*



(a) Total weighted cost (b) Average energy cost (c) Maximum water

(d) Average water (e) Maximum carbon (f) Average carbon

Figure 1: The energy cost, carbon and water footprint of eGLB with different learning rates η under full GLB flexibility. The results for eGLB-Off, GLB-Energy, GLB-Carbon, and GLB-Water are shown for comparison.

average" ratio in terms of both the carbon footprint and water footprint, effectively reducing the regional disparity and improving environmental equity.

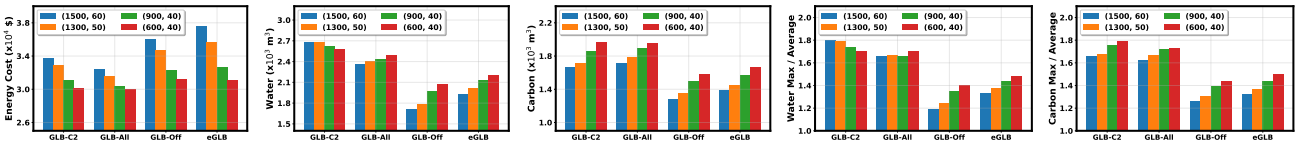
Interestingly, while GLB-Energy, GLB-Carbon, and GLB-Water can minimize the total energy cost, carbon footprint, and water footprint, respectively, they amplify the environmental inequity compared to GLB-Nearest. This is due to the inequity *unawareness* of these algorithms — their aggressive exploitation of certain regions may come at the cost of harming these regions in terms of environmental impacts. For example, GLB-Energy exploits the cheaper energy price of Texas by assigning more workloads to this region, but this can result in a disproportionately high environmental footprint in Texas due to its worse carbon intensity and/or WUE than some other regions. While GLB-C2 and GLB-All can balance the energy cost and environmental footprints in terms of the average/total metric, they can still result in disproportionately high environmental burdens on the already-disadvantaged regions due to the unawareness of equity. This is similar to algorithmic unfairness against disadvantaged individuals or user groups caused by an AI model that purely minimizes the average loss [14, 49].

While the prior studies [19, 32] have demonstrated that the total carbon footprint and water footprint are often in tension with the energy cost, our results further add that environmental equity may not be cost-free either. Nonetheless, by balancing the energy cost and environmental equity as formulated in (4a), the price we pay for equity can be reasonably low.

eGLB vs. eGLB-Off. In comparison to the offline optimizer, eGLB only has access to online causal information. This naturally leads to worse performance than if full information were available; however, the carbon and water footprints for the most disadvantaged regions in eGLB are still better than all other GLB algorithms. In Table 1, the maximum/average carbon and water footprints of eGLB are mostly within 10% of the offline optimal solutions from eGLB-Off, which demonstrates the empirical effectiveness of eGLB and complements our theoretical analysis in Theorem 1.

eGLB vs. eGLB-MPC. Unlike the offline optimal policy eGLB-Off, eGLB-MPC is limited to future information within the next 24-hour prediction window. This limitation reflects real-world constraints, where perfect foresight of the entire future is unavailable. As expected, eGLB-MPC outperforms eGLB (with sequentially revealed information) in terms of the average energy cost, water footprint, and carbon footprint. Interestingly, the maximum-to-average ratios for carbon and water footprints are similar between eGLB-MPC and eGLB, despite eGLB-MPC's 24-hour prediction window. This observation emphasizes the fundamental challenge of achieving long-term environmental equity when faced with limited predictions of the future.

As shown in Theorem 1, η is an important parameter that determines the cost gap between eGLB and its offline version eGLB-Off. Thus, we also evaluate the impact of the learning rate η in Fig. 1. The total cost is calculated by summing up the energy cost and equity-related carbon/water costs (weighted by μ_c and μ_w, respectively). Then, in line with Section 4.1.5, we divide the total cost by 10 data center locations and show it in Fig. 1a. As we increase the learning rate η, the total cost first decreases and then increases as suggested by Theorem 1. Empirically, the optimal learning rate η is around 3×10^{-5} in our setting. In Fig. 1b, the average energy cost increases as we increase η, since larger η leads to more aggressive



(a) Average energy cost (b) Maximum water (c) Maximum carbon (d) Max/avg for water (e) Max/avg for carbon

Figure 2: The energy cost, carbon footprint, and water footprint of eGLB with different (μ_w, μ_c) shown in the legend under full GLB flexibility. The results for eGLB-Off, GLB-Energy, GLB-Carbon, and GLB-Water are also shown for comparison.

Table 2: Comparison of different GLB algorithms. The default workload trace is augmented to 180 days to evaluate the long-term impact of different GLB algorithms. The results of eGLB with the learning rate $\eta = 1.7 \times 10^{-4}$ are bolded.

GLB Flexibility	Metric		Algorithm							
			GLB-Energy	GLB-Carbon	GLB-Water	GLB-C2	GLB-All	GLB-Nearest	eGLB-Off	**eGLB**
Full	**Energy** (US$)	avg	279620	454608	539847	326104	312372	450992	341998	**359433**
	Water (m^3)	avg	14329.6	12992.8	11694.2	13822.4	13338.4	13584.9	13439.3	**13591.5**
		max	23753.4	24779.5	19478.0	25154.2	21307.6	19662.3	16339.6	**18199.0**
		max/avg	1.66	1.91	1.67	1.82	1.60	1.45	1.22	**1.34**
	Carbon (ton)	avg	1098.29	830.66	947.89	925.28	975.76	1035.97	951.91	**977.92**
		max	1868.37	1544.89	2110.61	1566.99	1656.06	1342.44	1202.91	**1294.23**
		max/avg	1.70	1.86	2.23	1.69	1.70	1.30	1.26	**1.32**
Partial	**Energy** (US$)	avg	283551	456028	516966	324368	314254	450992	342498	**359510**
	Water (m^3)	avg	14312.2	13249.7	11755.2	13903.6	13298.1	13584.9	13508.3	**13619.4**
		max	23961.6	25347.0	18852.3	24833.9	21734.6	19662.3	16824.7	**18267.1**
		max/avg	1.67	1.91	1.60	1.79	1.63	1.45	1.25	**1.34**
	Carbon (ton)	avg	1093.61	850.24	981.20	942.91	986.54	1035.97	964.03	**980.46**
		max	1874.35	1577.51	2110.77	1545.47	1695.69	1342.44	1240.80	**1301.70**
		max/avg	1.71	1.86	2.15	1.64	1.72	1.30	1.29	**1.33**

updates of the Lagrange multiplier κ. The Lagrange multiplier κ is used to penalize the cost objective according to the environmental footprint, which means larger κ shifts the objective function more towards environmental equity, as compared with purely minimizing the energy cost. Similarly, as we increase the learning rate η, the carbon and water footprints for the most disadvantaged regions decrease, as shown in Fig 1c and 1e. The underlying reason is similar — a larger learning rate η updates κ more rapidly, eventually leading to more attention to equity-related costs. Interestingly, the average carbon and water footprints of eGLB are very close to the offline version, eGLB-Off. Like in the task of machine learning training, the learning rate hyperparemeter can be tuned based on a validation dataset to get the desired performance in practice.

4.2.2 GLB with Partial Flexibility. Now, we consider the partial-flexibility scenario in which intra-continental workload routing is fully flexible but inter-continental workload routing is partially restricted. Specifically, we only allow partial inter-continental workload routing as follows: workloads can be flexibly routed between Asia and the western U.S. (Nevada), and between Europe and the eastern U.S. (Virginia and Georgia).

Our results are similar to those in the full-flexibility scenario. Specifically, while the inter-continental workload routing restriction limits the GLB decision space, eGLB-Off still has the lowest carbon and water footprints for the most disadvantaged regions. Meanwhile, the average energy cost, carbon footprint, and water footprint of eGLB-Off are all comparable to or even lower than the other GLB algorithms. Thus, even without full flexibility, eGLB-Off demonstrates a great potential to address AI's environmental inequity in today's geographically distributed data center infrastructures. Additionally, as shown in Table 1, the performance of eGLB is very close to its offline counterpart, eGLB-Off.

GLB-Nearest does not route workloads across data centers and hence is not affected by the partial GLB flexibility. Interestingly, the result of GLB-Carbon is not affected by the inter-continental workload routing restriction in our setting, because the workloads from each continent can be processed in at least one low-carbon data center in our setup (see Table 3).

With partial GLB flexibility, the impact of the learning rate κ is similar as that with full GLB flexibility. More details about the empirical results can be found in Appendix B.

4.2.3 GLB with Different μ_c and μ_w. Adjusting the weight hyperparameters μ_c and μ_w allows us to control the relative importance of the energy cost and environmental equity. Here, by using the default setup, we evaluate how the weights for carbon and water footprints impact the performance of different GLB algorithms. We show the results in in Fig. 2. We only compare the performance of GLB-C2, GLB-All, eGLB-Off and eGLB, as the other GLB algorithms are not affected by these weight hyperparameters. Naturally, by assigning lower weights to carbon and/or footprints, the emphasis on reducing the environmental inequity in terms of these footprints is lessened, allowing all the four GLB algorithms to have a reduced energy cost. However, this also results in a higher maximum carbon and/or water footprint (as well as higher maximum-to-average ratios) for these algorithms, with the only exception being GLB-C2. More specifically, unlike the other algorithms, GLB-C2 minimizes

the weighted sum of the energy cost and carbon footprint without accounting for the impact of water footprint. As a result, it has a higher (maximum) water footprint as we increase the weight of carbon for reducing the carbon footprint. This empirical finding suggests that the goals of reducing carbon emission and water usage may not be aligned, or even in opposition, necessitating a joint optimization of their combined weighted sum.

While eGLB applies with any $\mu_c \geq 0$ and $\mu_w \geq 0$, it is up to the AI system operator to tune weight hyperparameters (e.g., based on validation dataset) to achieve a desired outcome. This is also a common and standard practice in real systems (see Google's dynamic capacity planning to balance the energy cost and environmental impacts [64]).

4.2.4 Evaluation over a Longer Timescale. The open-source BLOOM inference trace is only for 18 days [46] and used in our default setup. Due to limited availability of public data, we extend the 18-day BLOOM inference trace to 180 days by using data augmentation techniques to evaluate the impacts of eGLB in terms of environmental equity over a longer timescale. More specifically, we add 25% random perturbations and append the perturbed workload trace to the original one to construct a 180-day trace. The results are shown in Table 2, offering similar insights as in the default case for both full and partial GLB scenarios. We can see that compared to the other equity-oblivious GLB algorithms, eGLB can effectively reduce the environmental inequity among different regions in terms of the maximum-to-average ratio for both carbon and water footprints. Even compared to eGLB-Off, eGLB delivers a similar performance in terms of environmental equity while only marginally increasing the total energy cost, which demonstrates the potential of eGLB to address AI's emerging environmental inequity in practice without knowing all the future information.

5 RELATED WORK

Our work is the first to address the critical concern of AI's emerging environmental inequity by leveraging GLB, and contributes to the GLB literature for cloud computing and data centers [1, 4, 8, 10, 19, 24, 28, 32, 40, 41, 45, 63, 64, 67, 69]. Specifically, prior studies focus on reducing the total energy cost, carbon footprint, water footprint, or a weighted combination of these metrics; ignoring the potential for regional disparities. We show in this paper that existing GLB algorithms can potentially amplify environmental inequity by further exploiting already vulnerable regions. For example, GLB algorithms that aggressively exploit lower electricity prices [63, 67] and/or more renewables [19, 45] may schedule more workloads to data centers (located in, for example, Arizona) that are extremely water-stressed; thus adding a disproportionately high pressure to local water systems.

Sustainable AI has received a significant amount of attention in recent years [9, 29, 59, 60, 71, 75, 79]. To make AI more energy-efficient and sustainable, a variety of approaches have been explored and studied, including computationally efficient training and inference [11, 65], energy-efficient GPU and accelerator designs [22, 59, 87], carbon-aware task scheduling [29, 86], green cloud infrastructures [1, 4, 18, 42], among others. While they are useful for overall sustainability, these studies do not address the emerging environmental equity among different regions for deploying AI services. Additionally, they have mostly focused on carbon footprint, neglecting other crucial environmental footprints, e.g., water footprint [20, 36, 43, 50]. In contrast, we holistically consider both carbon and water footprints and make novel contributions to sustainable AI from the perspective of environmental equity.

There also exist non-*computational* approaches to improving AI's environmental sustainability. For example, data center operators have increasingly adopted carbon-free energy such as solar and wind power to lower AI's carbon emissions [20, 36, 50, 86]. To cut on-site potable water consumption and mitigate the stress on already-limited freshwater resources, climate-conscious cooling system designs (e.g., air-side economizers and purifying non-potable water) have recently seen an uptick in the data center industry [21, 51]. These non-computational approaches alone are typically not the most effective solution to sustainable AI, and must be designed in conjunction with computational approaches (e.g., workload scheduling) [1, 2, 54, 64]. As such, our study of equity-aware GLB can inform the planning of on-site carbon-free energy and cooling system renovation projects to better achieve social and environmental justice.

Equity and fairness are crucial considerations for AI. The existing research in this space has predominantly focused on mitigating prediction unfairness against disadvantaged individuals and/or groups under a variety of settings [7, 14, 15, 17, 38, 44, 49, 61, 62, 70, 88, 89]. Our work on environmental equity adds a unique dimension of fairness and greatly complements the existing rich body of research, collaboratively and holistically building equitable and socially-responsible AI.

6 CONCLUDING REMARKS

In this paper, we take a first step to address the emerging environmental inequity of AI by balancing its regional negative environmental impact in an equitable manner. Concretely, we focus on the carbon and water footprints of AI model inference and propose equity-aware GLB to explicitly address the environmental impact on the most disadvantaged region. eGLB can optimize GLB decisions to fairly balance AI's environmental cost across different regions in an online manner. We run trace-based simulations by considering a set of 10 geographically distributed data centers that serve inference requests for a large language AI model. The results highlight that, compared to the existing GLB approaches, our proposed equity-aware GLB can significantly reduce the regional disparity in terms of AI's carbon and water footprints.

Our work demonstrates the need and great potential of equity-aware GLB to address AI's emerging environmental equity. It opens up multiple new research directions to further improve AI's environmental equity, such as how to jointly optimize GLB and non-IT resource (e.g., batteries) management and how to leverage environmental science tools to quantify the impact of AI's carbon and water footprints on each region's ecological system.

ACKNOWLEDGMENTS

Pengfei Li, Jianyi Yang, and Shaolei Ren were supported in part by the NSF under grants CNS-1910208 and CCF-2324941. Adam

Wierman was supported in part by the NSF under grants CNS-2146814, CPS-2136197, CNS-2106403 and NGSDI-2105648, and by the Resnick Sustainability Institute.

REFERENCES

[1] Bilge Acun, Benjamin Lee, Fiodar Kazhamiaka, Kiwan Maeng, Udit Gupta, Manoj Chakkaravarthy, David Brooks, and Carole-Jean Wu. 2023. Carbon Explorer: A Holistic Framework for Designing Carbon Aware Datacenters. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 118–132. https://doi.org/10.1145/3575693.3575754

[2] Thomas Anderson, Adam Belay, Mosharaf Chowdhury, Asaf Cidon, and Irene Zhang. 2022. Treehouse: A Case For Carbon-Aware Datacenter Software. In *HotCarbon*.

[3] Santiago R. Balseiro, Haihao Lu, and Vahab Mirrokni. 2022. The Best of Many Worlds: Dual Mirror Descent for Online Allocation Problems. *Operations Research* 0, 0 (May 2022), null. https://doi.org/10.1287/opre.2021.2242

[4] Noman Bashir, Tian Guo, Mohammad Hajiesmaili, David Irwin, Prashant Shenoy, Ramesh Sitaraman, Abel Souza, and Adam Wierman. 2021. Enabling Sustainable Clouds: The Case for Virtualizing the Energy System. In *Proceedings of the ACM Symposium on Cloud Computing* (Seattle, WA, USA) *(SoCC '21)*. Association for Computing Machinery, New York, NY, USA, 350–358. https://doi.org/10.1145/3472883.3487009

[5] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (Virtual Event, Canada) *(FAccT '21)*. Association for Computing Machinery, New York, NY, USA, 610–623. https://doi.org/10.1145/3442188.3445922

[6] Rachel Bergmann and Sonja Solomun. 2021. From Tech to Justice: A Call for Environmental Justice in AI. *AI Now Institute* (October 2021).

[7] Reuben Binns. 2020. On the Apparent Conflict between Individual and Group Fairness. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain) *(FAT* '20)*. Association for Computing Machinery, New York, NY, USA, 514–524. https://doi.org/10.1145/3351095.3372864

[8] Marco Brocanelli, Sen Li, Xiaorui Wang, and Wei Zhang. 2014. Maximizing the Revenues of Data Centers in Regulation Market by Coordinating with Electric Vehicles. *to appear in Sustainable Computing: Informatics and Systems* (2014).

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[10] Jose Camacho, Ying Zhang, Minghua Chen, and Dah Ming Chiu. 2014. Balance Your Bids Before Your Bits: The Economics of Geographic Load-balancing. In *e-Energy*.

[11] Lingjiao Chen, Matei Zaharia, and James Zou. 2023. FrugalGPT: How to Use Large Language Models While Reducing Cost and Improving Performance. arXiv:2305.05176 [cs.LG]

[12] Andrew A Chien, Liuzixuan Lin, Hai Nguyen, Varsha Rao, Tristan Sharma, and Rajini Wijayawardana. 2023. Reducing the Carbon Impact of Generative AI Inference (Today and in 2035). In *Proceedings of the 2nd Workshop on Sustainable Computer Systems* (Boston, MA, USA) *(HotCarbon '23)*. Association for Computing Machinery, New York, NY, USA, Article 11, 7 pages. https://doi.org/10.1145/3604930.3605705

[13] José-Luis Cruz and Esteban Rossi-Hansberg. 2022. *Local Carbon Policy*. Working Paper 30027. National Bureau of Economic Research. https://doi.org/10.3386/w30027

[14] Emily Diana, Wesley Gill, Michael Kearns, Krishnaram Kenthapadi, and Aaron Roth. 2021. Minimax Group Fairness: Algorithms and Experiments. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society* (Virtual Event, USA) *(AIES '21)*. Association for Computing Machinery, New York, NY, USA, 66–76. https://doi.org/10.1145/3461702.3462523

[15] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through Awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference* (Cambridge, Massachusetts) *(ITCS '12)*. Association for Computing Machinery, New York, NY, USA, 214–226. https://doi.org/10.1145/2090236.2090255

[16] Kai Fang, Reinout Heijungs, and Geert R. de Snoo. 2014. Theoretical Exploration for the Combination of the Ecological, Energy, Carbon, and Water Footprints: Overview of a Footprint Family. *Ecological Indicators* 36 (2014), 508–518. https://doi.org/10.1016/j.ecolind.2013.08.017

[17] Batya Friedman and Helen Nissenbaum. 1996. Bias in Computer Systems. *ACM Trans. Inf. Syst.* 14, 3 (jul 1996), 330–347. https://doi.org/10.1145/230538.230561

[18] Anshul Gandhi, Kanad Ghose, Kartik Gopalan, Syed Rafiul Hussain, Dongyoon Lee, Yu David Liu, Zhenhua Liu, Patrick McDaniel, Shuai Mu, and Erez Zadok. 2022. Metrics for Sustainability in Data Centers. In *HotCarbon*.

[19] Peter Xiang Gao, Andrew R. Curtis, Bernard Wong, and Srinivasan Keshav. 2012. It's not easy being green. *SIGCOMM Comput. Commun. Rev.* (2012).

[20] Google. 2022. Environmental Report. https://sustainability.google/reports/

[21] Google. 2023. Water Commitments. https://sustainability.google/commitments/water/

[22] Suyog Gupta and Berkin Akin. 2020. Accelerator-aware Neural Network Design using AutoML. *arXiv preprint arXiv:2003.02838* (2020).

[23] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S. Lee, David Brooks, and Carole-Jean Wu. 2022. ACT: Designing Sustainable Computer Systems with an Architectural Carbon Modeling Tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture* (New York, New York) *(ISCA '22)*. Association for Computing Machinery, New York, NY, USA, 784–799. https://doi.org/10.1145/3470496.3527408

[24] Udit Gupta, Young Geun Kim, Sylvia Lee, Jordan Tse, Hsien-Hsin S. Lee, Gu-Yeon Wei, David Brooks, and Carole-Jean Wu. 2022. Chasing Carbon: The Elusive Environmental Footprint of Computing. *IEEE Micro* 42, 4 (jul 2022), 37–47. https://doi.org/10.1109/MM.2022.3163226

[25] Philipp Hacker. 2023. Sustainable AI Regulation. In *Privacy Law Scholars Conference*. https://arxiv.org/abs/2306.00292

[26] Song Han, Huizi Mao, and William J. Dally. 2016. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. In *ICLR*.

[27] Elad Hazan et al. 2016. Introduction to online convex optimization. *Foundations and Trends® in Optimization* 2, 3-4 (2016), 157–325.

[28] Xin He, Prashant Shenoy, Ramesh Sitaraman, and David Irwin. 2015. Cutting the Cost of Hosting Online Services Using Cloud Spot Markets. In *HPDC*.

[29] Peter Henderson, Jieru Hu, Joshua Romoff, Emma Brunskill, Dan Jurafsky, and Joelle Pineau. 2020. Towards the Systematic Reporting of the Energy and Carbon Footprints of Machine Learning. *J. Mach. Learn. Res.* 21, 1, Article 248 (jan 2020), 43 pages.

[30] International Energy Agency (IEA). 2024. Data and Statistics. https://www.iea.org/data-and-statistics

[31] Iowa State University. 2024. Iowa Environmental Mesonet. https://mesonet.agron.iastate.edu/

[32] Mohammad A. Islam, Kishwar Ahmed, Hong Xu, Nguyen H. Tran, Gang Quan, and Shaolei Ren. 2018. Exploiting Spatio-Temporal Diversity for Water Saving in Geo-Distributed Data Centers. *IEEE Transactions on Cloud Computing* 6, 3 (2018), 734–746. https://doi.org/10.1109/TCC.2016.2535201

[33] Mohammad. A. Islam, Shaolei Ren, Gang Quan, Muhammad Z. Shakir, and Athanasios V. Vasilakos. 2015. Water-Constrained Geographic Load Balancing in Data Centers. *IEEE Trans. Cloud Computing* (2015).

[34] Mark Z. Jacobson. 2010. Enhancement of Local Air Pollution by Urban CO2 Domes. *Environmental Science & Technology* 44, 7 (2010), 2497–2502. https://doi.org/10.1021/es903018m PMID: 20218542.

[35] Amba Kak and Sarah Myers West. 2023. AI Now 2023 Landscape: Confronting Tech Power. *AI Now Institute* (April 2023).

[36] Shefy Manayil Kareem. 2023. Introducing critical new water data capabilities in Microsoft Cloud for Sustainability. https://www.microsoft.com/en-us/industry/blog/sustainability/2023/03/22/introducing-critical-new-water-data-capabilities-in-microsoft-cloud-for-sustainability/

[37] Leila Karimi, Leeann Yacuel, Joseph Degraft-Johnson, Jamie Ashby, Michael Green, Matt Renner, Aryn Bergman, Robert Norwood, and Kerri L. Hickenbottom. 2022. Water-Energy Tradeoffs in Data Centers: A Case Study in Hot-arid Climates. *Resources, Conservation and Recycling* 181 (2022), 106194. https://doi.org/10.1016/j.resconrec.2022.106194

[38] Oyku Deniz Kose and Yanning Shen. 2023. Fast&Fair: Training Acceleration and Bias Mitigation for GNNs. *Transactions on Machine Learning Research* (2023). https://openreview.net/forum?id=nOk4XEB7Ke

[39] Kien Le, Ricardo Bianchini, Thu D. Nguyen, Ozlem Bilgir, and Margaret Martonosi. 2010. Capping the Brown Energy Consumption of Internet Services at Low Cost. In *IGCC*.

[40] Kien Le, Ricardo Bianchini, Jingru Zhang, Yogesh Jaluria, Jiandong Meng, and Thu D. Nguyen. 2011. Reducing electricity cost through virtual machine placement in high performance computing clouds. In *SuperComputing*.

[41] Stephen Lee, Rahul Urgaonkar, Ramesh Sitaraman, and Prashant Shenoy. 2015. Cost Minimization Using Renewable Cooling and Thermal Energy Storage in CDNs. In *ICAC*.

[42] Chao Li, Amer Qouneh, and Tao Li. 2012. iSwitch: Coordinating and Optimizing Renewable Energy Powered Server Clusters. In *ISCA*.

[43] Pengfei Li, Jianyi Yang, Mohammad A. Islam, and Shaolei Ren. 2023. Making AI Less "Thirsty": Uncovering and Addressing the Secret Water Footprint of AI Models. *arXiv 2304.03271* (2023).

[44] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. 2020. Fair Resource Allocation in Federated Learning. In *International Conference on Learning Representations*. https://openreview.net/forum?id=ByexElSYDr

[45] Zhenhua Liu, Minghong Lin, Adam Wierman, Steven H. Low, and Lachlan L.H. Andrew. 2011. Greening geographical load balancing. In *SIGMETRICS*.

[46] Alexandra Sasha Luccioni, Sylvain Viguier, and Anne-Laure Ligozat. 2023. Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model. *Journal of Machine Learning Research* 24, 253 (2023), 1–15. http://jmlr.org/papers/v24/23-0069.html

[47] Jordan Macknick, Robin Newmark, Garvin Heath, and KC Hallett. 2011. A Review of Operational Water Consumption and Withdrawal Factors for Electricity Generating Technologies. *NREL Tech. Report: NREL/TP-6A20-50900* (2011).

[48] Electricity Maps. 2023. Open Database. https://www.electricitymaps.com/

[49] Natalia Martinez, Martin Bertran, and Guillermo Sapiro. 2020. Minimax Pareto Fairness: A Multi Objective Perspective. In *Proceedings of the 37th International Conference on Machine Learning (ICML'20)*. JMLR.org, Article 627, 10 pages.

[50] Meta. 2021. Sustainability Report. https://sustainability.fb.com/

[51] Meta. 2023. Sustainability — Water. https://sustainability.fb.com/water/

[52] D. Meyer and D. Thevenard. 2019. PsychroLib: A Library of Psychrometric Functions to Calculate Thermodynamic Properties of Air. *Journal of Open Source Software* 4, 33 (2019), 1137. https://doi.org/10.21105/joss.01137

[53] Microsoft. 2022. Environmental Sustainability Report. https://www.microsoft.com/en-us/corporate-responsibility/sustainability/report

[54] Microsoft. 2023. Carbon-aware Computing: Measuring and Reducing the Carbon Footprint Associated with Software in Execution. In *Whitepaper*.

[55] Steven Gonzalez Monserrate. 2022. The Cloud Is Material: On the Environmental Impacts of Computation and Data Storage. *MIT Case Studies in Social and Ethical Responsibilities of Computing* Winter (January 2022). https://mit-serc.pubpub.org/pub/the-cloud-is-material.

[56] M. J. Neely. [n. d.]. Universal Scheduling for Networks with Arbitrary Traffic, Channels, and Mobility, http://arxiv.org/abs/1001.0960.

[57] OECD. 2022. Measuring the Environmental Impacts of Artificial Intelligence Compute and Applications: The AI Footprint. *OECD Digital Economy Papers* 341 (2022). https://doi.org/https://doi.org/10.1787/7babf571-en

[58] Johanna Okerlund, Evan Klasky, Aditya Middha, Sujin Kim, Hannah Rosenfeld, Molly Kleinman, and Shobita Parthasarathy. 2022. What's in the Chatterbox? Large Language Models, Why They Matter, and What We Should Do About Them. *University of Michigan Technology Assessment Project Report* (April 2022).

[59] David Patterson, Joseph Gonzalez, Urs Hölzle, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David R. So, Maud Texier, and Jeff Dean. 2022. The Carbon Footprint of Machine Learning Training Will Plateau, Then Shrink. *Computer* 55, 7 (2022), 18–28. https://doi.org/10.1109/MC.2022.3148714

[60] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon Emissions and Large Neural Network Training. arXiv:2104.10350 [cs.LG]

[61] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. 2008. Discrimination-Aware Data Mining. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Las Vegas, Nevada, USA) *(KDD '08)*. Association for Computing Machinery, New York, NY, USA, 560–568. https://doi.org/10.1145/1401890.1401959

[62] Dana Pessach and Erez Shmueli. 2022. A Review on Fairness in Machine Learning. *ACM Comput. Surv.* 55, 3, Article 51 (feb 2022), 44 pages. https://doi.org/10.1145/3494672

[63] Asfandyar Qureshi, Rick Weber, Hari Balakrishnan, John Guttag, and Bruce Maggs. 2009. Cutting the Electric Bill for Internet-Scale Systems. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication* (Barcelona, Spain) *(SIGCOMM '09)*. Association for Computing Machinery, New York, NY, USA, 123–134. https://doi.org/10.1145/1592568.1592584

[64] 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. 2023. Carbon-Aware Computing for Datacenters. *IEEE Transactions on Power Systems* 38, 2 (2023), 1270–1280. https://doi.org/10.1109/TPWRS.2022.3173250

[65] Samyam Rajbhandari, Conglong Li, Zhewei Yao, Minjia Zhang, Reza Yazdani Aminabadi, Ammar Ahmad Awan, Jeff Rasley, and Yuxiong He. 2022. DeepSpeed-MOE: Advancing Mixture-of-Experts Inference and Training to Power Next-Generation AI Scale. In *ICML*.

[66] Bogdana Rakova and Roel Dobbe. 2023. Algorithms as Social-Ecological-Technological Systems: an Environmental Justice Lens on Algorithmic Audits. In *Proceedings of the 2023 Conference on Fairness, Accountability, and Transparency* (Chicago, Illinois) *(FAccT '23)*. Association for Computing Machinery, New York, NY, USA.

[67] L. Rao, X. Liu, L. Xie, and Wenyu Liu. 2010. Reducing electricity cost: Optimization of distributed Internet data centers in a multi-electricity-market environment. In *INFOCOM*.

[68] Paul Reig, Tianyi Luo, Eric Christensen, and Julie Sinistore. 2020. Guidance for Calculating Water Use Embedded in Purchased Electricity. *World Resources Institute* (2020).

[69] Chuangang Ren, Di Wang, Bhuvan Urgaonkar, and Anand Sivasubramaniam. 2012. Carbon-Aware Energy Capacity Planning for Datacenters. In *MASCOTS*.

[70] Nripsuta Ani Saxena, Karen Huang, Evan DeFilippis, Goran Radanovic, David C. Parkes, and Yang Liu. 2020. How Do Fairness Definitions Fare? Testing Public Attitudes Towards Three Algorithmic Definitions of Fairness in Loan Allocations. *Artificial Intelligence* 283 (2020), 103238. https://doi.org/10.1016/j.artint.2020.103238

[71] Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. 2020. Green AI. *Commun. ACM* 63, 12 (nov 2020), 54–63. https://doi.org/10.1145/3381831

[72] A. Shehabi, S. J. Smith, N. Horner, I. Azevedo, R. Brown, J. Koomey, E. Masanet, D. Sartor, M. Herrlin, and W. Lintner. 2016. United States Data Center Energy Usage Report. *Lawrence Berkeley National Laboratory, Berkeley, California. LBNL-1005775* (2016).

[73] David Sickinger, Otto Van Geet, Suzanne Belmont, Thomas Carter, and David Martinez. 2018. Thermosyphon Cooler Hybrid System for Water Savings in an Energy-Efficient HPC Data Center. *NREL Technical Report NREL/TP-2C00-72196* (2018).

[74] Md Abu Bakar Siddik, Arman Shehabi, and Landon Marston. 2021. The Environmental Footprint of Data Centers in the United States. *Environmental Research Letters* 16, 6 (2021), 064017.

[75] Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and Policy Considerations for Deep Learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 3645–3650. https://doi.org/10.18653/v1/P19-1355

[76] Jennifer Switzer, Gabriel Marcano, Ryan Kastner, and Pat Pannuto. 2023. Junkyard Computing: Repurposing Discarded Smartphones to Minimize Carbon. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 400–412. https://doi.org/10.1145/3575693.3575710

[77] L. Tassiulas and S. Sarkar. 2002. Maxmin fair scheduling in wireless networks. In *Proceedings 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 2. 763–772 vol.2. https://doi.org/10.1109/INFCOM.2002.1019322

[78] The Green Grid. 2011. Water Usage Effectiveness (WUE): A Green Grid Data Center Sustainability Metric. *Whitepaper* (2011).

[79] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. 2022. LaMDA: Language Models for Dialog Applications. arXiv:2201.08239 [cs.CL]

[80] Paul Allen Torcellini, Nicholas Long, and Ron Judkoff. 2003. Consumptive water use for US power production. *National Renewable Energy Laboratory Technical Paper (TP-550-33905)* (December 2003).

[81] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288* (2023).

[82] UNESCO. 2022. Recommendation on the Ethics of Artificial Intelligence. In *Policy Recommendation*.

[83] U.S. Department of Engergy. [n. d.]. What Is Environmental Justice? https://www.energy.gov/lm/what-environmental-justice

[84] U.S. Energy Information Administration. 2024. Open Data. https://www.eia.gov/opendata/
[85] U.S. Environmental Protection Agency. [n. d.]. About the U.S. Electricity System and its Impact on the Environment. ([n. d.]). https://www.epa.gov/energy/about-us-electricity-system-and-its-impact-environment
[86] Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, et al. 2022. Sustainable AI: Environmental Implications, Challenges and Opportunities. In *Proceedings of Machine Learning and Systems*, Vol. 4. 795–813.
[87] Pengfei Xu, Xiaofan Zhang, Cong Hao, Yang Zhao, Yongan Zhang, Yue Wang, Chaojian Li, Zetong Guan, Deming Chen, and Yingyan Lin. 2020. AutoDNNchip: An Automated DNN Chip Predictor and Builder for Both FPGAs and ASICs. In *FPGA*.
[88] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning Fair Representations. In *Proceedings of the 30th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 28)*, Sanjoy Dasgupta and David McAllester (Eds.). PMLR, Atlanta, Georgia, USA, 325–333. https://proceedings.mlr.press/v28/zemel13.html
[89] Xueru Zhang and Mingyan Liu. 2021. *Fairness in Learning-Based Sequential Decision Algorithms: A Survey*. Springer International Publishing, Cham, 525–555. https://doi.org/10.1007/978-3-030-60990-0_18

APPENDIX

A ADDITIONAL DETAILS ABOUT THE EXPERIMENTAL SETUP

In this section, we provide additional details about the data used in the experiments, particularly the sources of our data on the fuel mix and water intensity factors.

We do not have free access to the hourly energy fuel mix data for our data center locations in Europe and Asia. Thus, we generate synthetic hourly fuel mixes for these locations based on the U.S. data. We first obtain from [30] the average percentages of renewable and non-renewable energy in electricity generation between September 23 and October 11, 2022, for each data center location in Europe and Asia. Then, we scale the hourly energy fuel mix data in the U.S. to match the average percentages by mapping Texas' fuel mixes between June 1 and June 19, 2022, to Germany with non-renewable energy fuels scaled by 0.8503, Georgia's fuel mixes between June 1 and June 19, 2022, to Belgium with non-renewable energy fuels scaled by 1.5319, Georgia's fuel mixes between March 1 and March 19, 2022, to the Netherlands with non-renewable energy fuels scaled by 1.2759, Oregon's fuel mixes between July 1 and July 19, 2022, to Denmark with non-renewable energy scaled by 0.2657, Nevada's fuel mixes between March 1 and March 19, 2022, to Japan with non-renewable energy fuels scaled by 3.2374, Georgia's fuel mixes between May 1 and May 19, 2022, to Singapore with non-renewable energy fuels scaled by 4.4875. We choose different 18-day periods in order to de-correlate the European and Asian energy fuel mix traces from our actual U.S. data over the workload trace period (between September 23 and October 11, 2022).

We also show the estimated energy water intensity factor (EWIF) in m^3/MWh for common energy fuel types in the U.S. in Table 4 [43, 47], and the details of our 10 data center locations in Table 3.

B ADDITIONAL RESULTS FOR GLB WITH PARTIAL FLEXIBILITY

In this experiment, the goal is to evaluate the performance of eGLB under partial GLB flexibility. Intra-continental workload routing is fully flexible but inter-continental workload routing is partially restricted. Specifically, we only allow partial inter-continental workload routing as follows: workloads can be flexibly routed between Asia and the western U.S. (Nevada), and between Europe and the eastern U.S. (Virginia and Georgia). Similar to the full GLB flexibility scenario, the increase of the learning rate η helps improve the environmental equity at the expense of increasing the energy cost. The results are shown in Fig. 3.

C EXTENSION TO HETEROGENEOUS AI MODELS

For the same inference service, a set of heterogeneous AI models with distinct computing resource consumption and accuracy performance may be available via model pruning and compression in practice [26], offering flexible energy-accuracy tradeoffs. For example, there are eight different GPT-3 model sizes, ranging from the smallest one with 125 million parameters to the largest one with 175 billion parameters [9]. Now, we extend our problem formulation to this generalized setting.

Suppose that there are a set $\mathcal{L} = \{1, \cdots, L\}$ of heterogeneous AI models for our considered inference service. For time t, we can dynamically choose to run one or more AI models to serve the incoming workloads. This is also equivalent to distributing the workload $\sum_{j\in\mathcal{J}} x_{i,j}(t)$ to L heterogeneous AI models within data center i. We denote by $y_{i,l}(t) \geq 0$ as the amount of workloads distributed to AI model l in data center i. Naturally, $y_{i,l}(t) = 0$ means that the AI model l is not chosen in data center i at time t.

When deployed in data center i, the energy consumption and server resource usage of AI model l for processing workloads $y_{i,l}(t)$ are denoted by $e_{i,l}(y_{i,l}(t))$ and $r_{i,l}(y_{i,l}(t))$, respectively. Thus, the total server energy consumption in data center i becomes $\tilde{e}_i(y(t)) = \sum_{l\in\mathcal{L}} e_{i,l}(y_{i,l}(t))$, where $y(t) = \{y_{i,l}(t) \mid i \in \mathcal{N}, l \in \mathcal{L}\}$ represents the collection of decisions for workload assignment to different AI models. Similarly, with heterogeneous AI models, we can re-define the carbon footprint and water footprint as $\tilde{c}_{i,t}(y(t))$ and $\tilde{w}_{i,t}(y(t))$ by replacing $e_i(x(t))$ with $\tilde{e}_i(y(t)) = \sum_{l\in\mathcal{L}} e_{i,l}(y_{i,l}(t))$ in (2) and (3), respectively.

To optimally distribute workloads to AI models with different energy-accuracy tradeoffs, we need to consider the inference *cost* associated with different accuracies, since otherwise always choosing the smallest model can always result in the lowest energy consumption. Specifically, we refer to the cost as performance cost and denote it by $s_l(y_{i,l}(t))$, whose dependency on $y_{i,l}(t)$ can be explained by noting that the performance cost is potentially more significant when more users use the model (i.e., $y_{i,l}(t)$ is larger). Next, by combining the energy cost and performance cost, we consider a generalized *operational* cost as follows:

$$\tilde{g}_t(y(t)) = \sum_{i\in\mathcal{N}} \sum_{l\in\mathcal{L}} \left[p_{i,t}\gamma_{i,t} \cdot e_{i,l}(y_{i,l}(t)) + \phi \cdot s_l(y_{i,l}(t)) \right], \quad (15)$$

where the hyperparameter $\phi \geq 0$ converts the performance cost $s_l(y_{i,l}(t))$ to a monetary value and indicates the importance of inference performance relative to the energy cost.

Table 3: The detailed information of our data center locations. The values shown in the table are averaged over the 18-day period between September 23 and October 11, 2022.

Country	State/Province	City	Total WUE (m^3/MWh)	Carbon Intensity (ton/MWh)	Energy Price ($/MWh)
U.S.	Texas	Midlothian	5.7397	0.4011	64.931
U.S.	Virginia	Loudoun	5.9755	0.3741	77.793
U.S.	Georgia	Douglas	5.9001	0.4188	80.566
U.S.	Nevada	Storey	4.9306	0.2980	84.738
Germany	Hessen	Frankfurt	4.5889	0.3295	315.233
Belgium	Hainaut	Saint-Ghislain	4.9316	0.4802	247.083
Netherlands	Groningen	Eemshaven	3.0928	0.4454	248.258
Denmark	Fredericia	Fredericia	3.8900	0.1391	213.773
Japan	Chiba Prefecture	Inzai	2.4989	0.3280	129.269
Singapore	Singapore	Jurong West	5.8652	0.5260	155.462

Table 4: Estimated EWIF for common energy fuel types in the U.S. [47].

Fuel Type	Coal	Nuclear	Natural Gas	Solar (PV)	Wind	Other	Hydro
EWIF (L/kWh)	1.7	2.3	1.1	0	0	1.8	68 (0, if excluded)

Finally, we formulate the generalized GLB problem with heterogeneous AI models as follows:

$$\min_{x(t),y(t)} \sum_{t=1}^{T} \tilde{g}_t(y(t)) + \mu_c \cdot \max_{i\in\mathcal{N}} \left[\mathcal{H}_{i,c}\left(\sum_{t=1}^{T} \tilde{c}_{i,t}(y(t))\right)\right] + \mu_w \cdot \max_{i\in\mathcal{N}} \left[\mathcal{H}_{i,w}\left(\sum_{t=1}^{T} \tilde{w}_{i,t}\left(y(t)\right)\right)\right], \tag{16a}$$

$$s.t. \quad x_{i,j}(t) = 0, \ \text{ if } B_{i,j} = 0, \quad \forall\, i \in \mathcal{N}, j \in \mathcal{J}, t = 1, \cdots, T, \tag{16b}$$

$$\sum_{x\in\mathcal{N}} x_{i,j}(t) = \lambda_{j,t}, \quad \forall\, j \in \mathcal{J}, t = 1, \cdots, T, \tag{16c}$$

$$\sum_{k\in\mathcal{K}} r_{i,k}\left(y_{i,k}(t)\right) \le M_i, \quad \forall\, i \in \mathcal{N}, t = 1, \cdots, T, \tag{16d}$$

$$\sum_{j\in\mathcal{J}} x_{i,j}(t) = \sum_{k\in\mathcal{K}} y_{i,k}(t), \quad \forall\, j \in \mathcal{J}, t = 1, \cdots, T, \tag{16e}$$

where the objective (16a) is to minimize the generalized operational cost (including both energy and performance costs) while addressing environmental inequity, the constraint (16d) means that the total resource demand must be no more than the server cluster's capacity, and the last constraint (16e) ensures that the workload assigned to each data center is always served by one of the heterogeneous AI models. The problem (16a)–(16e) can be solved by introducing auxiliary variables and optimizing the Lagrangian with estimated dual variables. The algorithm is similar to Algorithm 1.

Next, we conduct a trace-based simulation to evaluate the performance of different GLB algorithms with heterogeneous AI models. We consider a similar setup as the default one with homogeneous AI models, where we keep the weights μ_w and μ_c unchanged. As the open-sourced BLOOM has only one model size and is not suitable for the heterogeneous AI model case, we consider the Llama-2 model released by Meta with three different available model sizes (7B, 13B, and 70B), corresponding to different accuracies and energy demands [81]. We normalize the average inference accuracy and energy consumption by that of the largest model. We set the accuracy performance weight such that the average inference accuracy is roughly the same as that of the model with the medium size. As each performance weight corresponds to an average inference constraint, this is essentially equivalent to constraining the average inference accuracy (weighted by the amount of requests for each model) to be equal to that of the medium-size model. Each of the 10 geo-distributed data centers can handle a specified quantity of requests for each model size subject to the total capacity constraint. By using the same traces for carbon intensity, water usage efficiency, and workloads as in the homogeneous case, we run different GLB algorithms and show the results Table 5.

The results provide similar insights as in the homogeneous case for both full and partial GLB scenarios. Specifically, we see that compared to the other equity-oblivious GLB algorithms, `eGLB` can effectively reduce the environmental inequity among different regions in terms of the maximum-to-average ratio for both carbon and water footprints. Additionally, even compared to `eGLB-Off`, `eGLB` delivers a comparable performance in terms of environmental equity while only slightly increasing the total energy cost. Again, this demonstrates the potential of `eGLB` to address AI's emerging environmental inequity.

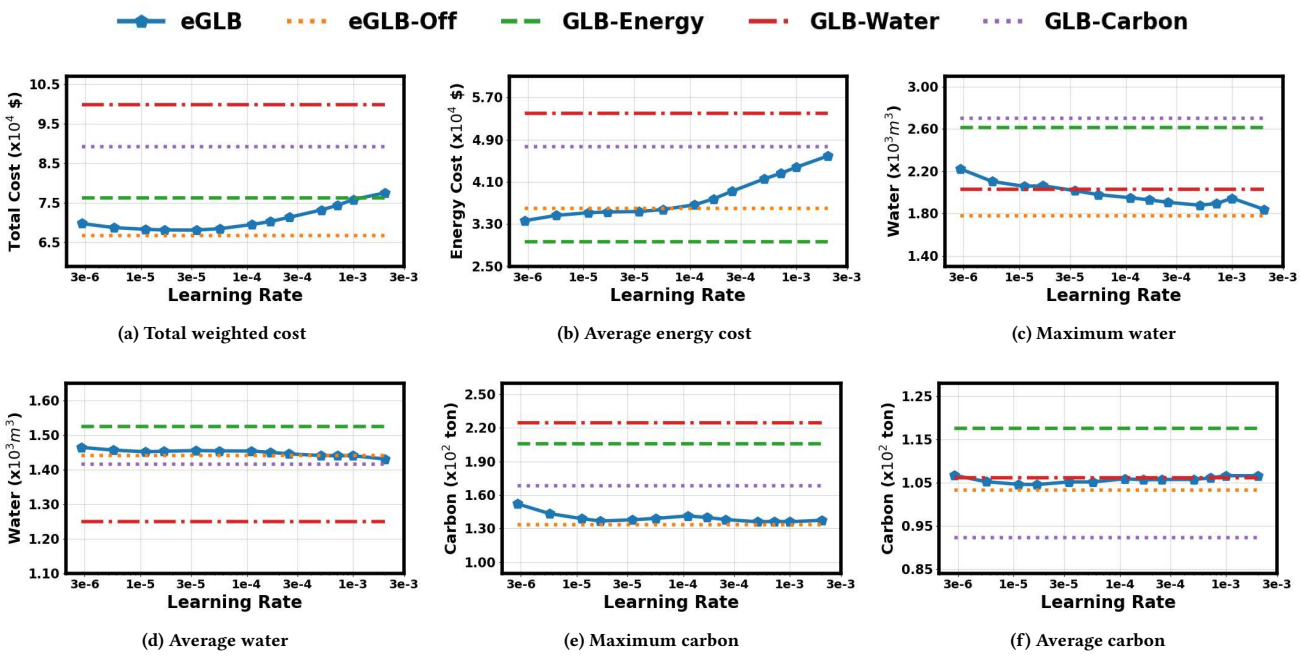
D PROOF OF THEOREM 1

When the reference function is $h(a) = \frac{1}{2}\|a\|^2$, the update rule in Line 7 in Algorithm 1 can be rewritten as

$$\frac{1}{\eta}\kappa_{t+1} = \left[\frac{1}{\eta}\kappa_t - d_t\right]^+ = \left[\frac{1}{\eta}\kappa_t + \left(\begin{bmatrix} C_t(x(t)) \\ \mathcal{W}_t(x(t)) \end{bmatrix} - \begin{bmatrix} z_c(t) \\ z_w(t) \end{bmatrix}\right)\right]^+ \tag{17}$$

where $[x]^+ = x$ when x is positive, otherwise it's set as zero. Given the dual variable κ_t, the optimization goal of Line 4 and Line 5 can be written as

$$\min_{x(t),z_c(t),z_w(t)} \frac{1}{\eta}\Big(g_t(x(t)) + \mu_c\|\mathcal{H}_c(z_c(t))\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t))\|_\infty\Big) + \frac{\kappa_t^\top}{\eta}\cdot(-d_t) \tag{18}$$



(a) Total weighted cost (b) Average energy cost (c) Maximum water

(d) Average water (e) Maximum carbon (f) Average carbon

Figure 3: The energy cost, carbon and water footprint of eGLB with different learning rates η under partial GLB flexibility. The results for eGLB-Off, GLB-Energy, GLB-Carbon, and GLB-Water are shown for comparison.

Table 5: Comparison of different GLB algorithms in terms of energy cost, carbon and water footprint with heterogeneous AI models. The results of eGLB with the learning rate $\eta = 10^{-5}$ are bolded.

GLB Flexibility	Metric		Algorithm							
			GLB-Energy	GLB-Carbon	GLB-Water	GLB-C2	GLB-All	GLB-Nearest	eGLB-Off	**eGLB**
Full	**Energy** (US$)	avg	20656	56891	58345	29047	26071	47038	29279	**35034**
	Water (m^3)	avg	1549.8	1311.9	1075.8	1476.7	1367.2	1446.7	1416.1	**1424.6**
		max	4521.5	4537.4	2788.5	3786.7	2640.2	2090.9	1612.7	**1830.9**
		max/avg	2.92	3.46	2.59	2.56	1.93	1.45	1.14	**1.29**
	Carbon (ton)	avg	121.37	67.21	100.17	87.73	98.63	111.80	96.90	**101.77**
		max	352.84	171.07	375.92	233.49	201.94	143.46	116.59	**129.94**
		max/avg	2.91	2.55	3.75	2.66	2.05	1.28	1.20	**1.28**
Full	**Energy** (US$)	avg	21684	49516	54478	27329	26094	47038	30011	**36170**
	Water (m^3)	avg	1561.8	1411.2	1106.9	1503.3	1385.9	1446.7	1442.1	**1437.6**
		max	4343.4	3699.6	2758.4	4106.1	3045.9	2090.9	1782.4	**1885.4**
		max/avg	2.78	2.62	2.49	2.73	2.20	1.45	1.24	**1.31**
	Carbon (ton)	avg	119.50	79.67	103.45	96.26	103.17	111.80	99.46	**103.33**
		max	340.16	183.15	351.76	254.43	230.12	143.46	131.22	**135.26**
		max/avg	2.85	2.30	3.40	2.64	2.23	1.28	1.32	**1.31**

by multiplying Eqn (18) with η, it corresponds to the change of Lagrange function at time t. Now we define a new variable $\Delta_1(t) = \frac{1}{2\eta^2}(\kappa_{t+1}^2 - \kappa_t^2)$ to quantify the change of Lagrange multiplier κ_t. Our next step is to provide bounds on this dual variable κ_t, which is done by the following lemma.

Lemma 2. *If the reference function $h(a) = \frac{1}{2}\|a\|^2$ and $\kappa_1 = 0$, then the dual variable κ_t is bounded by*

$$\|\kappa_{T+1}\| \leq \eta\sqrt{2T(B + \frac{M\theta_m}{\eta}(\mu_c c_m + \mu_w w_m))} \tag{19}$$

where constant $B = \frac{N}{2}(\bar{z}_c^2 + \bar{z}_w^2)$.

Proof. From Eqn (17), we have the following inequality

$$\Delta_1(t) \leq \frac{1}{2\eta^2}\left((\kappa_t - \eta d_t)^2 - \kappa_t^2\right) \tag{20}$$

$$= \frac{1}{\eta}\kappa_t \cdot (-d_t) + \frac{1}{2}d_t^2 \tag{21}$$

$$\leq \frac{1}{\eta}\kappa_t \cdot (-d_t) + B \tag{22}$$

where $B = \frac{N}{2}\cdot(\bar{z}_c^2 + \bar{z}_w^2)$, the first inequality is based on $([x]^+)^2 \leq x^2$ and the second inequality comes from our assumption that $\bar{z}_c$ and $\bar{z}_w$ are the largest possible values of the carbon and water footprint.

Suppose at time t, the optimal solution for Eqn (18) is $x(t)^\dagger$, $z_c(t)^\dagger, z_w(t)^\dagger$, for any other $z_c(t)' \in \mathcal{Z}_c, z_w(t)' \in \mathcal{Z}_w$ we have

$$\begin{aligned}&\Delta_1(t) + \frac{1}{\eta}\left(g_t(x(t)^\dagger) + \mu_c\|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty\right)\\ \le& B + \frac{1}{\eta}\left(g_t(x(t)^\dagger) + \mu_c\|\mathcal{H}_c(z_c(t)')\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)')\|_\infty\right)\\ &+ \frac{\kappa_t^\top}{\eta}\cdot\left(\begin{bmatrix}C_t(x(t)^\dagger)\\ \mathcal{W}_t(x(t)^\dagger)\end{bmatrix} - \begin{bmatrix}z_c(t)'\\ z_w(t)'\end{bmatrix}\right)\end{aligned} \tag{23}$$

In the second inequality, we choose $z_c(t)'$ and $z_w(t)'$ such that the term $\begin{bmatrix}C_t(x(t)^\dagger)\\ \mathcal{W}_t(x(t)^\dagger)\end{bmatrix} - \begin{bmatrix}z_c(t)'\\ z_w(t)'\end{bmatrix} = 0$, then we have

$$\begin{aligned}\Delta_1(t) \le& B + \frac{\mu_c}{\eta}\left(\|\mathcal{H}_c(z_c(t)')\|_\infty - \|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty\right) +\\ &\frac{\mu_w}{\eta}\left(\|\mathcal{H}_w(z_w(t)')\|_\infty - \|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty\right)\end{aligned} \tag{24}$$

$$\le B + \frac{M\theta_m}{\eta}(\mu_c c_m + \mu_w w_m) \tag{25}$$

where the second inequality results from the assumption of maximum carbon or water price, the maximum datacenter capacity M and θ_m, the maximum gradient of function $\mathcal{H}_w(\cdot)$ and $\mathcal{H}_c(\cdot)$. By summing up $\Delta_1(t)$ through $t = 1$ to T, then we have

$$\frac{1}{2\eta^2}(\kappa_{T+1}^2 - \kappa_1^2) \le T(B + \frac{M\theta_m}{\eta}(\mu_c c_m + \mu_w w_m)) \tag{26}$$

□

Using the previous result, we can now proceed by proving the following technical lemma, which when combined with the analysis above will let us complete the proof.

Lemma 3. *Suppose the optimal solution for Eqn* (18) *is* $x_{1:T}^\dagger$, $z_{c,1:T}^\dagger$ *and* $z_{w,1:T}^\dagger$, *for any* $x(t)', z_c(t)', z_w(t)'$ *satisfying the constraints in Eqn* (5b) – (5d) *we have*

$$\begin{aligned}&\sum_{t=1}^T \Delta_1(t) + \frac{1}{\eta}\sum_{t=1}^T\Big[g_t(x(t)^\dagger) + \mu_c\|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty\\ &+ \mu_w\|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty\Big]\\ \le& BT + BT(T-1) + \frac{\kappa_1}{\eta}\sum_{t=1}^T\left(\begin{bmatrix}C_t(x(t)'\\ \mathcal{W}_t(x(t)')\end{bmatrix} - \begin{bmatrix}z_c(t)'\\ z_w(t)'\end{bmatrix}\right)\\ &+ \frac{1}{\eta}\sum_{t=1}^T\left[g_t(x(t)') + \mu_c\|\mathcal{H}_c(z_c(t)')\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)')\|_\infty\right]\end{aligned} \tag{27}$$

Proof. Similar to Eqn (23), for any other $x(t) \in \mathcal{X}_t$, $z_c(t)' \in \mathcal{Z}_c, z_w(t)' \in \mathcal{Z}_w$ we have

$$\begin{aligned}&\Delta_1(t) + \frac{1}{\eta}\left(g_t(x(t)^\dagger) + \mu_c\|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty\right)\\ \le& B + \frac{1}{\eta}\left(g_t(x(t)') + \mu_c\|\mathcal{H}_c(z_c(t)')\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)')\|_\infty\right)\\ &+ \frac{\kappa_t^\top}{\eta}\cdot\left(\begin{bmatrix}C_t(x(t)')\\ \mathcal{W}_t(x(t)')\end{bmatrix} - \begin{bmatrix}z_c(t)'\\ z_w(t)'\end{bmatrix}\right)\end{aligned} \tag{28}$$

Now we define the subgradient of the action $x(t)'$ as $d_t' = \begin{bmatrix}C_t(x(t)')\\ \mathcal{W}_t(x(t)')\end{bmatrix} - \begin{bmatrix}z_c(t)'\\ z_w(t)'\end{bmatrix}$. According to the update rule of κ_t, the maximum difference between dual variables at $t = 1$ and $t = \tau + 1$ is bounded by

$$-\tau N(\bar{z}_c^2 + \bar{z}_w^2) \le \langle\frac{\kappa_{\tau+1}}{\eta} - \frac{\kappa_1}{\eta}, d_t'\rangle \le \tau N(\bar{z}_c^2 + \bar{z}_w^2) \tag{29}$$

Therefore, for all $t \in [1, T]$, we have

$$\frac{\kappa_t^\top}{\eta}\cdot(-d_t') \le \frac{\kappa_1^\top}{\eta}\cdot(-d_t') + N(t-1)(\bar{z}_c^2 + \bar{z}_w^2) \tag{30}$$

By summing up the inequality, we have

$$\begin{aligned}\sum_{t=1}^T\frac{\kappa_t^\top}{\eta}\cdot(-d_t') &\le \frac{\kappa_1^\top}{\eta}\left(\sum_{t=1}^T(-d_t')\right) + N(\bar{z}_c^2 + \bar{z}_w^2)\sum_{t=1}^T t-1\\ &= \frac{\kappa_1^\top}{\eta}\left(\sum_{t=1}^T(-d_t')\right) + T(T-1)\frac{N}{2}(\bar{z}_c^2 + \bar{z}_w^2)\end{aligned} \tag{31}$$

By summing up Eqn (28) through $t = 1$ to T, we finish the proof. □

We are now ready to complete the proof. Note that $\sum_{t=1}^T \Delta_1(t) \ge 0$. Suppose $x_{1:T}^*$ is the optimal solution to the Eqn (4), which also satisfies the constraints in Eqn (5b) – (5d). Substituting $x_{1:T}^*$ back to Eqn (27) gives

$$\begin{aligned}&\frac{1}{T}\sum_{\tau=t_0}^T\left[g_t(x(t)^\dagger) + \mu_c\|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty\right]\\ \le&\eta(B + B(T-1)) + \frac{1}{T}\sum_{\tau=t_0}^T\Big[g_t(x(t)^*) + \mu_c\|\mathcal{H}_c(z_c(t)^*)\|_\infty\\ &+ \mu_w\|\mathcal{H}_w(z_w(t)^*)\|_\infty\Big]\end{aligned} \tag{32}$$

where the $\sum_{t=1}^T\begin{bmatrix}z_c(t)^*\\ z_w(t)^*\end{bmatrix} = \sum_{t=1}^T\begin{bmatrix}C_t(x(t)^*)\\ \mathcal{W}_t(x(t)^*)\end{bmatrix}$, so the term $\kappa_1\sum_{t=1}^T d_t$ is equal to zero.

The left hand side of Eqn (32) is mixed up with GLB decisions $x(t)^\dagger$ and auxiliary variables $z_c(t)^\dagger$, $z_w(t)^\dagger$. The next step is to eliminate these auxiliary variables by bounding their difference. Based on the update rule of κ_t and Lemma 2, we have

$$\begin{aligned}&\frac{1}{T}\|\sum_{t=1}^T\left(\begin{bmatrix}C_t(x(t)')\\ \mathcal{W}_t(x(t)')\end{bmatrix} - \begin{bmatrix}z_c(t)'\\ z_w(t)'\end{bmatrix}\right)\|\\ \le&\frac{1}{T}\left(\|\frac{\kappa_{T+1}}{\eta}\| - \|\frac{\kappa_1}{\eta}\|\right)\\ \le&\sqrt{\frac{2}{T}(B + \frac{M\theta_m}{\eta}(\mu_c c_m + \mu_w w_m))}\end{aligned} \tag{33}$$

The maximum gradient of $\mu_c\|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty$ with respect to $[z_c(t), z_w(t)]$, is always bounded by $C = \theta_m(\mu_c + \mu_w)$. For simplicity, we define $D = \theta_m(\mu_c c_m + \mu_w w_m)$, then we have

$$
\begin{aligned}
&\mu_c\|\mathcal{H}_c(\frac{1}{T}\sum_{\tau=t_0}^{T} C_t(x(t)^\dagger))\|_\infty + \mu_w\|\mathcal{H}_w(\frac{1}{T}\sum_{\tau=t_0}^{T} C_t(x(t)^\dagger))\|_\infty \\
\leq &\mu_c\|\mathcal{H}_c(\frac{1}{T}\sum_{\tau=t_0}^{T} z_c(t)^\dagger)\|_\infty + \mu_w\|\mathcal{H}_w(\frac{1}{T}\sum_{\tau=t_0}^{T} z_w(t)^\dagger)\|_\infty \\
&+ \left[\theta_m(\mu_c+\mu_w)\right]\sqrt{\frac{2}{T}(B+\frac{M}{\eta}D)} \\
\leq &\frac{1}{T}\sum_{\tau=t_0}^{T}\left[\mu_c\|\mathcal{H}_c(z_c(t)^\dagger)\|_\infty + \mu_w\|\mathcal{H}_w(z_w(t)^\dagger)\|_\infty\right] \\
&+ \left[\theta_m(\mu_c+\mu_w)\right]\sqrt{\frac{2}{T}(B+\frac{M}{\eta}D)}
\end{aligned}
\tag{34}
$$

where the first inequality is based on the maximum gradient D, the second inequality is from Jensen's inequality. By substituting Eqn (34) back to Eqn (32), we recover the cost objective in Eqn (4a) and finish the proof.