2013 IEEE Third International Conference on Cloud and Green Computing

Optimizing Water Efficiency in Distributed Data Centers

Shaolei Ren
School of Computing and Information Sciences
Florida International University
E-mail: sren@cs.fiu.edu

***Abstract*—The number and scale of data centers explode with the dramatically surging demand for cloud computing services, resulting in huge electricity consumption as well as an enormous impact on sustainability. While numerous efforts have been dedicated to decreasing the carbon footprint of data centers, there is a surprising and also embarrassing lack of attention to the enormity of data center water consumption despite its emergence as a critical concern for future sustainability. In this paper, we take the first step towards the data center water efficiency. We first identify two characteristics of data center water efficiency: water efficiency varies by location and also over time. Then, by exploiting these characteristics, we propose a novel geographical load balancing (GLB) algorithm, called GLB for Water Sustainability (GLB-WS), which dynamically schedules workloads to water-efficient data centers for improving the overall water usage effectiveness (an emerging metric for quantifying data center water efficiency) while satisfying the electricity cost constraint. We also perform a trace-based simulation study to validate the analysis. The result shows that compared to the state-of-the-art cost-minimizing GLB, GLB-WS significantly improves the water efficiency (by 60%) and reduces the water consumption (by 51%).**

***Keywords*-Data center, Geographic load balancing, Sustainability, Water efficiency**

I. Introduction

The emergence of a plethora of Internet and cloud services has been constantly urging service providers to expand the number and scale of data centers, resulting in a huge demand for electricity as well as a profound impact on the existing ecosystem. In light of the serious sustainability concerns, tremendous efforts have been dedicated to decreasing the energy consumption as well as carbon footprints of data centers (see, e.g., [15], [17], [18] and references therein).

While reducing carbon footprint is clearly essential for sustainability, an equally, if not more, important aspect of data center sustainability is *water footprint*. Just as carbon footprint is embedded in electricity energy (e.g., produced by coal) and attributed to data centers, data centers are also held accountable for the enormous *water* consumption associated with electricity generation (i.e., evaporated water during steam condensation) [19], [24]. A recent study shows that even without considering the water usage in hydroelectricity, 1.8 liters of water is *consumed* for producing 1 kilowatt hour (KWh) of electricity on average in the U.S. [19], [24]. In addition to the indirect water consumption incurred on the energy source side, data centers also directly consume a significant amount of on-site water (mostly for cooling systems): e.g., it is reported that the U.S. National Security Agency's massive data center in Utah consumes *1.5 million* gallons of cooling water each day [20]. Combining both source-side and on-site water consumption [24], data centers' water footprint requires immediate attention, as urged by industry consortium and public media (e.g., [5], [24]). Nonetheless, despite its emergence as an extremely important concern in sustainability, water efficiency of data centers has been embarrassingly long-neglected, thereby motivating us to take the first step to rigorously address the increasingly critical water issue in data centers.

The extensively-researched energy-efficient techniques (e.g., [14], [17]) might seem to be sufficient for reducing water footprints, but they are far from being adequate because they fail to incorporate the *temporal* and *spatial* diversities of data center water efficiency: as specified in the next section for the first time, water efficiency (quantified in terms of water consumption for each kWh of IT energy) varies significantly by location and also over time. Thus, the total water footprint can be effectively reduced by appropriately deciding "where" and "when" to process workloads, while the total energy consumption would not be affected. More recently, large IT companies have begun to reduce direct on-site cooling water consumption via facility innovation: e.g., using recycled/industry water or seawater instead of potable water, and directly using outside cold air as the cooling mechanism [2], [6]. These *engineering*-based approaches, however, suffer from several **limitations**. First, they require appropriate climate conditions and/or desirable locations that are not applicable for all data centers (e.g., "free air cooling" is ideally suitable in cold areas such as Dublin where Google has one data center [2]). Second, they do not address, and may even increase, indirect off-site water consumption (e.g., on-site facilities for treating industry water or seawater save freshwater but may consume more electricity [23]). Last but not least, some of these techniques, such as building water treatment facilities, often require substantial capital investments that may not be affordable for all data center operators.

In this paper, by exploiting *temporal* and *spatial* diversities of water efficiency, we propose a novel geographical load balancing (GLB) approach, called GLB for Water Sustainability (GLB-WS), which dynamically schedules

978-0-7695-5114-2/13 $26.00 © 2013 IEEE
DOI 10.1109/CGC.2013.19

workloads to water-efficient data centers for improving the overall water efficiency while satisfying the electricity cost constraint. Our approach is *software*-based and fundamentally differs from the existing engineering-based techniques that focus on facility innovation. We also perform a trace-based simulation study to complement the analysis. The result is consistent with our analysis: it shows that GLB-WS can effectively direct workloads from water-consuming data centers to water-efficient ones and that, compared to the state-of-the-art cost-minimizing GLB approach, GLB-WS significantly improves the water efficiency (by approximately 60%) and reduces the water consumption (by approximately 51%).

In summary, the specific goal of this paper is to optimize the water efficiency using GLB while satisfying the electricity cost constraint. To our best knowledge, this paper makes the first step towards optimizing water efficiency via workload management in data centers. Compared to the existing studies (and in particular, GLB techniques [15], [16], [18], [21], [22], [25]), our research on water efficiency provides an important, unique and complementary perspective to the existing data center research, and we take the position that incorporating water efficiency is essential in future research efforts.

The rest of this paper is organized as follows. Section II provides a brief description and two characteristics of data center water usage. Section III describes the model. In Section IV, we present the problem formulation and develop our algorithm, GLB-WS. Section V provides a simulation study. Related work is reviewed in Section VI, and finally, concluding remarks are offered in Section VII.

II. Water Consumption in Data Centers

In this section, we provide a brief background for data center water consumption, introduce an emerging metric for quantifying water efficiency, and also identify two key characteristics of water efficiency in data centers.

A. Water consumption

We would like to first draw the readers' attention to the subtle difference between water *withdrawal* and water *consumption*. The former refers to getting water from somewhere (e.g., public water facilities), whereas the latter refers to "losing" water (e.g., into the environment via evaporation) and producing waste water (e.g., into sewage systems[1]) [24]. In this paper, we focus on water consumption (also interchangeably referred to as water usage wherever applicable) which bears an immediate impact on the fresh clean water availability.

In general, data centers consume water both directly and indirectly [24].

[1]Treating waste water may be energy-consuming and hence also indirectly "consumes" fresh clean water [19].

Figure 1. Direct WUE of Facebook's data center in Prineville, OR (Feb. 27 to May 28, 2013) [6].

• *Direct water consumption*: Cooling systems (especially water-cooled chiller systems that employ evaporation as the heat rejection mechanism) in data centers use water directly. For example,[2] even with outside air cooling, Facebook's water efficiency is still 0.22L/KWh (for cooling systems) in its latest data center in Prineville, OR [6], whereas eBay uses 1.62L/kWh (as of May 29, 2013) [4].

• *Indirect water consumption*: Indirect water usage stems from the process of thermoelectricity generation that employs evaporation for cooling and hence consumes an astonishing amount of water [6], [19]. While certain types of electricity energy (e.g., by solar photovoltaics and wind) consume virtually zero water, "water-free" electricity only takes up a very small portion in the total electric generation capacity (e.g., less than 10% in the U.S. [19]). Moreover, although much of the water withdrawn by power plants for steam condensing eventually returns to the system (hence, not considered as "consumed"), a non-negligible fraction of the withdrawn water is "lost/consumed" by evaporation: e.g.,, the U.S., the national average water *consumption* is 1.8L/kWh (which is also referred to as Energy Water Intensity Factor, or EWIF) [19], [24].

Adding up direct and indirect water consumption, data centers are increasingly "thirsty" for water and the net water consumption requires immediate attention [24].

B. Measuring water efficiency

To evaluate the water efficiency, Green Grid has recently developed a new metric, called *water usage effectiveness* (WUE), defined as [24]

$$\text{WUE} = \frac{\text{Direct Water Usage + Indirect Water Usage}}{\text{IT Equipment Energy}}. \quad (1)$$

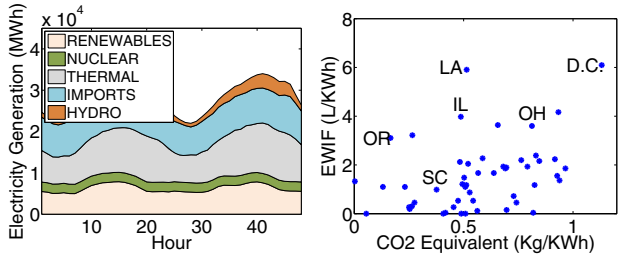
Note that the less WUE, the more water efficient a data center is, and theoretically, the minimum WUE is zero (L/KWh).

C. Characteristics of data center water efficiency

Data center water efficiency exhibits both spatial and temporal diversities, as specified below.

• *Spatial diversity:* Both direct and indirect WUEs demonstrate a significant variation across different geographic locations. For example, Fig. 2(b) shows the spatial diversity

[2]To our best knowledge, as of June 10, 2013, Facebook and eBay are the only two companies reporting data center water usage information [4], [6].



(a) California electricity fuel mix on June 16&17, 2013 [1]

(b) State-level EWIF versus CO_2 emissions in the U.S. [3], [24]

Figure 2. California electricity fuel mix and EWIF versus carbon emissions.

of the average EWIF in state level (because some states use more water-efficient technologies or produce more solar/wind electricity, while other states produce more water-consuming thermal and nuclear electricity). By comparing the direct WUEs (only for cooling systems) of Facebook's data centers in Prineville, OR and Forest City, NC, we also notice a significant variation between the two locations (due to different cooling technologies). Such spatial diversity in direct WUE can also be seen by comparing the direct WUEs of Facebook's and eBay's data centers (i.e., 0.22 versus 1.62 L/KWh) [4], [6].

• *Temporal diversity:* Fig. 1 shows a 90-day history of average daily (direct) WUE, and we can see that the WUE changes drastically over the time. While there is no public data for real-time EWIF in different cities/states, it is evidently time-varying as well because, as in Fig. 2(a), electricity fuel mix is time-varying and different electricity generation methods consume different amounts of water (excluding hydroelectric for which water consumption is difficult to evaluate, nuclear electricity consumes the most water, followed by coal-based thermal electricity and then solar PV/wind electricity [19]).

The existing GLB techniques that focus on either carbon efficiency (e.g., [16], [18], [25]) or electricity cost minimization (e.g., [21], [22]) do not necessarily optimize the water efficiency, because carbon/electricity-efficient data centers may not be water-efficient. This can be seen from Fig. 2(b), in which the (indirect) water efficiency is not in proportion to carbon efficiency. The relation between electricity cost and water efficiency is similar (see [3], [24]), but not shown here for brevity. Therefore, a new GLB is needed for optimizing data center water efficiency, which we will address in the following sections.

III. Model

We consider a discrete-time model with equal-length time slots indexed by $t = 1, 2, \cdots$, each of which has a duration that matches the timescale for which the data center operator can accurately predict the future information (including the workload arrival rate, on-site renewable energy supply and/or electricity price) and update its resource management decision. In the following analysis, as in the existing literature (e.g. [18], [22]), we mainly focus on hour-ahead prediction and hourly decisions for the convenience of presentation. The time index t is omitted wherever applicable without causing ambiguity.

Table I
List of notations.

Notation	Description	Notation	Description
$\lambda_j(t)$	Job arrivals	$a_i(t)$	Load distribution
$p_i(t)$	Server power	$r_i(t)$	On-site renewables
$w_i(t)$	Water usage	$m_i(t)$	# of active servers
$e_i(t)$	Electricity cost	$d_i(t)$	Average delay

Next, we present the modeling details for the data centers and workloads. Key notations are summarized in Table I.

Server. We consider N geo-distributed data centers, indexed by $i = 1, 2, \cdots, N$. Each data center i is partially powered by on-site renewable energy plants (e.g., solar panel and/or wind turbines) and contains M_i servers that are homogeneous within the data center, while heterogeneous servers can be easily captured by extending our model.[3] Processing speed of a server is quantified according to the service rate (rather than the actual clock rates), i.e., the *average* number of jobs processed in a unit time. Specifically, the service rate of a server in data center i is μ_i.

We denote by $m_i(t) \in [0, M_i]$ the number of servers turned on in data center i at time t. While in theory $m_i(t)$ should be integers, approximating it as continuous values does not affect the optimization result significantly because there are usually tens of thousands of servers in a data center [17], [18]. In our study, we focus on server power consumption, while the power consumption of other parts such as power supply system and cooling system are captured by the (possibly time-varying) power usage effectiveness (PUE) factor which, multiplied by the server power consumption, gives the total data center power consumption. Mathematically, we denote the total server power consumption[4] of data center i during time t by $p_i(a_i(t), m_i(t))$, which can be expressed as

$$p_i(a_i(t), m_i(t)) = m_i(t) \cdot \left[e_{0,i} + e_{c,i} \frac{a_i(t)}{m_i(t)\mu_i} \right], \qquad (2)$$

where $a_i(t) = \sum_{j=1}^{J} \lambda_{i,j}(t)$ is the total amount of workloads dispatched to data center i (with $\lambda_{i,j}(t)$ being the amount of workloads originating from the j-th gateway, as will be specified in the next subsection), $e_{0,i}$ is the static server power regardless of the workloads (as long as a server is turned on) and $e_{c,i}$ is the computing power incurred only when a server is processing workloads in data center i.

[3] If heterogeneous servers are considered, we need to decide how many servers of each type are turned on to process workloads.

[4] This is equivalent to energy consumption, since the length of each time slot is the same.

Electricity cost. We denote the electricity price in data center i at time t by $u_i(t)$, which is known to the data center operator no later than the beginning of time t and may change over time if the data centers participate in real-time electricity markets (e.g., hourly market [18]). Given $r_i(t) \in [0, r_{i,\max}]$ amount of available on-site renewable energy in data center i, we can express the incurred electricity cost of data center i as

$$e_i(a_i(t), m_i(t)) = u_i(t) \left[\gamma_i(t) \cdot p_i(a_i(t), m_i(t)) - r_i(t)\right]^+, \tag{3}$$

where $\gamma_i(t)$ is the PUE of data center i and $[\ \cdot\ ]^+ = \max\{\cdot, 0\}$ indicates that no electricity will be drawn from the power grid if on-site renewable energy is already sufficient. While we use Eqn. (3) to represent the electricity cost for data center i at time t (as considered by [18], [22]), our analysis is not restricted to a linear electricity cost function and can also model other electricity cost functions such as nonlinear convex functions (e.g., data centers are charged at a higher price if it consumes more power).

Water consumption. Water is consumed both directly (i.e., by cooling system) and indirectly (i.e., by electricity generation) in data centers. The direct water consumption can be easily obtained by multiplying the server power consumption with direct WUE, while the indirect water consumption depends on the electricity usage and the local EWIF. Specifically, we can express the water consumption of data center i at time t as

$$\begin{aligned} w_i(t) =& \epsilon_{i,d}(t) \cdot p_i(a_i(t), m_i(t)) \\ &+ \epsilon_{i,id}(t) \cdot \left[\gamma_i(t) \cdot p_i(a_i(t), m_i(t)) - r_i(t)\right]^+, \end{aligned} \tag{4}$$

where $\epsilon_{i,d}(t)$ is the direct WUE at time t and $\epsilon_{i,id}(t)$ is the EWIF of the electricity powering data center i. Note that, while the value of $\epsilon_{i,id}(t)$ is determined based on the energy fuel mix and can be obtained by inquiring the local utility company, acquiring the value of $\epsilon_{i,d}(t)$ is not straightforward because it seems to depend on various factors such as cooling technology, humidity and temperature and there is no publicly available study on this (to our best knowledge and also as corroborated by Green Grid's claim that the study of WUE is still at the very beginning [24]). In this paper, we assume that $\epsilon_{i,d}(t)$ is known at the beginning of time slot t, while noting that it could be a potential separate research topic to find the key factors affecting $\epsilon_{i,d}(t)$.

Workload. As in [18], [22], we consider a scenario in which there are J gateways, each of which represents a geographically-concentrated source of workloads (e.g., a state or province) and then forwards the incoming workloads to the N geo-distributed data centers. The term "workload" is generic, representing a synthesis of computing tasks/jobs (e.g., search requests, video streaming). We denote the workload arrival rate at the j-th gateway by $\lambda_j(t) = [0, \lambda_{j,\max}]$, and the workload is dispatched to data center i at a rate of $\lambda_{i,j}(t)$ that we shall optimize. We assume that $\lambda_j(t)$ is available (e.g., by using regression-based prediction) at the beginning of each time slot t, as widely considered in prior work [10], [15], [18].

We quantify the *overall* end-to-end delay performance for processing workloads from gateway j in data center j using the average delay $d_{i,j}(a_i(t), m_i(t))$, which is intuitively increasing in $a_i(t) = \sum_{j=1}^{J} \lambda_{i,j}(t)$ and decreasing in $m_i(t)$ where $m_i(t)$ is the number of (homogeneous) servers turned on in data center i [17], [18], [22]. As a concrete example, we can model the service process at each server as an M/G/1/PS queue [18]. Then, by incorporating the network transmission delay, the end-to-end average delay of workloads scheduled from gateway j to data center i is

$$d_{i,j}(a_i(t), m_i(t)) = \frac{1}{\mu_i - a_i(t)/m_i(t)} + l_{i,j}(t), \tag{5}$$

where $a_i(t) = \sum_{j=1}^{J} \lambda_{i,j}(t)$ represents the total workloads processed in data center i, and $l_{i,j}(t)$ is average network delay approximated in proportion to the distance between data center i and the j-th gateway, which can be well estimated by various approaches such as mapping and synthetic coordinate approaches. It should be further made clear that our delay model is mainly intended to characterize the general *trend* of the overall end-to-end performance and to facilitate the server provisioning decision, while the delay performance for specific tenants/applications are handled using separate techniques beyond the scope of our study.

IV. Geographic Load Balancing

In this section, we first present optimization objective, constraints, as well as the problem formulation for minimizing WUE via geographic load balancing. Then, we develop an efficient algorithm to solve the problem.

A. Problem Formulation

In this subsection, we first specify the optimization objective as well as constraints, and then present the problem formulation.

Objective. Based on the metric recently developed for measuring data center water efficiency [6], [24], we focus on maximizing the overall (hourly) WUE of all the data centers, specified as follows

$$g(\lambda(t), \mathbf{m}(t)) = \frac{\sum_{i=1}^{N} w_i(t)}{\sum_{i=1}^{N} p_i(a_i(t), m_i(t))}, \tag{6}$$

where $w_i(t)$ is the water usage (both directly and indirectly) and $p_i(\lambda_i(t), m_i(t))$ is the server power consumption in data center i, given by (4) and (2), respectively. As our study makes the first research effort to rigorously optimize data center water efficiency from the resource management perspective (fundamentally differing from the costly engineering approach of renovating the cooling system [6]), we make the following three remarks to clarify our objective. **First**, while WUE was originally proposed as a metric

for quantifying data center water efficiency over a year [24], we take the position that optimizing hourly WUE provides more timely information and may facilitate data center managers to improve their operations more promptly. This position has also been strengthened by Facebook's recent move to publicly report their hourly (direct) WUE [6]. **Second**, although we choose to optimize the *combined* WUE of all the data centers (because all the water usage will be attributed to the common data center owner, such as Google and Microsoft), alternative objectives such as (weighted) sum WUE of individual data centers and total water consumption can also be optimized as variants of our study. **Third**, while our current study focuses on optimizing water efficiency (which is undoubtedly an emerging critical issue for sustainable computing [6], [24]) without explicitly taking into account electricity energy minimization, we do not intend to downplay the seriousness of the soaring electricity consumption in data centers. In fact, one of our constraints is imposed on the total electricity cost, which *implicitly* bounds the maximum electricity consumption. We will jointly optimize the energy and water efficiency in our future work.

Constraints. The server provisioning and load distribution decisions need to satisfy

$$d_{i,j}(a_i, m_i) \leq D,\ \forall i, j, t, \tag{7}$$

$$m_i(t) \leq M_i,\ \forall i, t, \tag{8}$$

$$\sum_{i=1}^{N} \lambda_{i,j}(t) = \lambda_j(t),\ \forall j, t, \tag{9}$$

$$m_i(t)\mu_i > a_i(t) = \sum_{j=1}^{J} \lambda_{i,j}(t), \tag{10}$$

where (7) specifies the maximum average delay constraint to avoid intolerable data center performance, (8) imposes a capacity constraint, (9) and (10) prohibit workload dropping and server overloading, respectively. In addition, as considered in [25], the data center needs to satisfy its budget constraint (i.e., electricity cost in our study). In particular, the following constraint needs to be satisfied

$$\sum_{i=1}^{N} e_i(a_i(t), m_i(t)) \leq B(t),\ \forall i, t, \tag{11}$$

where $e_i(a_i, m_i)$ is the electricity cost of data center i given by (3). Note that the budget $B(t)$ is treated as exogenously given in our study, while we note that it may be optimized as a separate study based on long-term workload estimations (see [25] for details). Finally, note that water consumption is more of a sustainability issue and its bill is not considered as operational cost, because: (a) indirect water consumption is already paid in electricity bills; and (b) direct water consumption is relatively cheaper compared to electricity and the bill may be further reduced by using recycled or "grey" water [2].

Problem formulation. We now present the problem formulation as follows.

$$\mathbf{P1}: \quad \max_{\mathcal{A}(t)} g(\lambda(t), \mathbf{m}(t)) \tag{12}$$

$$s.t., \quad \text{constraints (7)–(11)}, \tag{13}$$

where $\mathcal{A}$ represents the server provisioning and load distribution decisions, i.e., $\mathbf{m}(t)$ and $\lambda(t)$, which we need to optimize. Solving the problem **P1** only requires the current electricity price, incoming workloads, available on-site renewable energies and local WUEs, which are readily available in practice by leveraging hour-ahead prediction techniques [17], [18], [22].

B. GLB-WS

In this subsection, we present an efficient solution to the problem **P1**, called GLB-WS, by reformulating **P1** as linear-fractional programming [8].

We note first that the non-linear delay constraint in (7) and the operator $[\ \cdot\ ]^+ = \max\{\cdot, 0\}$ prohibit direct application of linear-fractional optimization. To circumvent this difficulty, we *linearize* the constraint (7) and the operator $[\cdot]^+ = \max\{\cdot, 0\}$ by rewriting (7) as $\sum_{j=1}^{J} \lambda_{i,j}(t) \leq (\mu_i - \frac{1}{D}) m_i(t),\ \forall i, t$ and by introducing an auxiliary decision variable $z_i(t)$ indicating the amount of electricity usage of data center i such that $z_i(t) \geq p_i(a_i(t), m_i(t)) - r_i(t)$ and $z_i(t) \geq 0$. Next, we reformulate **P1** as a linear-fractional programming problem as follows.

$$\mathbf{P2}: \quad \min_{\mathcal{A} \bigcup z} \frac{\sum_{i=1}^{N} [\epsilon_{i,d} \cdot p_i(a_i, m_i) + \epsilon_{i,id} \cdot z_i]}{\sum_{i=1}^{N} p_i(a_i, m_i)} \tag{14}$$

$$s.t., \quad \text{constraints (8)–(11)}, \tag{15}$$

$$\sum_{j=1}^{J} \lambda_{i,j} \leq (\mu_i - \frac{1}{D + l_{i,j}}) m_i,\ \forall i, j \tag{16}$$

$$z_i \geq 0 \text{ and } z_i \geq p_i(a_i, m_i) - r_i,\ \forall i, \tag{17}$$

where, for brevity, we omit the time index t without causing ambiguity.

It can be seen from (2) that the server power $p_i(a_i(t), m_i(t)) = m_i(t) e_{0,i} + e_{c,i} \frac{\sum_{j=1}^{J} \lambda_{i,j}(t)}{\mu_i}$ is affine in $\mathbf{m}(t)$ and $\lambda(t)$. Thus, **P2** belongs to linear-fractional programming (and also quasiconvex programming) [8]. In what follows, we present an efficient iterative algorithm based on bisection method to solve **P2**. To begin with, we introduce another auxiliary variable $v \geq 0$, and define the following inequality

$$\sum_{i=1}^{N} [\epsilon_{i,d} \cdot p_i(a_i, m_i) + \epsilon_{i,id} \cdot z_i] \leq v \cdot \sum_{i=1}^{N} p_i(a_i, m_i). \tag{18}$$

Then, the bisection-based iterative method can be formally described in Algorithm 1, where $MaxNum$ is the maximum possible WUE and $\epsilon > 0$ is a small positive number governing the stopping criterion. During each iteration, a feasibility

Algorithm 1 GLB-WS

1: Input λ_j, r_i, $\epsilon_{i,d}$, $\epsilon_{i,id}$, and u_i, for $i = 1, 2, \cdots, N$
2: Initialize: $lb = 0$, $ub = MaxNum$, $v = \frac{lb+ub}{2}$
3: **while** $ub - lb > \epsilon$ **do**
4: Check if there exist $\mathbf{m}$, λ, and z that satisfy (18) and constraints (15)–(17); if "yes", then $ub = v$; else $lb = v$
5: $v = \frac{lb+ub}{2}$
6: **end while**

checking problem is solved, which is linear programming and hence easy to solve [8]. The final output of Algorithm 1 is a feasible decision satisfying (18) and constraints (15)–(17). Algorithm 1 requires exactly $\lceil \log_2(\frac{ub-lb}{\epsilon}) \rceil$ iterations, and the final WUE will be within $\epsilon > 0$ of the optimum. Thus, the total complexity of solving Algorithm 1 is quite affordable for data centers (even for very small $\epsilon > 0$), making it an appealing candidate for future geographic load balancing decisions.

V. Performance Evaluation

This section presents a trace-based simulation study to validate our analysis. We first present our data set and then compare GLB-WS with prior research

A. Data set

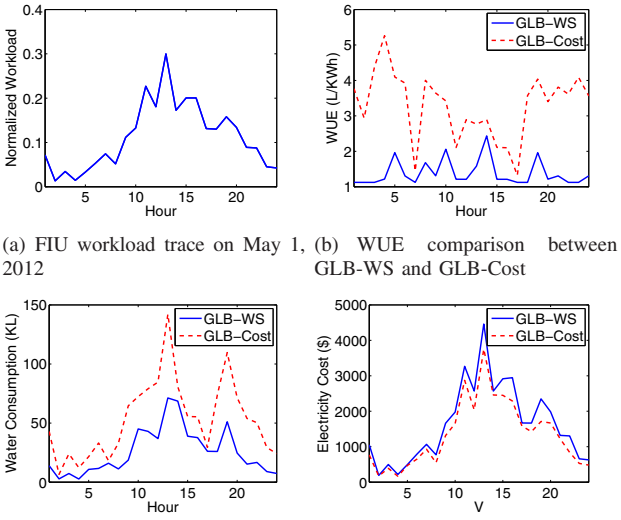
We consider four geographically distributed data centers located in: (#1) Prineville, OR, (#2) Northlake, IL, (#3) Forest City, NC, and (#4) Somerset, NJ. The four data centers have peak powers of 25MW, 18MW, 30MW and 20MW, respectively. For the convenience of illustration, the PUEs are chosen as 1.30 for all the data centers. In these four data centers, each server has a maximum power of 200W, 160W, 220W, and 250W, respectively, and static/idle server power takes up 60% of the maximum power. The normalized service rates of each server in the four data centers are chosen to be 1.00, 0.90, 1.25 and 1.10, respectively. All the workloads are distributed to the four data centers by the front-end gateway in St. Luis, MO, which has comparable distances to all the data centers. By default, the average response time constraint is 500ms (as in the case of web services [22]). The data center capacity provisioning and load distribution decisions are updated hourly.

• Workloads: We obtain a 24-hour workload trace by profiling the server usage log of Florida International University (FIU, a large public university in the U.S. with over 50,000 students) on May 1, 2012, and scale the FIU workload proportionally. Fig. 3(a) shows the trace normalized with respect to the total computing capacity of the four data centers. Other synthetic workloads are also tested and the results are similar.

• On-site renewable energy: We obtain from [3] four sets of the hourly renewable energies (generated through solar panels and wind turbines) on May 1 of 2012 from four locations that are closest to the data centers, and scale them proportionally such that on-site renewable energy supply takes up approximately 4% of the maximum energy demand in each data center on average.

• Electricity price: We obtain from [3] hourly electricity prices from four trading nodes closest to our considered data centers on May 1, 2012.

• Indirect EWIF and direct WUE for cooling system: Due to the lack of access to exact EWIF data in the four data center locations, we use the (time-varying) state-level EWIF values [24]. Since only Facebook has just started to report (hourly) direct WUE for its data center in Prineville, OR, we use the data presented in [6] for calculating the on-site water usage in data center #1. For the other three data centers, we use 1.62L/kWh (the same as eBay's direct WUE in May 29, 2013), 0.30L/kWh, and 1.00L/kWh, respectively, while randomly adding up to 20% noises to incorporate the temporal diversity.



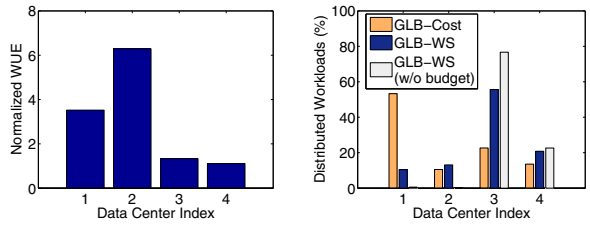
(a) FIU workload trace on May 1, 2012
(b) WUE comparison between GLB-WS and GLB-Cost
(c) Water consumption comparison between GLB-WS and GLB-Cost
(d) Electricity cost comparison between GLB-WS and GLB-Cost

Figure 3. FIU workload trace and performance comparison between GLB-WS and GLB-Cost.

B. Results

In this subsection, we compare GLB-WS with the state-of-the-art research using the above trace data.

GLB has been extensively studied for data center optimization from various perspectives: e.g., minimizing electricity cost [21], [22], maximizing renewable energy utilization [16], [18], [25], minimizing both electricity and delay cost [17], and capping long-term energy consumption [15]. While it is not possible to compare GLB-WS against all the existing GLB techniques, we choose GLB for electricity cost minimization [21], [22], called **GLB-Cost**, as our benchmark, as it is *one* of the most widely-considered GLB



(a) Normalized WUE in each data center

(b) Workload distribution under GLB-WS and GLB-Cost

Figure 4. Water efficiency and distributed workloads in each data center.

techniques. Comparison against other GLB techniques (e.g., those in [16], [18]) is similar and hence omitted for brevity.

Improved water efficiency. We first see from Fig. 3(b) that GLB-WS significantly improves the data center water efficiency compared to GLB-Cost: GLB-WS reduces the average WUE by nearly 60% than GLB-Cost. This is because GLB-WS can dynamically schedule workloads to water-efficient data centers, while GLB-Cost is water-oblivious and may "inappropriately" process more workloads in water-consuming (but cost-effective) data centers.

Reduced water consumption. While the explicit objective of GLB-WS is minimizing the WUE by scheduling workloads to water-efficient data centers, a direct byproduct of GLB-WS is the reduced water consumption. We see from Fig. 3(c) that, compared to GLB-Cost, GLB-WS reduces the water consumption remarkably (by over 51% on average), making data centers much more water efficient.

Comparable electricity cost and energy consumption. The focus of GLB-WS is not minimizing the electricity cost, but rather satisfying the given budget constraint. Thus, as can be seen from Fig. 3(d), GLB-WS incurs a higher electricity bill (by approximately $5 - 20\%$) compared to GLB-Cost that explicitly minimizes the electricity cost. While electricity cost is certainly important for data centers, we argue that sustainability, in particular water sustainability, is also critical and needs to be taken into consideration in the future design of data centers. Nonetheless, as water-efficient data centers is typically different from cost-effective ones (as can be seen from Fig. 4(b)), it may not be possible to optimize both metrics *simultaneously*, and an inherent tradeoff exists between (water) sustainability and data center operational cost, pointing to a potential research direction as our future work. Although not shown in the paper for brevity, we note that the average electricity energy consumptions by GLB-WS and GLB-Cost are almost the same (i.e., within 1% in our case study).

***Water-driven* scheduling.** Just as GLB-Cost is cost-driven and schedules workloads to cost-effective data centers [21], [22], GLB-WS is water-driven and can effectively schedule workloads to water-efficient data centers. We show in Fig. 4(a) the average *normalized* WUE (i.e., water usage per unit computing capacity): data centers #3 and #4 are water efficient (but cost inefficient), while data centers #1 and #2 are on the opposite side. Thus, it can be seen from Fig. 4(b) that GLB-WS can schedule more workloads to data centers #3 and #4, whereas GLB-Cost favors data centers #1 and #2 for processing workloads.[5] This intuition is further highlighted when we remove the electricity budget constraint for GLB-WS: as shown in Fig. 4(b), without considering budget constraint, GLB-WS will schedule almost all workloads to water-efficient data centers (i.e., #3 and #4).[6]

To sum up, GLB-WS focuses on a unique aspect of data center operation, i.e., water efficiency. It can effectively schedule workloads to water-efficient data centers to reduce the total water consumption, thereby improving the water efficiency. Nonetheless, except for water efficiency, we do not imply that GLB-WS outperforms all the existing GLB techniques in every other aspect (e.g., GLB-Cost minimizes the electricity cost, whereas the GLB in [16], [18] focuses on carbon footprint). Instead, we emphasize that GLB-WS is complementary to the existing research and that water sustainability deserves more attention from the research community.

VI. Related Work

We provide a snapshot of the related work from the following aspects.

Data center optimization. There has been a growing interest in optimizing data center operation from various perspectives such as cutting electricity bills [9], [14], [21], [22], minimizing brown energy consumption [13], [15], [18], and minimizing response times [12]. For example, "power proportionality" via dynamically turning on/off servers based on the workloads has been extensively studied and advocated as a promising approach to reducing the energy cost of data centers [14], [17], [22]. By exploring spatial diversities of electricity prices and/or energy "greenness", [21] study geographical load balancing among multiple data centers to minimize energy cost, [15] caps the long-term energy consumption based on predicted workloads, and [13], [18] propose to reduce brown energy usage by scheduling workloads to data centers with more green energies.

Water efficiency in data centers. To our best knowledge, there have been no research activities that explicitly optimize water efficiency in data centers. The only research works that are broadly relevant to data center water consumption are [7], [11], [23], which either point out the criticality of water conservation [11] or develop a dashboard for visualizing the water efficiency [7], [23], but no effective solutions have been proposed towards water sustainability in data centers. Publicly known efforts for water efficiency in data centers

[5]GLB-Cost schedules more workloads to data center #3 than to #2, because data center #3 has a much higher capacity.

[6]With a less stringent electricity budget constraint, the overall WUE will clearly be reduced using GLB-WS.

are mainly restricted to engineering approaches and include installing innovative cooling systems (e.g., outside air economizer), using recycled water, and powering data centers with on-site renewable energies to reduce the consumption of electricity (and hence, *indirect* water consumption, too) [2], [6]. These engineering methods are costly and orthogonal to our proposal: we are using a "*software*" approach that improves water efficiency from the perspective of workload management without substantial capital investments.

To sum up, our work takes the first step to rigorously address water sustainability in data centers. While our proposed GLB-WS may not possibly outperform all the existing GLB techniques in every aspect (e.g., GLB-WS incurs a higher electricity cost than GLB-Cost that particularly minimizes the cost), GLB-WS explicitly focuses on water efficiency that is becoming a critical concern in future data centers in light of the global water shortage trend. Moreover, our research on data center water efficiency provides an important, unique and complementary perspective to the existing data center research, and we take the liberty of envisioning that incorporating water efficiency is increasingly essential in future research efforts.

VII. Conclusion

In this paper, we made the first step towards water sustainability in data centers. We showed that data center WUE also exhibits spatial and temporal diversities. Leveraging these characteristics, we proposed GLB-WS, a GLB technique that can dynamically dispatch workloads to water-efficient data centers while satisfying the electricity cost and delay constraint. We also performed a trace-based simulation study to complement the analysis. The result was consistent with our analysis: compared to the state-of-the-art cost-minimizing GLB approach, GLB-WS significantly improves water efficiency and reduces water consumption. A natural future research direction is combining water efficiency with other metrics (e.g., carbon efficiency, electricity cost) in data center optimization.

References

[1] California ISO, http://www.caiso.com/.

[2] Google's data center efficiency.

[3] http://energy.gov/.

[4] eBay, http://dse.ebay.com.

[5] http://www.datacenterknowledge.com/archives/2009/04/09/.

[6] Facebook data center dashboard, http://www.fbpuewue.com.

[7] C. Bash, T. Cader, Y. Chen, D. Gmach, R. Kaufman, D. Milojicic, A. Shah, and P. Sharma. Cloud sustainability dashboard, dynamically assessing sustainability of data centers and clouds. *HP Labs Tech. Report (HPL-2011-148)*.

[8] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[9] N. Buchbinder, N. Jain, and I. Menache. Online job migration for reducing the electricity bill in the cloud. In *IFIP Networking*, 2011.

[10] N. Deng, C. Stewart, D. Gmach, M. Arlitt, and J. Kelley. Adaptive green hosting. In *ICAC*, 2012.

[11] E. Frachtenberg. Holistic datacenter design in the open compute project. *Computer*, 45(7):83–85, July 2012.

[12] A. Gandhi, M. Harchol-Balter, R. Das, and C. Lefurgy. Optimal power allocation in server farms. In *SIGMETRICS*, 2009.

[13] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav. It's not easy being green. *SIGCOMM Comput. Commun. Rev.*, 42(4):211–222, Aug. 2012.

[14] B. Guenter, N. Jain, and C. Williams. Managing cost, performance and reliability tradeoffs for energy-aware server provisioning. In *IEEE Infocom*, 2011.

[15] K. Le, R. Bianchini, T. D. Nguyen, O. Bilgir, and M. Martonosi. Capping the brown energy consumption of internet services at low cost. In *IGCC*, 2010.

[16] K. Le, R. Bianchini, J. Zhang, Y. Jaluria, J. Meng, and T. D. Nguyen. Reducing electricity cost through virtual machine placement in high performance computing clouds. SuperComputing, 2011.

[17] M. Lin, Z. Liu, A. Wierman, and L. L. H. Andrew. Online algorithms for geographical load balancing. In *IGCC*, 2012.

[18] Z. Liu, M. Lin, A. Wierman, S. H. Low, and L. L. Andrew. Greening geographical load balancing. In *SIGMETRICS*, 2011.

[19] U.S. Dept. of Energy. Energy demands on water resources. Dec. 2006.

[20] National Public Radio. Amid data controversy, NSA builds its biggest data farm, http://www.npr.org/2013/06/10/190160772/.

[21] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs. Cutting the electric bill for internet-scale systems. In *SIGCOMM*, 2009.

[22] L. Rao, X. Liu, L. Xie, and W. Liu. Reducing electricity cost: Optimization of distributed internet data centers in a multi-electricity-market environment. In *IEEE Infocom*, 2010.

[23] R. Sharma, A. Shah, C. Bash, T. Christian, and C. Patel. Water efficiency management in datacenters: Metrics and methodology. In *ISSST*, 2009.

[24] The Green Grid. Water usage effectiveness (WUE): A green grid data center sustainability metric. *Whitepaper*, 2011.

[25] Y. Zhang, Y. Wang, and X. Wang. Greenware: Greening cloud-scale data centers to maximize the use of renewable energy. In *Middleware*, 2011.