# GreenWare: Greening Cloud-Scale Data Centers to Maximize the Use of Renewable Energy

Yanwei Zhang[1], Yefu Wang[1], and Xiaorui Wang[1,2]

[1] Department of Electrical Engineering and Computer Science
University of Tennessee, Knoxville, TN 37996
[2] Department of Electrical and Computer Engineering
The Ohio State University, Columbus, OH 43210
{yzhang82,ywang38}@eecs.utk.edu, xwang@ece.osu.edu

**Abstract.** To reduce the negative environmental implications (e.g., $CO_2$ emission and global warming) caused by the rapidly increasing energy consumption, many Internet service operators have started taking various initiatives to operate their cloud-scale data centers with renewable energy. Unfortunately, due to the intermittent nature of renewable energy sources such as wind turbines and solar panels, currently renewable energy is often more expensive than brown energy that is produced with conventional fossil-based fuel. As a result, utilizing renewable energy may impose a considerable pressure on the sometimes stringent operation budgets of Internet service operators. Therefore, two key questions faced by many cloud-service operators are 1) how to dynamically distribute service requests among data centers in different geographical locations, based on the local weather conditions, to maximize the use of renewable energy, and 2) how to do that within their allowed operation budgets.

In this paper, we propose GreenWare, a novel middleware system that conducts dynamic request dispatching to maximize the percentage of renewable energy used to power a network of distributed data centers, subject to the desired cost budget of the Internet service operator. Our solution first explicitly models the intermittent generation of renewable energy, e.g., wind power and solar power, with respect to varying weather conditions in the geographical location of each data center. We then formulate the core objective of GreenWare as a constrained optimization problem and propose an efficient request dispatching algorithm based on linear-fractional programming (LFP). We evaluate GreenWare with real-world weather, electricity price, and workload traces. Our experimental results show that GreenWare can significantly increase the use of renewable energy in cloud-scale data centers without violating the desired cost budget, despite the intermittent supplies of renewable energy in different locations and time-varying electricity prices and workloads.

## 1 Introduction

Recent years have seen the rapid growth of large and geographically distributed data centers deployed by Internet service operators to support various services

F. Kon and A.-M. Kermarrec (Eds.): Middleware 2011, LNCS 7049, pp. 143–164, 2011.

such as cloud computing. As an effort to deal with the increasingly severe global energy crisis, reducing the high energy consumption of those cloud-scale data centers has become a serious challenge. For example, some cloud-service data centers are termed as *mega data centers*, because they host hundreds of thousands of servers and can draw tens to hundreds of megawatts of power at peak [22]. It has also been reported that in a conservative estimation, Google hosts more than 500,000 servers in its data centers distributed in different locations and consumes at least $6.3 \times 10^5$ MWh in total annually [39]. Therefore, minimizing the energy consumption of cloud-scale data centers has recently received a lot of research attention (e.g., [20,17,29,13,47,18]).

In addition to high electricity bills, the enormous energy consumption of cloud-scale data centers can also lead to negative environmental implications (*e.g.*, $CO_2$ emission and global warming), due to their large carbon footprints. The reason is that most of the produced electricity around the world comes from carbon-intensive approaches, *e.g.*, coal burning [29]. Such energy produced with conventional fossil-based fuel is commonly referred to as brown energy. Therefore, to mitigate the negative environmental implications caused by the rapidly increasing energy consumption, many Internet service operators have started taking various initiatives to operate their cloud-scale data centers with renewable (or *green*) energy. In contrast to brown energy, green (or clean) energy is normally generated from renewable energy sources, such as wind turbines and solar panels, and is thus more environmentally friendly. For example, major Internet service operators, *e.g.*, Google, Microsoft, and Yahoo!, have all started to increasingly power some of their data centers using renewable energy, and so reduce their dependence on brown energy [38,4,43]. Therefore, since data centers in different geographical locations may have different availabilities of renewable energy depending on the local weather conditions, it is important for cloud-service operators to dynamically distribute service requests among different data centers to maximize the use of renewable energy.

Unfortunately, due to the intermittent nature of renewable energy sources such as wind and sunlight, currently renewable energy can be often more expensive to produce than brown energy [2,11]. While some data centers are trying to build their own wind farms or solar photovoltaic (PV) power plants, due to concerns such as expensive facility investments and management, many Internet service operators choose to work with professional renewable energy producers and utilize the green energy integrated into the power grid. For example, Google has recently purchased 20 years' worth of wind energy from an Iowa wind farm, which will be sufficient to power several of its data centers in Oklahoma [12]. Google also invested $100 million in the Shepherds Flat Wind Farm in Oregon to generate 845 megawatts of green power, which will be sold directly to Southern California Edison's power grid. As a result of its higher production costs, renewable energy coming from the grid can be more expensive than brown energy. For example, the industrial electricity price for solar energy can be 16.14 cents per KWh in a sunny climate and 35.51 cents per KWh in a cloudy climate [8]. In contrast, the wholesale brown energy price can be around 6 cents per KWh [39]. The Los Angeles

Department of Water and Power also estimates that the extra cost for green energy is at least 3 cents per KWh [5]. Therefore, utilizing renewable energy may impose a considerable pressure on the sometimes stringent operation budgets of Internet service operators, as the electricity cost of operating data centers has become a significant portion, *e.g.*, 20% or more of the monthly costs of those enterprises [22]. Hence, a key dilemma faced by many service operators is how to exploit renewable energy to the maximum degree that is allowed by their monthly operating budgets.

In this paper, we propose *GreenWare*, a novel middleware system that conducts dynamic request dispatching to maximize the percentage of renewable energy used to power a network of distributed data centers, subject to the desired cost budgets of Internet service operators. We first model the intermittent generation of renewable energy, *i.e.*, wind power and solar power, with respect to the varying weather conditions in the geographical location of each data center. For example, the available wind power generated from wind turbines is modeled based on the ambient wind speed [35,9], while the available solar power from solar plants is estimated by modeling the maximum power point on irradiance (*i.e.*, solar energy per unit area of the solar panel's face) and temperature [31,41]. Based on the models, we formulate the core objective of GreenWare as a constrained optimization problem, in which the constraints capture the Quality of Service (QoS, *e.g.*, response time) requirements from customers, the intermittent availabilities of renewable energy in different locations, the peak power limit of each data center, and the monthly cost budget of the Internet service operator. We then transfer the optimization problem into a linear-fractional programming (LFP) formulation for an efficient request dispatching solution with a polynomial time average complexity.

Specifically, this paper makes the following major contributions:

- We propose a novel GreenWare middleware system in operating geographically distributed cloud-scale data centers. GreenWare dynamically dispatches incoming service requests among different data centers, based on the time-varying electricity prices and availabilities of renewable energy in their geographical locations, to maximize the use of renewable energy, while enforcing the monthly budget determined by the Internet service operator.
- We explicitly model renewable energy generation, *i.e.*, wind turbines and solar panels, with respect to the varying weather conditions in the geographical location of each data center. As a result, our solution can effectively handle the intermittent supplies of renewable energy.
- We formulate the core objective of GreenWare as a constrained optimization problem and propose an efficient request dispatching solution based on LFP.
- We evaluate GreenWare with real-world weather, electricity price, and workload traces. Our experimental results show that GreenWare can significantly reduce the dependence of cloud-scale data centers on fossil-fuel-based energy without violating the desired cost budget, despite the intermittent supplies of renewable energy and time-varying electricity prices and workloads.

The rest of the paper is organized as follows. Section 2 introduces the overall architecture of the proposed GreenWare middleware system. Section 3 presents the modeling and formulations of GreenWare. Section 4 discusses the simulation strategy. Section 5 evaluates GreenWare with real-world traces. Section 6 reviews the related work and Section 7 concludes the paper.

## 2 GreenWare Architecture

In this section, we provide a high-level description of the proposed GreenWare system. GreenWare dynamically conducts request dispatching among data centers in order to maximize the percentage of renewable energy used to power a network of distributed data centers, based on the time-varying electricity prices and availabilities of renewable energy in their geographical locations. In the meantime, GreenWare guarantees the desired QoS for customers and effectively maintains the electricity bill within a cost budget determined by the Internet service operators.

In this work, we assume that a network of distributed data centers share a common cost budget, which can be determined by the Internet service operator periodically in each budgeting period (*e.g.*, a month). A local optimizer is assumed to be present in each single data center in the network to dynamically adjust the number of active servers to minimize the power consumption of the data center, while maintaining a desired level of QoS based on a QoS model detailed in Section 3.2. We also assume that the short-term weather conditions (*e.g.*, in one hour) and the configurations of wind turbines and solar panels of each data center are available. As shown in Figure 1, *GreenWare* is a centralized system that manages a data center network for maximizing the use of renewable energy within the cost budget. While such a centralized architecture is commonly used in the management of data center networks [29,40,39], GreenWare can be extended to work in a hierarchical way, which is our future work. Similar to [13,34,48], we use one month as the budgeting period and one hour as the period for GreenWare to be invoked and conduct the request dispatching operation.

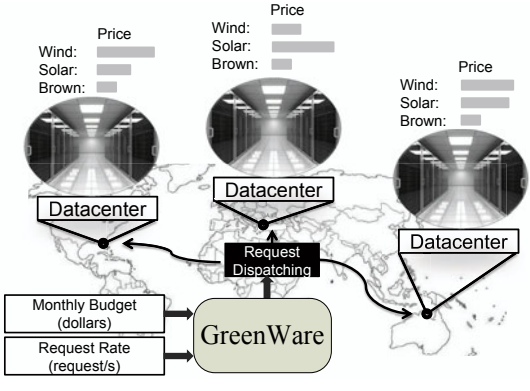


**Fig. 1.** Proposed GreenWare system for distributed cloud-scale data center networks

In every invocation period, GreenWare performs three steps: First, GreenWare computes the hourly budget based on the monthly cost budget from the service operator and the electricity cost already consumed in the previous invocation periods, as well as the observations of the workload's historical behaviors in the same hours in the past (*e.g.*, last two weeks) as discussed in Section 4.3. Second, based on the time-varying electricity prices and availability of renewable energy at each data center, with respect to the varying weather conditions in their geographical location (*e.g.*, irradiance, temperature, and wind speed), GreenWare runs the optimization algorithm in Section 3 to compute the desired request dispatching (*e.g.*, the fraction of workload allocated to each data center) such that (1) the overall percentage of renewable energy used to power a network of distributed data centers is maximized within budget constraints; (2) the total electricity cost is below the budget of the current hour; and (3) the application-level QoS (*e.g.*, desired response time) for customers is guaranteed. Third, GreenWare redirects the incoming requests among data centers based on the determined request dispatching in Step (2), using the dynamic request routing mechanism already deployed in cloud-scale data center networks. Note that dynamic request routing has already been implemented by many Internet service operators to map requests to servers, for the purposes of customer QoS guarantees and fault-tolerance [39].

## 3 Design Methodology of GreenWare

In this section, we first present the problem formulation of the optimization objective of GreenWare. We then introduce the adopted performance and server power models, as well as the wind power model and solar power model. Finally, we discuss our request dispatching solution. Note that we focus mainly on wind power and solar power in this work because there exists meteorological data [6] for us to simulate their intermittent availabilities in distributed data centers. GreenWare can be applied to other types of renewable energy, such as hydro-electric and geothermal, if their corresponding meteorological data is also available.

### 3.1 Problem Formulation

We first introduce the following notation. $N$ data centers are operated in a cloud-scale data-center network. The $i^{th}$ data center consumes $pW_i$ kilowatts of wind energy, $pS_i$ kilowatts of solar energy and $pB_i$ kilowatts of brown energy, respectively. The total power consumption $p_i$ (*i.e.*, $p_i = pW_i + pS_i + pB_i$) of the $i^{th}$ data center should not exceed the peak power limit $Ps_i$ of the data center. The intermittent availabilities of the renewable energy in the local power market of the $i^{th}$ data center are denoted as $PW_i$ and $PS_i$. In particular, $PW_i$ and $PS_i$ are the estimated wind power output from the wind farm and the maximum solar power output from the solar plant, respectively. The corresponding wind farm and solar plant are assumed to be the renewable energy sources for the local

power market of the $i^{th}$ data center. $PrW_i$, $PrW_i$ and $PrB_i$ are the current electricity prices of the three types of energy from the power market of the $i^{th}$ data center, respectively. The whole system has an incoming workload of $\lambda$ requests per hour. Our algorithm allocates the $i^{th}$ data center with a workload of $\lambda_i$ requests per hour to maximize the percentage of renewable energy used, depending on the wind and solar power models based on local weather conditions (presented in Sections 3.3 and 3.4), within the allocated cost budget $Cs$. The average response time of the $i^{th}$ data center is $R_i$ and the corresponding response time set point is $Rs_i$.

Given a workload of $\lambda$ requests per hour, the optimization goal is to dynamically choose a request dispatching strategy such that the $i^{th}$ data center is assigned $\lambda_i$ requests to maximally use renewable energy to power the data center network within the cost budget. Specifically, in order to maximize the overall renewable energy usage of all the $N$ data centers, $x_i$ percentage of wind power and $y_i$ percentage of solar power out of the total power consumption $p_i$ by the $i^{th}$ data center will have to be determined. Then, $z_i$ percentage of the total power consumption is supplemented in the form of brown energy. It is clear that $z_i = 1 - x_i - y_i$. In summary, the optimization problem can be expressed as follows.

**Problem 1:**

$$Maximize : \frac{\sum_{i=1}^{N} (pW_i + pS_i)}{\sum_{i=1}^{N} (pW_i + pS_i + pB_i)} \tag{1}$$

subject to

$$\sum_{i=1}^{N} \lambda_i = \lambda \tag{2}$$

$$\lambda_i \geq 0 \tag{3}$$

$$R_i \leq Rs_i \tag{4}$$

$$0 \leq pW_i \leq PW_i \tag{5}$$

$$0 \leq pS_i \leq PS_i \tag{6}$$

$$0 \leq pW_i + pS_i + pB_i \leq Ps_i \tag{7}$$

$$\sum_{i=1}^{N} (PrW_i \cdot pW_i + PrS_i \cdot pS_i + PrB_i \cdot pB_i) \leq Cs \tag{8}$$

Specifically, $pW_i$, $pS_i$, $pB_i$, $PW_i$, and $PS_i$ (in KW) will be numerically the same as energy (in KWh) since the invocation period used in this work is assumed to be one hour. In order to solve the optimization **Problem 1**, it is important to model the variables $pW_i$, $pS_i$ and $pB_i$ as functions of $\lambda_i$, $x_i$ and $y_i$. It is clear that

$$pW_i = x_i \cdot p_i; pS_i = y_i \cdot p_i; pB_i = z_i \cdot p_i \tag{9}$$

where $x_i + y_i + z_i = 1$.

Thus, in the following we first model the power consumption $p_i$ and the average response time $R_i$ with the request distribution rate $\lambda_i$ for the $i^{th}$ data center. We then model the availabilities of wind power and solar power, *i.e.*, $PW_i$ and $PS_i$, respectively, based on the weather condition of the $i^{th}$ data center, *e.g.*, irradiance, temperature, and wind velocity. We discuss an efficient solution design for **Problem 1** in Section 3.5.

### 3.2 Response Time and Power Models

Queueing theory is widely used to model the performance of a web server [13,14]. In this paper, we use the M/M/n queueing model in queueing theory [40] to model the response time for a data center. The average response time of the requests to a web server consists of two portions: (1) the average waiting time that the requests spend in a queue waiting to be serviced and (2) the service time, *i.e.*, $\frac{1}{\mu}$, given the service rate $\mu$ of the data center. Specifically, the average waiting time for a data center with $n$ active servers can be expressed as $\frac{1}{n \cdot \mu - \lambda} \cdot P_Q$, where $P_Q$ represents the probability that the incoming requests need to wait in a queue to be serviced. Furthermore, we assume that all the active servers will likely keep busy, *i.e.*, running at close to 100% utilization, because a local optimizer running in each data center minimizes the number of active servers. Hence, without loss of generality, $P_Q$ is assumed to be 1, since all the active servers are assumed to be running at close to 100% utilization. The same assumption is used in existing solutions on electricity cost minimization for data centers [40]. Therefore, we have

$$R_i = \frac{1}{\mu_i} + \frac{1}{n_i \cdot \mu_i - \lambda_i} \tag{10}$$

where $n_i$ is the number of active servers and $\mu_i$ is the average service rate of a single server, *i.e.*, the number of requests the server is able to process in a unit time, in the $i^{th}$ data center .

As discussed in Section 2, we assume that a local optimizer runs in every data center and dynamically adjusts the number of active servers to provide a desired level of QoS (*e.g.*, response time) with the least number of servers. As a result, given a request rate of $\lambda_i$ and a desired response time $Rs_i$ of the $i^{th}$ data center, the number of desired active servers $n_i$ can be derived from equation (10). Thus, we have $p_i = n_i \cdot sp_i$, where $sp_i$ is the average power consumption of a single server in the $i^{th}$ data center. Although the power consumption of a server is usually a function of the utilization of the server, we assume that $sp_i$ is constant because when the local optimizer minimizes the number of active servers, all the

servers remaining active will likely run close to a 100% utilization. Thus, the utilization will be approximately the same. It is then clear that a linear server power model based on the incoming work rate $\lambda_i$ for the $i^{th}$ data center can be derived, *i.e.*, $p_i = f(\lambda_i)$, where $f(\lambda_i)$ is a linear function.

### 3.3 Wind Power Model

The number of wind turbine installations is rapidly growing worldwide. It is expected that the US can get 20% of its electricity from wind energy by the year 2030 [25,37]. It has been shown that wind power generated by wind turbines in a wind farm can be modeled as a function of the actual wind speed [35,9]. For example, the wind power output $p_{wind}$ by a single wind turbine, with respect to a wind speed of $v$, can be approximated as follows

$$p_{wind} = \begin{cases} 0 & v < v_{in}, v > v_{out} \\ p_r \cdot \frac{v - v_{in}}{v_r - v_{in}} & v_{in} < v < v_r \\ p_r & v_r < v < v_{out} \end{cases}$$

where $v_r$, $p_r$ are the rated speed and power of the wind turbine and $v_{in}$, $v_{out}$ are cut-in and cut-out wind speeds. Specifically, the cut-in speed is the wind speed at which the turbine first starts to rotate and generate power, *e.g.*, a typical value between 3 and 4 meters per second; while the cut-out speed is employed by the braking system to bring the rotor to a standstill to eliminate the risk of damaging the turbine rotor due to the continuously rising wind speed, *e.g.*, a cut-out speed of usually around 25 meters per second.

In the case of a large-scale wind power generation farm, *e.g.*, one consisting of a large number $m_w$ of wind-turbines, the overall wind power output is estimated as the sum of the power output values sampled at different turbines for simplicity [21]

$$PW = \sum_{k=1}^{m_w} p_{wind}^k$$

where $p_{wind}^k$ is the power output from the $k^{th}$ wind turbine with respect to the wind speed $v$, with t05e assumption that the wind turbines have the same wind speed in the same wind farm.

### 3.4 Solar Power Model

The worldwide photovoltaic (PV) power capacity installation grows in a nearly exponential way, despite their relatively high cost [41]. In this work, we model the solar power generated by solar plants with respect to the varying weather conditions, such as irradiance and temperature, based on a single diode equation [41,36]. In particular, the single diode equation has been widely used to simulate the available electrical power generated from a single PV panel. Specifically, the resulting current-voltage characteristic of a PV panel is

$$i = I_{ph} - I_o \cdot (e^{\frac{v + i \cdot R_s}{n_s \cdot V_{th}}} - 1) - \frac{v + i \cdot R_s}{R_{sh}} \tag{11}$$

where $I_{ph}$ is the photo-generated current while $I_o$ is the dark saturation current with respect to the ambient weather pattern. Moreover, the single-diode model takes into account both the series and parallel (shunt) resistance of the PV panel, referred to as $R_s$ and $R_{sh}$, respectively. $V_{th}$ is the junction thermal voltage, *i.e.*, $V_{th} = k \cdot T/q$, where $k$ is Boltzmann's constant, $q$ is the charge of the electron and $T$ is the ambient temperature. $n_s$ is the number of the solar cells in the PV panel connected in series, *e.g.*, $n_s = 72$ in BP-MSX 120 panels [1].

To show the solar power output from PV panels with respect to the varying weather conditions (*e.g.*, irradiance and temperature), equation (11) can then be transformed as equation (12) by including these two key factors, *i.e.*, irradiance and temperature [41]. In particular, it has been demonstrated that the dark saturation current of $I_o$ just varies with the ambient temperature $T$, independent on the irradiance condition $G$ [41,16]. Furthermore, for a high-quality solar cell, it typically has a low series resistance $R_s$ but a high parallel resistance $R_{sh}$. As a result, the solar model in this work only takes into considerations the series resistance (*i.e.*, $R_{sh} = \infty$), which is consistent with the prior study [31]. We thus have the fact that $I_{ph}$ can be approximated by $I_{sc}$ for simplicity, where $I_{sc}$ is the short-circuit current. In particular, $I_{sc}$ is directly proportional to the irradiance as well as the ambient temperature. Thus, we have

$$i(G,T) = I_{sc}(G,T) - I_o(T) \cdot e^{\frac{v(G,T)+i(G,T)\cdot R_s}{n_s \cdot V_{th}}} \tag{12}$$

where $I_{sc}(G,T) = \frac{G}{G_0} \cdot I_{sc} \cdot (1 + \frac{k_i}{100} \cdot (T - T_0))$ and $I_o(T) = I_{sc} \cdot (1 + \frac{k_i}{100} \cdot (T - T_0)) \cdot e^{-\frac{V_{oc} + k_v \cdot (T - T_0)}{n_s \cdot V_{th}}}$. $G_0$ and $T_0$ are the respective irradiance level and temperature in Standard Test Conditions (STC), *i.e.*, $G_0 = 1000W/m^2$ and $T_0 = 25^oC$. $I_{sc}$, $V_{oc}$, $k_v$ and $k_i$ are the given parameters of short-circuit current, open-circuit voltage, temperature coefficients of the short-circuit and the open-circuit in STC from the datasheet of PV panels, respectively.

In particular, the solar power produced by a PV panel with respect to the varying weather conditions, based on the current-voltage characteristic shown as equation (11) is the product of the output voltage and current. Namely, $p_{solar} = v(G,T) \cdot i(G,T)$. It has been demonstrated that the power output $p_{solar}$ generated by a PV panel shows a unique maximum value under uniform irradiation and temperature [31,41]. In order to achieve the maximum efficiency of solar plants, some researchers have already put efforts in extracting the maximum power point from solar plants [19,27]. We thus estimate the solar power output by a PV panel as the maximal power value which can be extracted from the PV panel (referred to as *mpp*). Specifically, *mpp* is achieved with respect to an optimal load $r_{mp}$ and the corresponding current $i_{mp}$ [19], where $r_{mp} = R_s + \frac{n_s \cdot V_{th}}{I_{sc}(G,T) + I_o(T) - i_{mp}}$. Thus, $mpp = i_{mp}^2 \cdot r_{mp}$. The Lambert $W$-function method is then used to calculate the maximum power point *mpp* of the PV panel with respect to the varying weather conditions. We assume that there are $m_s$ PV panels installed in a large-scale solar plant. Thus, the overall solar power output by the solar plant is estimated as

$$PS = \sum_{k=1}^{m_s} mpp^k$$

where $mpp^k$ is the maximum power point from the $k^{th}$ PV panel with respect to the irradiance $G$ and temperature $T$.

### 3.5 Problem Solution

Based on the analysis above, the optimization **Problem 1** is a non-linear programming problem with both a non-linear objective function and non-linear constraints, with respect to decision variables of $\lambda_i$, $x_i$ and $y_i$. However, for a service operator, it is important to design an efficient solution in order to dynamically make decisions to green the data centers with acceptable runtime overheads. We thus transfer the non-linear optimization **Problem 1** into a well-studied linear-fractional programming formulation as in the form of **Problem 2**, which can be further transferred into a standard linear programming problem. Specifically, note that for the equations (9) with respect to $pW_i$, $pS_i$ and $pB_i$ as discussed in Section 3.1, we can alternatively assume that among the $\lambda_i$ requests serviced by the $i^{th}$ data center, $\lambda_i^W$, $\lambda_i^S$ and $\lambda_i^B$ requests are serviced with wind energy, solar energy and brown energy, respectively. Thus, we can limit the decision variables for the optimization **Problem 1** in (1 - 8) to only workload-related variables of $\lambda_i^W$, $\lambda_i^S$ and $\lambda_i^B$, instead of both workload-related variables (*i.e.*, $\lambda_i$) and percentage variables (*i.e.*, $x_i$ and $y_i$).

Since $\lambda_i = \lambda_i^W + \lambda_i^S + \lambda_i^B$, **Problem 1** in (1 - 8) can be further transferred as follows.

**Problem 2:**

$$Maximize: \frac{\sum_{i=1}^{N} f(\lambda_i^W + \lambda_i^S)}{\sum_{i=1}^{N} f(\lambda_i^W + \lambda_i^S + \lambda_i^B)} \tag{13}$$

subject to

$$\sum_{i=1}^{N} (\lambda_i^W + \lambda_i^S + \lambda_i^B) = \lambda \tag{14}$$

$$\lambda_i^W \geq 0 \tag{15}$$

$$\lambda_i^S \geq 0 \tag{16}$$

$$\lambda_i^B \geq 0 \tag{17}$$

$$R_i \leq Rs_i \tag{18}$$

$$0 \leq f(\lambda_i^W) \leq PW_i \tag{19}$$

$$0 \leq f(\lambda_i^S) \leq PS_i \tag{20}$$

$$0 \leq f(\lambda_i^W + \lambda_i^S + \lambda_i^B) \leq Ps_i \tag{21}$$

$$\sum_{i=1}^{N}(PrW_i \cdot f(\lambda_i^W) + PrS_i \cdot f(\lambda_i^S) + PrB_i \cdot f(\lambda_i^B)) \leq Cs \tag{22}$$

Specifically, $f(\lambda_i^W)$, $f(\lambda_i^S)$ and $f(\lambda_i^B)$ represent the amount of wind energy, solar energy and brown energy consumed in the $i^{th}$ data center, respectively. It is clear that $f(\lambda_i^W)$, $f(\lambda_i^S)$ and $f(\lambda_i^B)$ are linear functions as discussed in Section 3.2.

**Problem 2** is thus a specific case of linear-fractional programming problem with a fractional objective function and linear constraints. In order to solve the LFP-based optimization **Problem 2**, we leverage a standard technique discussed in [24] to transfer the problem in (13 - 22) to a linear programming problem. The detailed transformation is not shown due to space limitations, but the steps can be found in [24]. In our system, we implement the proposed GreenWare middleware system based on the *linprog* solver in Matlab. In particular, *linprog* uses an simplex method, which has been proven to have a low complexity in practice [7].

## 4 Simulation Setup

We aim to use realistic parameters in our experimental setup. We design a simulator and use real-world weather data, Web request traces, as well as electricity price data from utility companies to evaluate the proposed GreenWare system. As discussed, GreenWare dynamically conducts request dispatching to maximize the percentage of renewable energy used to power a network of distributed data centers within the cost budget determined by the Internet service operator. These evaluations primarily target web server-based applications, which provide the request-response type of web services. Specifically, the setup simulates an Internet-scale data center network such as Google's data centers within the US.

### 4.1 Datacenter Parameters

In our evaluation, we simulate a large system composed of four geographically distributed data centers for an Internet service operator (*e.g.*, Google). Accordingly, four different locations are assumed in the simulator, *i.e.*, *San Luis Valley* in *Colorado*, *Los Angeles* in *California*, *Oak Ridge* in *Tennessee* and *Lanai* in *Hawaii*, which are the locations whose meteorological data are available in [6].

The power consumption profile of each server in the same location is assumed to be approximately the same, which is usually true when homogeneous servers

and configurations are used in each data center [40,33]. Specifically, similar to a related study [32], the server configuration in each location is respectively assumed to be as follows: Data Center 1 (2.0 GHz AMD Athlon processor), Data Center 2 (1.2 GHz Intel Pentium 4630 processor), Data Center 3 (2.9 GHz Intel Pentium D950 processor), and Data Center 4 (2.7 GHz AMD Athlon processor). Their power consumption is assumed to be 88.88, 34.10, 149.19, and 141.28 Watts and their processing capacity coefficients are estimated as 500, 300, 725, and 675 requests per second, respectively.

### 4.2 Renewable Energy Availability

To emulate the intermittent availabilities of renewable energy in the locations of different data centers, *i.e.*, wind power and solar power, we use meteorological data from the Measurement and Instrumentation Data Center (MIDC) [6] of the National Renewable Energy Laboratory. A variety of meteorological data, including irradiances, temperature, and wind speed, is covered in those records from the MIDC. Moreover, prior studies have shown that the data from the MIDC is sufficiently accurate [31]. In particular, we use meteorological data from the four stations, *e.g.*, *Sun Spot One*, *Loyola Marymount University Rotating Shadowband Radiometer*, *Oak Ridge National Laboratory* and *La Ola Lanai*, since they have consistent time periods with available meteorological data, beginning from June 1st, 2010 to June 30th, 2010. We further assume that there are 200 turbines installed in each wind farm and 10,000 solar panels installed in each solar plant to provide renewable energy to the local power utilities of the 4 data centers. In particular, BP-MSX 120 panels produced by British Petroleum are assumed to be used in the solar plants [1].

Specifically, based on the power models discussed in Sections 3.3 and 3.4, as well as the varying weather conditions obtained from MIDC, the available renewable energy of all the 4 data centers throughout the entire simulated month is demonstrated in Figures 2 and 3. In particular, Figure 2 depicts the overall available wind energy of all the 4 data centers, while Figure 3 shows the overall available solar energy. As shown in these two figures, the available renewable energy shows a diurnal pattern. This is due to the fact that the local weather conditions have a nearly diurnal pattern.

### 4.3 Real-World Workload Traces

To build our workloads in the simulator, we use a trace of Internet traffic from Wikipedia.org [45]. In particular, we use this tracefile with 2-month long data, which contains 10% of user requests that arrived at Wikipedia between October 1st, 2007 and November 30th, 2007. Figure 4 shows the hourly behavior of user requests in October and November, 2007. As illustrated in the figure, the users' behavior shows a very clear weekly pattern in visiting the Wikipedia website. Specifically, we take the 1-month long Wikipedia trace of November as the incoming workload in the simulator while using the October trace data to work as the historical observations of the workload to predict hourly cost budgets.

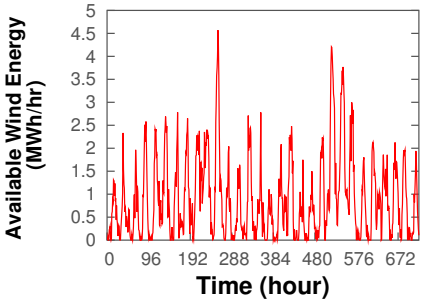


**Fig. 2.** The trace of available wind energy throughout the entire simulated month

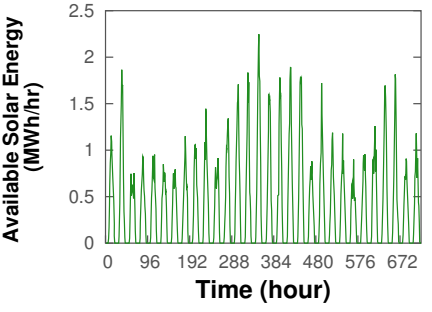


**Fig. 3.** The trace of available solar energy throughout the entire simulated month

To do so, we maintain a history of the request arrival rate seen during each hour of the week over the past several weeks. We then calculate every averaged hourly workload weight of the whole week over the past several weeks as the hourly budget weight in the coming week. Based on experiments, we find that for this Wikipedia trace, a 2-week long history trace data can provide a reasonable prediction on hourly cost budgets. Note that more sophisticated prediction methods, such as [46], can also be integrated into our system.

To make our evaluation more general, we also stress test GreenWare with another workload trace from the 1998 World Cup game, which includes the request data of 33 servers from 4 geographical locations. In particular, it records the incoming requests to all the servers with a granularity of 1 second from April 30th to July 26th, 1998.

### 4.4 Electricity Price Traces

To simulate the electricity price for the brown energy, we use the price trace from New York Independent System Operator (NYISO) [10], since they have complete and accurate price data records. Specifically, we use the Day-Ahead price data from November 1st, 2007 to November 30th, 2007, which is consistent with the dates of the Wikipedia traces. We apply the price data from the four

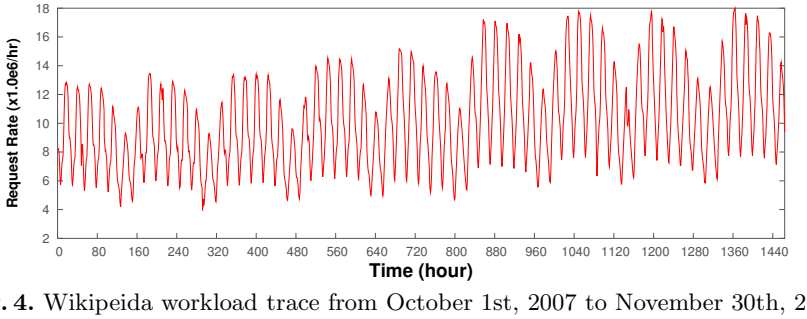


**Fig. 4.** Wikipeida workload trace from October 1st, 2007 to November 30th, 2007

zones, including Capital, Central, Dunwoodie and Genesee to the 4 data centers in our simulation.

On the other hand, regarding the electricity price of renewable energy, it is usually true that renewable energy has a higher electricity price compared to brown energy [2,5], due to the intermittent nature of renewable energy sources such as wind and sunlight, as well as expensive facility investments and management. For example, renewable energy costs an additional 1.5 cents per KWh compared to the regular energy in the power market of Virginia [2]. Furthermore, solar energy is typically much more expensive than wind energy, due to the relatively high capital expenses [3,11]. Thus, to be more practical, in our simulation we assume that the wind electricity price is 1.5 cents higher per KWh than brown energy [2]; while solar energy is 18.0 cents higher per KWh [3].

## 5 Evaluation Results

In this section, we first introduce two baselines. We then compare the proposed GreenWare middleware system against the baselines.

### 5.1 Baselines

In our work, we use two baselines in our experiments, a cost minimization only policy and a green energy usage maximization only policy, referred to as *Min-Cost* and *Max-Green*, respectively. (1) **Min-Cost.** Similar to GreenWare, Min-Cost also tries to minimize the electricity cost by distributing requests among geographically distributed data centers to leverage the varying electricity prices in different locations. However, different from GreenWare, Min-Cost is unaware of renewable energy and thus prefers brown energy in cost minimization. Min-Cost is similar to the state-of-the-art work [40] in minimizing the electricity bill in operating data center networks. (2) **Max-Green.** Similar to GreenWare, Max-Green tries to maximize the use of renewable energy by distributing more requests to data centers where more renewable energy is available. However, Max-Green does so regardless of the cost budget and thus may lead to a high operation cost for the Internet service operators and sometimes even budget violations. This scheme is similar to the state-of-the-art work [42] in powering data centers with renewable energy.

### 5.2 Impacts of the Monthly Cost Budget

In this experiment, we evaluate the proposed GreenWare middleware with respect to different monthly cost budgets.

Figures 5 and 6 depict how GreenWare works with the Wikipedia workload under a monthly cost budget of $340K. In particular, these two figures show that with a sufficient monthly cost budget (*e.g.*, as shown in Figure 5, the allocated hourly budget is sufficient throughout the entire month), brown energy is used only in the invocation periods with insufficiently available renewable energy.

That is, as indicated in Figure 6, only when the available renewable energy supply is less than the actual renewable energy demand (*i.e.*, a difference lower than 0), the corresponding renewable energy usage does not reach 100%, *e.g.*, the hours of 2, 5, 6, 7 and etc. Note that there are some invocation periods which have a zero usage of renewable energy, *e.g.*, the hours of 1, 3, 4 and etc. This is because that there is no available renewable energy at all due to the weather conditions in those invocation periods. In addition, Figure 5 demonstrates that the hourly allocated cost budget within one week shows a growing trend. This is due to the fact that we carry over the unused allocated cost budget from previous invocation periods to the remaining invocation periods in the same week.

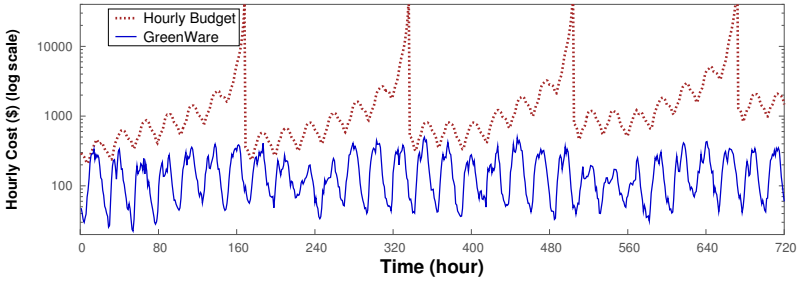


**Fig. 5.** Hourly electricity cost by GreenWare with a sufficient monthly cost budget of $340K, with respect to Nov. 2007 Wikipedia trace

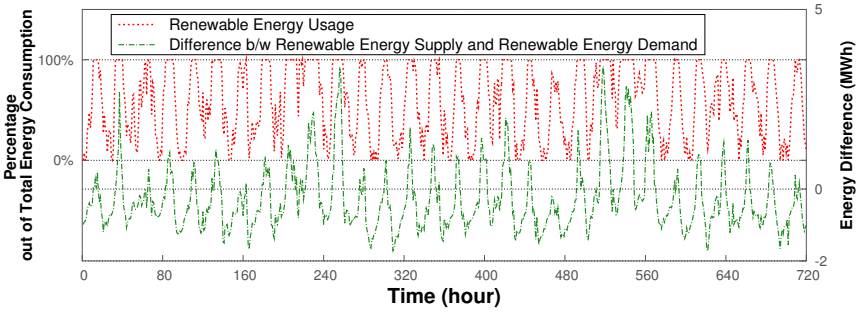


**Fig. 6.** Hourly renewable energy usage by GreenWare with a sufficient monthly cost budget of $340K, with respect to Nov. 2007 Wikipedia trace

We then study GreenWare under a series of different monthly cost budgets. As shown in Figure 7, with the increase of the monthly cost budget, the monthly average percentage of renewable energy usage keeps rising and then stays stable. This is due to the fact that fewer invocation periods are allocated with an insufficient cost budget in the case with a higher monthly cost budget. Therefore, more renewable energy can be used to power the data center networks. For example, with a monthly cost budget of $100K, there are 202 invocation periods

which have sufficient renewable energy supply but with an insufficient allocated cost budget; while as low as only 42 invocation periods are allocated with an insufficient cost budget in the case with a $160K monthly cost budget. As a result, a higher monthly average percentage of 58.17% of renewable energy usage is achieved with the monthly cost budget of $160K, compared to a percentage of 45.95% with the monthly budget $100K. Thus, when all the invocation periods have a sufficient budget due to a sufficient monthly cost budget, *e.g.*, $320K and $340K, the monthly average renewable energy usage stays stable. This set of experiments demonstrates that GreenWare can significantly increase the use of renewable energy in powering the data center network, subject to the desired cost budget.

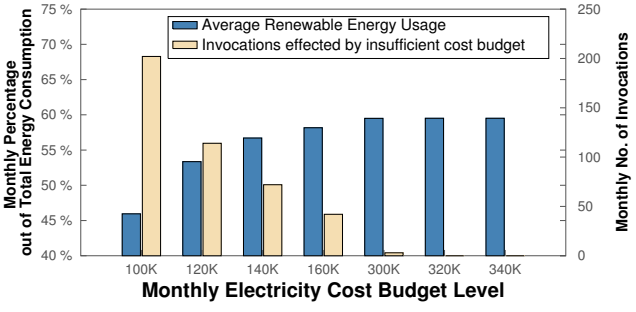


**Fig. 7.** Average percentage of renewable energy usage by GreenWare with a series of different monthly cost budgets

### 5.3 Comparison with Baselines

In this experiment, we compare GreenWare with the two baselines: Min-Cost and Max-Green.

Figure 8 depicts the cost and brown energy consumption of GreenWare, Max-Green and Min-Cost, with respect to a given monthly budget, *e.g.*, $100K, for the Wikipeida workload. The results are normalized against Min-Cost, which actually indicates the case of only using brown energy in powering data center networks. Figure 8 shows that although Max-Green (*i.e.*, maximizing the use of green energy regardless of cost budget) can decrease brown energy consumption by 58% compared to Min-Cost by utilizing as much renewable energy as possible. However, due to its unawareness of cost budget, Max-Green results in a 109% cost increase and exceeds the monthly cost budget by 29%. On the other hand, GreenWare can achieve an as-much-as-42% decrease in brown energy consumption at only a 52% cost increase, compared to Min-Cost. More importantly, GreenWare successfully controls the electricity bill to stay within the cost budget for the Internet service operator.

To demonstrate the effectiveness of GreenWare with different workloads, we also stress test GreenWare using the 1998 World Cup trace. Specifically, we use the request trace in June as the incoming workload in the simulation, and the

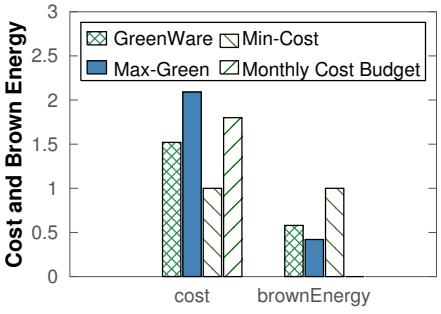


**Fig. 8.** Comparison between GreenWare and baselines with respect to Nov. 2007 Wikipeida trace

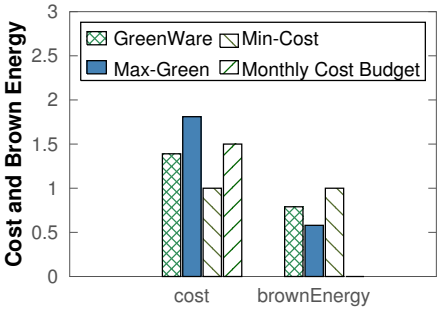


**Fig. 9.** Comparison between GreenWare and baselines with respect to Jun. 1998 World Cup trace

May trace as historical data to predict the hourly cost budget. To simulate the workload of cloud-service data centers, we proportionally increase the request numbers. Figure 9 shows the experiment results on the comparison between GreenWare and the two baselines. As demonstrated in the figure, Max-Green achieves a 42% decrease in brown energy consumption compared to Min-Cost. However, the electricity bill exceeds the given monthly cost budget (*e.g.*, $100K) by 31%. On the other hand, GreenWare obtains an as-much-as-21% decrease in brown energy consumption while successfully controlling the electricity bill to stay within the monthly cost budget.

### 5.4 Impacts of Pricing Policies of Renewable Energy

In this experiment, we show that the proposed GreenWare middleware always prefers the type of renewable energy that has a lower electricity price. Thus, an efficient cost minimization is guaranteed. Since in our work we just consider two types of the most popular renewable energy, *i.e.*, wind energy and solar energy, we assume two different pricing policies: (1) wind energy has a lower electricity price, as discussed in Section 4.4; and (2) solar energy has a lower price than wind energy. Note that the current practice is that wind energy is typically less expensive than solar energy. However, in order to stress test GreenWare, we assume a lower price for solar energy in (2).

Figures 10 and 11 demonstrate how the usage of different types of renewable energy varies with different pricing policies as discussed above. Intuitively, the more expensive renewable energy is taken into use only when the less expensive type of renewable energy is used up. As shown in Figure 10, with the first pricing policy (*i.e.*, wind energy price is lower), solar energy is used to power data centers only after all the supplied wind energy has been used up, as indicated in the second data center (DC#2). Similarly, with the second pricing policy, wind energy is used to power data centers only after all the available less expensive solar energy is consumed, as in all the data centers in Figure 11. Note that

in Figure 10, Data Centers 1, 3 and 4 begin to use the more expensive solar energy though there is still some wind energy left. This is because that there are some invocation periods when the available wind energy is too much to serve the incoming workload. As a result, some wind energy is left unused and the unused wind energy cannot be used in the following invocation periods due to the intermittent feature of the renewable energy.

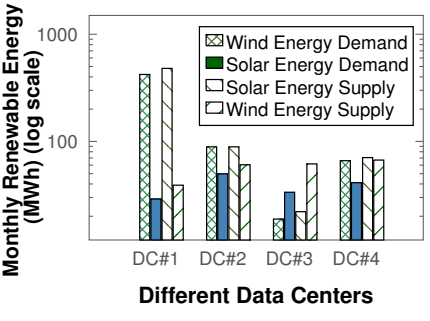


**Fig. 10.** Monthly renewable energy usage by GreenWare when wind energy price is lower than solar energy price

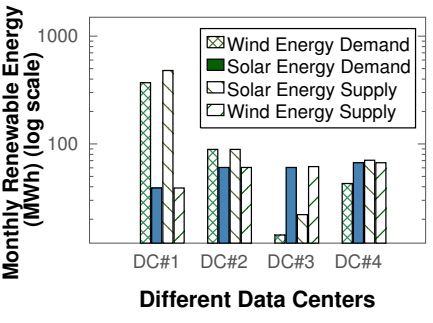


**Fig. 11.** Monthly renewable energy usage by GreenWare when solar energy price is lower than wind energy price

## 6 Related Work

Greening data centers is becoming an increasingly important topic in operating cloud-scale data center networks for Internet service operators, due to (1) data centers having become major energy consumers [44] and (2) the global energy crisis and environmental concerns (*e.g.*, global warming) [31]. To our best knowledge, our study is the first one that proposes to maximally use the renewable energy supplied by the local power utilities for Internet service operators, while being aware of the time-varying electricity price and enforcing a desired cost budget. Compared with the state of the art, the considerations of various realistic constraints make GreenWare more practical. We now discuss the related work.

**Energy conservation in data centers.** Many recent research projects have tried to minimize the energy consumption of data centers. For example, Chen et al. [18] and Chase et al. [17] reduce the energy consumption of connection servers hosting long-lived TCP-connection services and web servers providing request-response type of services, respectively. Heo et al. [23] have developed an adaptation graph analysis mechanism to solve the conflicts between interacting adaptive components, *e.g.*, *On/Off* and *dynamic voltage scaling* policies in server farms, to minimize energy consumption. Elnozahy et al. [20] investigate various combinations of dynamic voltage scaling and node on/off policies to reduce the energy consumption in server farms. Other strategies on reducing energy consumption of servers are also proposed (*e.g.*, [26,47]).

Our work differs from these efforts in several ways: (1) none of them try to use renewable energy to power data center networks; and (2) none of them put efforts on managing the electricity cost for the Internet service operators.

**Managing electricity cost in data centers.** A few recent projects have proposed to minimize the electricity bills of data center networks. For example, Qureshi et al. [39] try to lower the electricity bill by utilizing the varying electricity prices in different locations of distributed data centers. Rao et al. [40] consider a multi-electricity-market environment to reduce the electricity bill. In a recent study, Zhang et al. [48] propose an electricity bill capping algorithm to minimize the electricity cost within the cost budget for data center networks. Lin et al. [33] have tried to minimize the energy cost together with delay cost by rightly sizing data centers. Our work differs significantly from these efforts in that none of them try to maximize the use of renewable energy in powering data center networks for the Internet service operators.

**Utilizing renewable energy in data centers.** This is a relatively new topic with only few initial studies. Le et al. [29,28] propose to cap the consumption of brown energy while maintaining service level agreements (SLAs). Liu et al. [34] investigate how renewable energy can be used to lower the electricity price of brown energy in a specific power market, *i.e.*, where the brown energy is dynamically priced in proportion to the total brown energy consumption. Brown et al. [15] propose a simulation infrastructure to model a data center using renewable energy sources. In contrast to those studies, GreenWare aims to solve a related but different problem, *i.e.*, maximizing the use of renewable energy subject to the cost budget of the Internet service operators. Steward et al. [42] also try to maximize the use of renewable energy in data centers. However, their study assumes that Internet service operators have their own wind farms or solar plants. In contrast, GreenWare considers a different case where the service operators buy renewable energy from the power grid, which is a more common case for many data centers because of concerns such as expensive facility investments and management. In addition, their study does not consider the extra cost of renewable energy and may lead to budget violations as shown in the comparisons between GreenWare and Max-Green in Section 5.3. Li et al. [30] propose a load power tuning scheme for managing intermittent renewable power in a single data center without considering the costs. In contrast, we focus on distributing requests among data centers in different locations.

## 7 Conclusion

Two key questions faced by many cloud-service operators are 1) how to dynamically distribute service requests among data centers in different geographical locations, based on the local weather conditions, to maximize the use of renewable energy, and 2) how to do so within their allowed operation budgets. In this paper, we have presented GreenWare, a novel middleware system that conducts dynamic request dispatching to maximize the percentage of renewable energy

used to power a network of distributed data centers, subject to the desired cost budget of the Internet service operators. Our solution first explicitly models the intermittent generation of renewable energy, e.g., wind power and solar power, with respect to varying weather conditions in the geographical location of each data center. We then formulate the core objective of GreenWare as a constrained optimization problem and propose an efficient request dispatching algorithm based on linear-fractional programming (LFP). We evaluate GreenWare with real-world weather, electricity price, and workload traces. Our experimental results show that GreenWare can significantly increase the use of renewable energy in cloud-scale data centers without violating the desired cost budget, despite the intermittent supplies of renewable energy in different locations and time-varying electricity prices and workloads.

**Acknowledgments.** This work was supported, in part, by NSF under CNS-0720663 and CAREER Award CNS-0845390.

## References

1. BP MSX 120 solar module, http://pdf.directindustry.com/pdf/bp-solar/bp-msx-120-solar-module/15873-68158.html
2. Dominion Virginia Power, http://www.dom.com/
3. Energy modality comparison based on projected cents per kilowatt-hour, http://peswiki.com/
4. Green House Data: Greening the data center, http://www.greenhousedata.com/
5. Los Angeles Department of Water & Power, http://www.ladwp.com/
6. Measurement and Instrumentation data center, http://www.nrel.gov/midc/
7. The Running Time of the Simplex Method, http://www.mpi-inf.mpg.de
8. Solar Electricity Prices, http://solarbuzz.com/
9. WindPower Program, http://www.wind-power-program.com/index.htm
10. NYISO (1999), http://www.nyiso.com/
11. Solar Power at Data Center Scale (2009), http://www.datacenterknowledge.com/
12. Google Buys 20 Years' Worth of Wind Energy To Power Data centers (2010), http://www.huffingtonpost.com/
13. Ahmad, F., Vijaykumar, T.N.: Joint optimization of idle and cooling power in data centers while maintaining response time. In: ASPLOS (2010)
14. Bolch, G., Greiner, S., de Meer, H., Trivedi, K.S.: Queueing Networks and Markov Chains. Wiley Interscience (2005)
15. Brown, M., Renau, J.: Rerack: Power simulation for data centers with renewable energy generation. In: GreenMetrics (2011)
16. Castaner, L., Silvestre, S.: Modelling Photovoltaic Systems Using PSpice. John Wiley & Sons (2002)
17. Chase, J.S., Anderson, D.C., Thakar, P.N., Vahdat, A.M., Doyle, R.P.: Managing energy and server resources in hosting centers. In: SOSP (2001)
18. Chen, G., He, W., Liu, J., Nath, S., Rigas, L., Xiao, L., Zhao, F.: Energy-aware server provisioning and load dispatching for connection-intensive internet services. In: NSDI (2008)

19. Ding, J., Radhakrishnan, R.: A new method to determine the optimum load of a real solar cell using the lambert w-function. Solar Energy Materials and Solar Cell (2008)
20. Elnozahy, E.N.M., Kistler, J.J., Rajamony, R.: Energy-Efficient Server Clusters. In: Falsafi, B., VijayKumar, T.N. (eds.) PACS 2002. LNCS, vol. 2325, pp. 179–196. Springer, Heidelberg (2003)
21. Kariniotakis, G.N., Stavrakakis, G.S., Nogaret, E.F.: Wind power forecasting using advanced neural networks models. IEEE Transactions on Energy Conversion, 762–767 (1996)
22. Greenberg, A., Hamilton, J., Maltz, D.A., Patel, P.: The cost of a cloud: research problems in data center networks. ACM SIGCOMM Computer Communication Review (2008)
23. Heo, J., Henriksson, D., Liu, X., Abdelzaher, T.: Integrating adaptive components: An emerging challenge in performance-adaptive systems and a server farm case-study. In: RTSS (2007)
24. Hillier, F.S., Lieberman, G.J.: Introduction to Operations Research. McGraw-Hill (2005)
25. Hohl, A.: Wind Power for Data Centers (2009), `http://www.renewableenergyworld.com/rea/blog/post/2009/08/wind-power-for-data-centers`
26. Horvath, T., Abdelzaher, T., Skadron, K., Liu, X.: Dynamic voltage scaling in multitier web servers with end-to-end delay control. IEEE Transactions on Computers, 444–458 (2007)
27. Kothari, L.S., Mathur, P.C., Kapoor, A., Saxena, P., Sharma, R.P.: Determination of optimum load for a solar cell. Journal of Applied Physics, 5982–5984 (2009)
28. Le, K., Bianchini, R., Martonosi, M., Nguyen, T.D.: Cost- and energy-aware load distribution across data centers. In: HOTPOWER (2009)
29. Le, K., Bilgir, O., Bianchini, R., Martonosi, M., Nguyen, T.D.: Managing the cost, energy consumption, and carbon footprint of internet services. In: SIGMETRICS (2010)
30. Li, C., Qouneh, A., Li, T.: Characterizing and analyzing renewable energy driven data centers. In: SIGMETRICS (2011)
31. Li, C., Zhang, W., Cho, C.B., Li, T.: Solarcore: Solar energy driven multi-core architecture power management. In: HPCA (2011)
32. Li, J., Li, Z., Ren, K., Liu, X., Su, H.: Towards optimal electric demand management for internet data centers. In: Techreport (2010)
33. Lin, M., Wierman, A., Andrew, L.L.H., Thereska, E.: Dynamic right-sizing for power-proportional data centers. In: INFOCOM (2011)
34. Liu, Z., Lin, M., Wierman, A., Low, S.H., Andrew, L.L.H.: Greening geograhpical load balancing. In: SIGMETRICS (2011)
35. Patel, M.R.: Power systems: Design, Analysis, and Operation. CRC Press (2006)
36. Paukshto, M.V., Lovetskiy, K.: Invariance of single diode equation and its application. In: PVSC (2008)
37. Petru, T., Thiringer, T.: Modeling of wind turbines for power system studies. IEEE Transactions on Power Systems, 1132–1139 (2002)
38. Pistoia, G.: Battery Operated Devices and Systems: From Portable Electronics to Industrial Products. Elsevier (2011)
39. Qureshi, A., Weber, R., Balakrishnan, H., Guttag, J., Maggs, B.: Cutting the electric bill for internet-scale systems. In: SIGCOMM (2009)

40. Rao, L., Liu, X., Xie, L., Liu, W.: Minimizing electricity cost: optimization of distributed internet data centers in a multi-electricity-market environment. In: INFOCOM (2010)
41. Sera, D., Teodorescu, R., Rodriguez, P.: PV panel model based on datasheet values. In: ISIE (2007)
42. Stewart, C., Shen, K.: Some joules are more precious than others: Managing renewable energy in the datacenter. In: HOTPOWER (2009)
43. Thibodeau, P.: Wind power data center project planned in urban area (2008), http://www.computerworld.com/
44. United states environmental protection agency. Report to congress on server and data center energy efficiency (2007)
45. Urdaneta, G., Pierre, G., van Steen, M.: Wikipedia workload analysis for decentralized hosting. Elsevier Computer Networks 53(11), 1830–1845 (2009), http://www.globule.org/publi/WWADH_comnet2009.html
46. Urgaonkar, B., Shenoy, P., Chandra, A., Goyal, P.: Dynamic provisioning of multitier internet applications. In: ICAC (2005)
47. Verma, A., Dasgupta, G., Nayak, T.K., De, P., Kothari, R.: Server workload analysis for power minimization using consolidation. In: USENIX ATC (2009)
48. Zhang, Y., Wang, Y., Wang, X.: Capping the electricity cost of cloud-scale data centers with impacts on power markets. In: HPDC (2011)