CHAPTER

11 Energy-Efficient Cloud Computing: A Green Migration of Traditional IT

Hussein T. Mouftah, Burak Kantarci
School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON, Canada

11.1 INTRODUCTION

Over the last decade, information technology (IT) has introduced tremendous advances for the purpose of high performance data processing, data storage, and high speed wired/wireless communications in the Internet. Due to the limitations of local resources and power-efficiency concerns, distributed systems have emerged as feasible solutions each of which aims at a specific target. For instance, IBM's *autonomic computing* aims at building self-configuring, self-healing, self-optimizing, and self-protecting distributed computer systems for complex and unpredictable computing environments [1]. As another distributed system, *grid computing* comprises a group of computers located at physically distant locations and sharing computing and storage resources to accomplish a specific task on an on-demand basis [2]. Client–server and peer-to-peer networking are other distributed systems where the former denotes a distributed system consisting of a high performance server and lower performance clients, and the latter defines a distributed system consisting of peers that share hardware resources such as storage, processing power, and network link capacity [3]. In parallel to the distributed computing services *utility computing* has been proposed as another business model that delivers the computing resources among several computers based on an on-demand basis and bills the computing facilities such as in electricity and Internet billing [4].

The advent of distributed systems have eventually evolved to the *cloud computing* concept where almost infinite computing resources are available on-demand as well as the dynamic provisioning, release, and billing of computing facilities. Furthermore, capital/operational expenditures of the service providers are being reduced [5]. In fact, as stated by Zhang et al., cloud computing is not a new technology but it is a novel business model that brings several distributed system concepts together [6]. Therefore, several definitions of cloud computing exist [7] while most of the literature relies on the definition of the National Institute of Standards Technology (NIST). According to the NIST, cloud computing denotes a shared pool of resources available

Handbook of Green Information and Communication Systems. http://dx.doi.org/10.1016/B978-0-12-415844-3.00011-5

to the users, which can dynamically be provisioned and released without interacting with the service provider [8].

Grid computing seems to be the closest paradigm to cloud computing since it offers a distributed powerful and cost-efficient computing platform over the Internet. However, cloud computing offers a homogeneous pool of resources to the users as opposed to the heterogeneity of grid computing. Furthermore, cloud computing also differs from grid computing by the high reliability assurance of the service providers [9]. For instance, Amazon-EC2 assures 99.995% reliability, which leads to 27 min or less downtime per year [10].

Figure 11.1 illustrates an overview of the cloud computing architecture. As seen in the figure, four layers, namely the application, platform, infrastructure, and hardware layers form the *cloud*. The above two layers provide Software as a Service (SaaS) and Platform as a Service (PaaS), respectively, while the two bottom layers provide

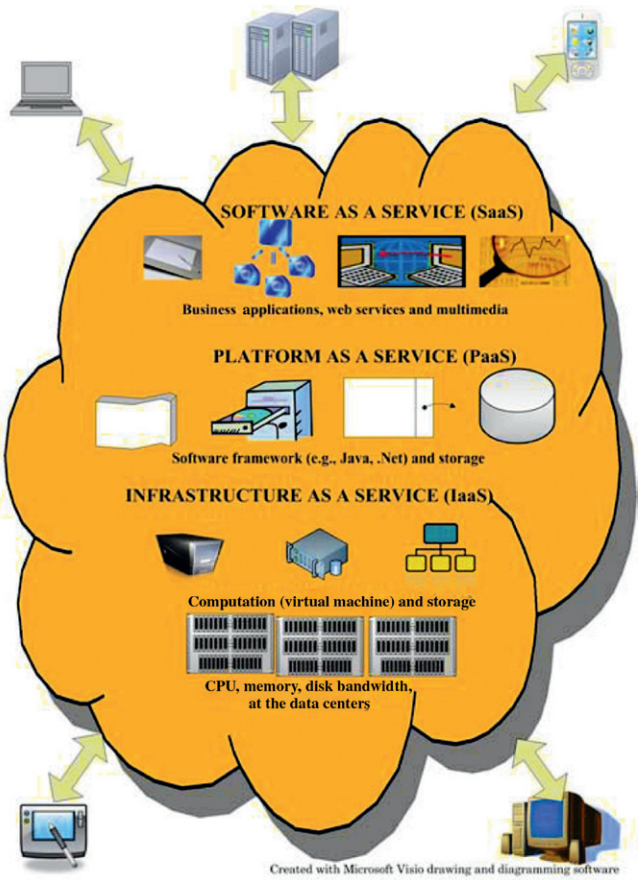

Figure 11.1 Cloud computing infrastructure of business enterprises.

Infrastructure as a Service (IaaS) to the end users. Hayes defines cloud computing as basically a shift in the geography of computation [11] and reports the attractive features of cloud computing along with the challenges. For instance, users are able to use the word processors offered by Google Docs or the Adobe's Buzzword without installing any word processors in their local computers. Besides, enterprise computing applications such as marketing and customer relation services can run on corporate servers without installation of any software on the local computers. Storage in the cloud is also another service offered to the users in terms of IaaS, e.g., Amazon Web Services, Google App Engine, IBM Smart Business Storage Cloud, etc.

Four deployment models of cloud computing exists as follows: (i) *Public clouds* serve through the Internet backbone and operate in a pay-as-you-go fashion (see Figure 11.2a). (ii) *Private clouds* are dedicated to an organization where the files and tasks are hosted within the corresponding organization. Thus, the users from various departments of the organization can access the cloud through a corporate network which is connected to the data center network through a gateway router (see Figure 11.2b). (iii) *Community clouds* enable several organizations to access a shared pool of cloud services forming a community of a special interest, and (iv) *hybrid clouds* are a combination of the public, private, and the community clouds with the objective of overcoming the limitations of each model [12].

In [6,11] several challenges in cloud computing have are discussed including security, privacy, reliability, virtual machine migration, automated service provisioning, and energy management. For the last decade, information and communication

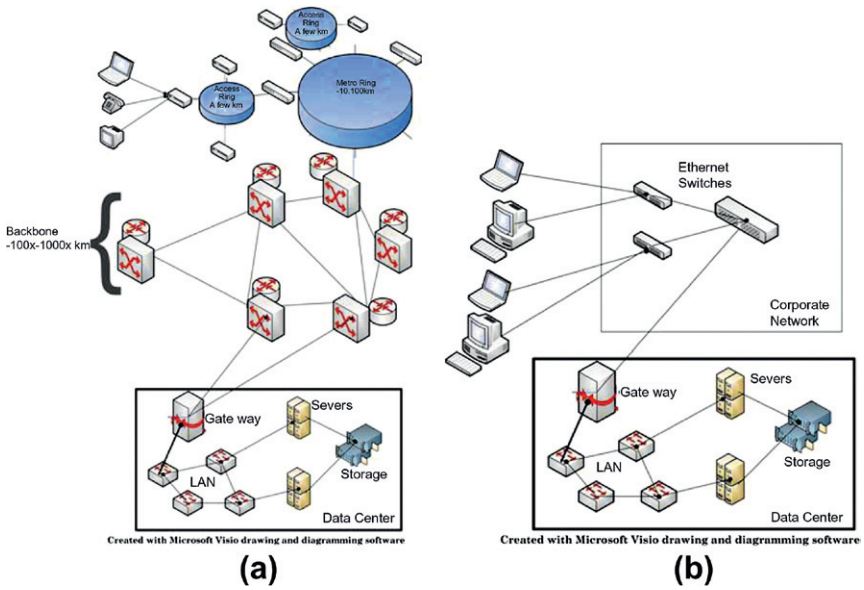

Figure 11.2 Fundamental deployment models of cloud computing: (a) public cloud, (b) private cloud.

technologies (ICTs) have been experiencing the energy bottleneck problem due to the exponential growth of the power consumption in the Internet [13], and improving the energy-efficiency in the Internet has been an emerging challenge [14]. As a part of the ICTs, processing, storage, and transmission of the cloud services contribute to the energy consumption of the Internet significantly, and energy-efficient design with respect to power consumption of the network equipment, service types, demographics, and service level agreements is emergent. For instance, power consumption of an IPTV service has three components such as storage power of the video, processing power consumed by the video servers, and the transmission power consumed by the transport network. Storage power is independent of the download demands however as the number of download demands per hour increases, power consumption of the video servers and the transport medium increases exponentially where the dominating factor is the transmission power. In [15], it has been shown that, for a single standard definition video replicated in 20 data centers, the total power consumption moves towards hundreds of kilowatts as the download request rate approaches 100 demands per hour.

This chapter focuses on the energy management challenges in cloud computing, which mainly have three aspects, namely processing, storage, and transport [16]. The chapter is organized as follows: in Section 11.2, we explain the relation between green ICT and energy-efficient cloud computing. We present the energy management in the data centers in terms of energy-efficient processing and storage in Section 11.3. Section 11.4 studies the optimal data center placement problem in order to ensure efficient utilization of renewable energy resources. Section 11.5 presents the studies to minimize the power consumption of transport services in cloud computing. Finally, the conclusion, research challenges, and further discussions are presented in Section 11.6.

11.2 GREEN ICT AND ENERGY-EFFICIENT CLOUD COMPUTING

Since the mid-2000s, energy consumption of the Internet-based applications has become a concern due to the rise of greenhouse gas (GHG) emissions; hence power-saving architectures and protocols have started being considered [17]. As of 2009, ICTs consume 4% of the global electricity consumption, and it is expected to double in the next few years [18]. Zhang et al. have presented a comprehensive survey on energy-efficiency schemes in various parts of the telecommunication networks [19]. Energy-management is mainly based on turning off the unused components such as line cards, network interfaces, router, and switch ports or putting these components in *low-power* (*sleep*) mode when they are idle. However, energy-efficient design and planning is also emergent for energy-efficient ICTs and it must be complemented by power-saving hardware and architectures [20].

According to a report in 2009, contributions of IT equipment to global GHG emissions are as follows: PCs and monitors contribute 40%, servers and their cooling equipment 23%, fixed line telecommunication equipments 15%, mobile telecommunications equipment 9%, local area and office network devices 7%, and

printers 6% [18]. Based on this distribution, moving the IT services to distant servers can lead to significant energy savings in PCs and monitors. In this sense, cloud computing seems to provide a solution to the green IT objective. However, migration of services to distant servers in remote data centers will have two main impacts on global energy consumption and consequently GHG emissions: (i) utilization of the data center servers will increase, and (ii) migrating the resources over the Internet will overload the Internet backbone traffic leading to increased power consumption of routing and switching equipments. Hence, as Baliga et al. have stated, energy-efficiency in cloud computing needs to maintain a balance between processing, storage, and transport energy [16].

11.2.1 Motivation for Green Data Centers

Figure 11.3 illustrates a sample data center layout and the associated thermal activity. As seen in the figure, a data center consists of several rack towers where each rack contains servers, switches, or storage devices [21]. In order to enable the transport of cool air supplied by the computer room air conditioner (CRAC) into the aisles of the data center, the rack towers are placed on a raised floor in either a back-facing or front-facing manner. Cool air is supplied into the aisles through the vents towards the front faces of the assets in the racks forming cool aisles. Assets in the racks emit the hot air through back faces, which is channeled to the CRAC. As stated in [21], this layout design is cost-efficient however it may lead to hot spots in the data center since hot and cold air is mixed, and cooling is not uniformly distributed in the data center. Hence, cooling power may even exceed the IT equipment power. In the absence of an energy-efficiency design, a typical data center consumes 2000–3000 kWh/(m^2year) leading to high operational and capital expenditures [22]. According to the report of the Energy Star program in 2007 [23], data centers consumed 61 billion kWh of energy in 2006, which contributed to 1.7–2.2% of the total electricity consumption in the U.S., and it was expected to double in the next year. The worldwide picture is similar to that in the U.S., where 1.1–1.5% of the global electricity consumption is contributed by the worldwide data centers. A significant portion of this energy consumption is due to the supporting systems such as lighting, uninterrupted power

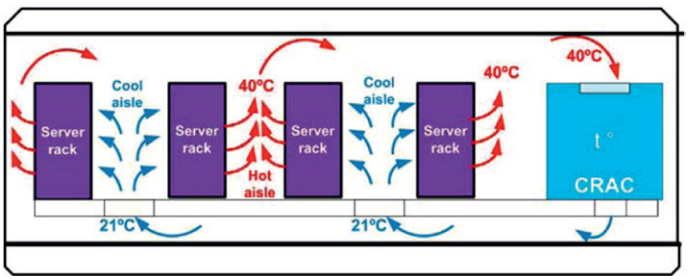

Figure 11.3 Layout design and the associated thermal activity in a typical data center.

supply (UPS), and heating, ventilation, and air conditioning (HVAC) units in the data centers. Thus, energy savings are possible if the power consumption of the supporting systems can be reduced.

Two main metrics are used to evaluate the energy-efficiency of a data center. The first metric is the ratio of the power consumption of the HVAC (i.e., CRAC) equipment to the power consumption of the IT equipment as shown in Eq. (11.1); thus, it denotes the cooling efficiency in the data center, i.e., *data center efficiency (DCE)*. A reasonable DCE value for a data center is 0.5 while a good target is reported to be above 0.625 [24]. The second metric evaluates the ratio of the power consumed by the IT equipment to the total power consumption in the data center as shown in Eq. (11.2); thus, it denotes the overall energy-efficiency in the data center, i.e., *power usage efficiency (PUE)*[25]. A typical PUE value for a data center is 2 however, today companies like Google, Amazon, and Microsoft have been able to build data centers with PUE values below 1.3 [26]:

$$\text{DCE} = \frac{P_{\text{HVAC}}}{P_{\text{IT Equipment}}} \tag{11.1}$$

$$\text{PUE} = \frac{P_{\text{IT Equipment}}}{P_{\text{total}}} \tag{11.2}$$

Figure 11.4 illustrates the data center efficiency report of Google where quarterly and 12-months average PUE of all of its data centers are shown. According to the power efficiency report of Google, by the end of the first half of 2011, 12-month and quarter-based average PUE values have been reduced to 1.15 and 1.14, respectively. It has been reported that reduced PUE can be achieved by joint use of the following three methods: thermal management, power-efficient

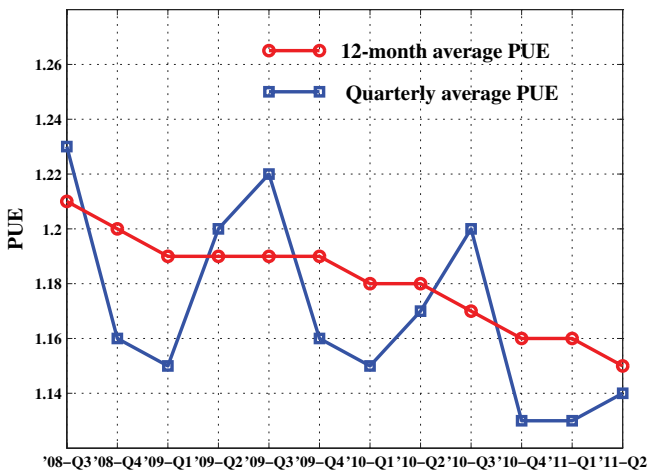

Figure 11.4 Quarterly and 12-month average PUE of all Google data centers [27].

hardware in servers together with optimized power distribution among the servers, and deployment of free (without chillers) cooling methods such as evaporating water [27].

11.2.2 Motivation for Green ICT

Baliga et al. have formulated the energy to transmit one bit from a data center to the user through a corporate network (i.e., private clouds) and energy to transmit one bit from a data center to a user through the Internet (i.e., public clouds) [16]. The notation used in the formulation is as follows:

F_{overhead}	Overhead factor for redundancy, cooling, and other factors
F_{util}	Overhead for underutilization of the core routers
P_{ses}	Power consumption of small Ethernet switches
C_{es}	Capacity (bps) of small Ethernet switches
P_{es}	Power consumption of Ethernet switches
C_{es}	Capacity (bps) of Ethernet switches
P_{bg}	Power consumption of border gateway routers
C_{bg}	Capacity (bps) of border gateway routers
P_{g}	Power consumption of data center gateway routers
C_{g}	Capacity (bps) of data center gateway routers
P_{ed}	Power consumption of edge routers
C_{ed}	Capacity (bps) of edge routers
P_{c}	Power consumption of core routers
C_{c}	Capacity (bps) of core routers
O_{c}	Overprovisioning factor for core routers
H_{c}	Hop count factor for the core traffic
P_{w}	Power consumption of wavelength division multiplexing (WDM) equipment
C_{w}	Capacity (bps) of WDM equipment

$$E_{\text{public}} = F_{\text{overhead}} \cdot F_{\text{util}} \cdot \left(\frac{3 \cdot P_{\text{es}}}{C_{\text{es}}} + \frac{P_{\text{bg}}}{C_{\text{bg}}} + \frac{P_{\text{g}}}{C_{\text{g}}} + \frac{2 \cdot P_{\text{ed}}}{C_{\text{ed}}} + \frac{O_{\text{c}} \cdot H_{\text{c}} \cdot P_{\text{c}}}{C_{\text{c}}} + \frac{(H_c - 1) \cdot P_{\text{w}}}{2 \cdot C_{\text{w}}} \right). \tag{11.3}$$

Equation (11.3) uses the layout in Figure 11.2a to formulate the energy to transmit one bit from a data center to a user over the Internet (E_{public}). In the equation, power consumption of Ethernet switches (P_{es}) is multiplied by a factor of three denoting the Ethernet switches in the metro network and in the data center network. Similarly, edge routers are multiplied by a factor of two considering the edge routers in the metro edge and in the data center. Considering the fact that core routers are overprovisioned for future growth of the demand, power consumption of core routers is multiplied by an overprovisioning factor $O_{\text{c}} > 1$, which is taken as 2 [16]. Furthermore, power consumption of the core routers is multiplied by a factor (H_{c}) denoting the

core hops, which also contributes to the power consumption of the WDM equipment. Since the WDM equipment is reused while routing within the core, the number of hops in the WDM network is degraded by a factor, which is 2. The whole summation is multiplied by two overhead factors, which are F_{overhead} denoting the redundancy and cooling overheads and F_{util} denoting the underutilization of the core routers. Baliga et al. have reported that setting F_{overhead} and F_{util} to 3 and 2, respectively, is feasible for energy per bit calculation in the public cloud [16]:

$$E_{\text{private}} = F_{\text{overhead}} \cdot F_{\text{util}} \cdot \left(\frac{P_{\text{ses}}}{C_{\text{ses}}} + \frac{3 \cdot P_{\text{es}}}{C_{\text{es}}} + \frac{P_{\text{g}}}{C_{\text{g}}} \right) \tag{11.4}$$

In Eq. (11.4), energy consumption per bit for a private cloud is shown. In the equation, small Ethernet switches are the ones connecting the users to the corporate network while those connecting the corporate network to the data centers are the regular Ethernet switches. Since the formulation includes the Ethernet switches in both the corporate network and the data center, power consumption of the Ethernet switches is multiplied by 3. Similar to the public cloud energy consumption, the whole summation is multiplied by the two overhead factors. F_{overhead} is proposed to be set to 3. However, as opposed to the public clouds, utilization is expected to be significantly less than that in the public cloud; hence underutilization overhead is set to 3 assuming 33% of network utilization [16].

Indeed, power consumption in watts can be derived as the product of energy per bit and the traffic intensity in terms of bits per second, and as stated in [14], transport energy starts dominating the energy consumption as user demands get heavier. Furthermore, in [28], the emergence of energy savings in transport services is explained by the following analogy: An average passenger car consumes 150 g CO_2/km while an IP router port consuming 1 kWh energy emits 228 g CO_2 if renewable resources are not used leading to 2 tons of CO_2 emission in a year. Hence, saving an energy consumption (or GHG emission) of one IP router port can save 13,000 km of the journey of a passenger vehicle.

11.3 ENERGY-EFFICIENT PROCESSING AND STORAGE IN CLOUD COMPUTING

11.3.1 Energy-Efficient Processing in Data Centers

Energy management in data centers has various aspects as specified in Section 11.2. These aspects are thermal management, energy-efficient server hardware, optimal power distribution among the servers, and deployment of chiller-less cooling techniques in the data centers. As stated in Section 11.2.1, water evaporation is one of the free-cooling techniques. Cooling towers are used in the data centers where hot water from the data center is carried to the top of the cooling tower and while flowing down, some portion of the water evaporates letting the remaining water cool, and the cooled water is supplied back to the data center from the ground. In [29], four main power saving techniques are surveyed: dynamic component deactivation, workload

consolidation, resource throttling, and dynamic voltage and frequency sampling. There are other cooling techniques that eliminate chiller-based cooling, however, this chapter studies the energy-efficient computing perspective of cloud services; hence we will focus on thermal management and optimal power distribution among the servers in the data centers.

11.3.1.1 Thermal-Aware Workload Placement

Optimal power distribution among the servers aims at minimizing heat recirculation in the data center in order to avoid hot spots. Minimizing heat recirculation requires thermal-aware scheduling of the jobs. The total heat that recirculates in the data center (δQ) with n servers is formulated as shown in Eq. (11.5). In the equation, T_i^{in} and T_{sup} denote the inlet temperature of the server-i and the temperature supplied by the CRAC, respectively. Air flow rate at the server-i is denoted by m_i (in kg/s) while C_p is the specific heat of air (in W s/kg K):

$$\delta Q = \sum_{i=1}^{n} C_p \cdot m_i \cdot (T_i^{\text{in}} - T_{\text{sup}}). \tag{11.5}$$

The Minimum Heat Recirculation (MinHR) algorithm has been proposed in [30]. The MinHR algorithm aims at distributing the power (P_i) proportional to the ratio of the heat produced (Q_i) to the heat recirculated (δQ_i) at server i. A reference heat value (Q_{ref}) is set whenever an *event* such as a new CRAC or a new group of servers is installed, and a reference heat recirculation value (δQ_{ref}) is calculated accordingly. In the second step of MinHR, a group of adjacent servers are considered to be a pod; thus, the data center network consists of P pods. The power level of the servers in each pod is determined by P iterations. In each iteration (say iteration j), the power level of the servers in pod-j is set to maximum; then the amount of generated and recirculated heat is calculated denoting the Heat Recirculation Factor for pod-j (HRF_j) as shown in Eq. (11.6). Once all HRF values are obtained, the total power level of each pod is multiplied by the normalized sum of HRFs as seen in Eq. (11.6) thus, power levels ($P_{\text{pod-}j}$) are set as seen in Eq. (11.7):

$$\text{HRF}_j = \frac{Q_j - Q_{\text{ref}}}{\delta Q_j - \delta Q_{\text{ref}}}, \tag{11.6}$$

$$P_{\text{pod-}j} = \frac{\text{HRF}_j}{\sum_{i=1}^{P} \text{HRF}_i} \cdot P_j. \tag{11.7}$$

At the end of each placement, the supply temperature of the CRAC (T_{sup}) is adjusted by T_{adj} by considering the maximum values of the server inlet (T_{in}) and redline temperatures. The redline temperature denotes the safe server inlet temperature. As seen in Eq. (11.8), T_{adj} can have a negative value if the maximum observed inlet temperature exceeds the maximum redline temperature. In such a case, T_{sup} is decreased in order to supply cooler air into the data center:

$$T_{\text{adj}} = T_{\text{redline}}^{\max} - T_{\text{in}}^{\max}. \tag{11.8}$$

MinHR can jointly guarantee minimum heat recirculation and maximum server utilization. However, thermal-aware workload placement lacks cooling awareness since dynamic cooling behaviors of the CRACs are not considered. Thus, minimum heat recirculation cannot always lead to minimum PUE [31].

11.3.1.2 Thermal and Cooling-Aware Workload Placement

Challenges faced by thermal-aware workload placement are being addressed by introducing cooling awareness [31]. Cooling awareness denotes consideration of the cooling behavior of CRAC so that cooling costs are minimized. Figure 11.5 illustrates two scenarios to explain the motivation behind cooling and thermal awareness. Two jobs arrive at the data center where three servers are available. In the first scenario (Figure 11.5a), the jobs are placed in a first-come first-serve (FCFS) fashion, i.e., no cooling nor thermal awareness. Thus, job 1 is scheduled on server 1, and job 2 is scheduled on server 2. Server 3 has the lightest load; hence it requires the thermostat to be 22°C. Scheduling job 2 on server 2 leads to a thermostat setting of 20°C in order to let server 2 work properly. On the other hand, scheduling job 1 on server 1 generates a hot spot in the data center; hence server 1 requires the thermostat setting to be 18°C. In the second scenario, job placement is done by considering thermal activity along with the cooling behavior in the data center. Scheduling job 2 on server 3 does not lead to a thermostat setting lower than 22°C while scheduling job 1 on server 2 requires the thermostat setting to be 20°C for server 2 to work properly. While adjusting the thermostat supply temperature, the lowest temperature requirement has to be considered to ensure that all servers are working properly. Hence in the second scenario, the thermostat supply temperature has to be set to 20°C where in the first scenario, the thermostat setting should be adjusted to 18°C leading to higher cooling costs.

In order to manage this scheduling process, Banerjee et al. have proposed a spatial scheduling algorithm, namely the *Highest Thermostat Setting (HTS)*. The proposed method differs from conventional solutions due to considering multimode operation of the CRAC unit rather than supplying cool air at a constant temperature. Indeed, each operation mode of the CRAC extracts a certain level of heat which

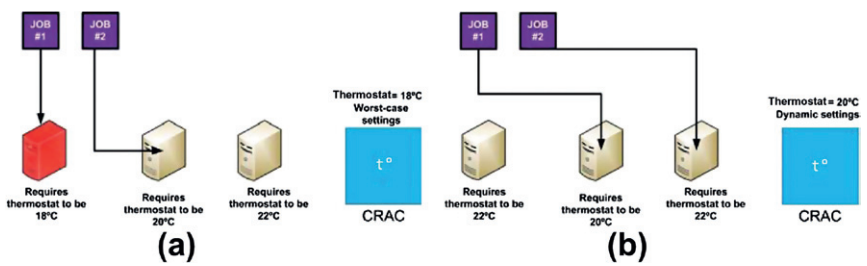

Figure 11.5 Workload placement: (a) first-come first-serve; (b) thermal and cooling-aware [31].

is specific to the corresponding mode. The HTS algorithm initially schedules the arriving jobs in the time domain using either FCFS or earliest deadline first (EDF). Cooling-thermal-aware workload placement is based on coordination between the three steps in (i)–(iii) as shown in Figure 11.6. These three steps can be explained as follows:

(i) *Static ranking of servers:* This step assigns ranks to servers based on their CRAC thermostat setting requirements to meet their redline inlet temperatures for 100% utilization. The server with the highest thermostat temperature requirement is assigned the lowest rank so that the incoming jobs are more likely to be scheduled on the servers with the highest thermostat setting requirement leading to reduced cooling costs. In [31], CRAC high threshold requirements for a server ($T^{\text{th}}_{\text{high}_i}$) are formulated as shown in Eq. (11.9):

$$T^{\text{th}}_{\text{high}_i} = \frac{T^{\text{redline}} - \sum_j d_{ij} \cdot P^{\text{full}}_j}{\sum_j f_{ij}} + \frac{p^{\text{low}}_{\text{ex}}}{r_{\text{ac}}} - \frac{(P^{\text{comp}}_h)^{\text{full}} - p^{\text{low}}_{\text{ex}}}{r_{\text{room}}} \cdot t_{\text{sw}} \quad (11.9)$$

According to the static server ranking function, high threshold requirements of a server are a function of the maximum allowed inlet temperature for the

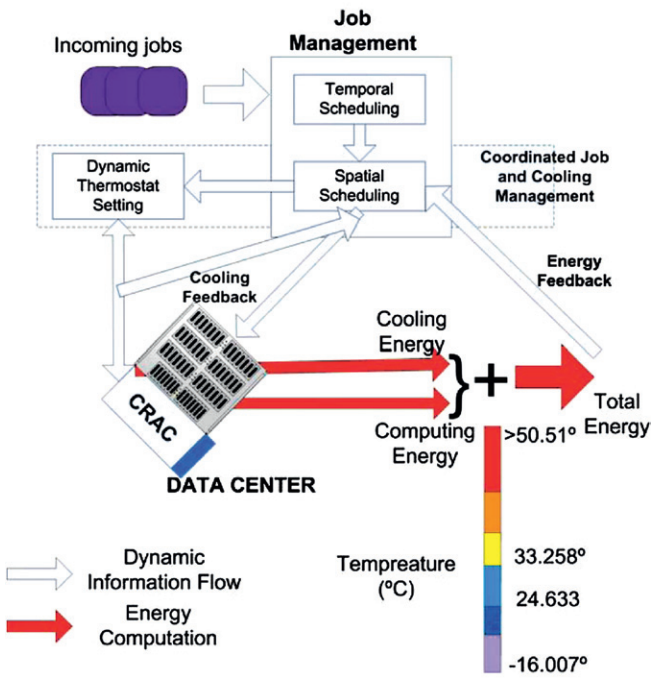

Figure 11.6 Coordinated workload placement proposed in [31].

corresponding server (i.e., the first summation term), temperature change due to CRAC power extraction (i.e., the second summation term), and the increase in temperature due to switching time of CRAC (i.e., the third summation term). In the first term denoting the maximum inlet temperature for server i, T^{redline} denotes the maximum inlet temperature that is specified by the manufacturer, d_{ij} is the heat recirculation coefficient between server i and server j, P_j^{full} is the power consumption of the server when fully utilized, and f_{ij} denotes the fraction of cold air supply flowing from CRAC to server-j. In the second term, $p_{\text{ex}}^{\text{low}}$ denotes the power extracted by CRAC in low-power mode while r_{ac} is the thermal capacity of air supplied by CRAC per unit time. In the third term, where the increase in temperature due to switching time of the CRAC unit is formulated, $(P_h^{\text{comp}})^{\text{full}}$ denotes the total computation power in the data center for a period of h while r_{room} and t_{sw} are the thermal capacity of the air in the data center and the switching time of CRAC between its operation modes, respectively.

(ii) *Job placement:* Upon scheduling the jobs in the time domain and determining the server ranks, servers are sorted in decreasing order with respect to their $T_{\text{high}_i}^{\text{th}}$ values. Jobs are popped from the virtual queue where they are temporally scheduled and each job is aimed to be assigned to the server with the lowest rank value.

(iii) *CRAC thermostat setting determination:* As the jobs are spatially scheduled, the power distribution vector is obtained denoting the power consumption throughout the data center. As opposed to step (ii) where 100% utilization of the servers is considered, here, actual server utilization levels are used to compute the highest possible thermostat value ($T_{\text{high}}^{\text{th}}$). In order to derive the equation for $T_{\text{th}}^{\text{high}}$, heat input to server i and heat output from server i during a short time of dt are formulated. The following notation is used in the equation sets:

l_i	Fraction of air flowing from CRAC to server i
r_{ac}	Thermal capacity of air flowing out of CRAC
a_{ji}	Fraction of air recirculating from server j to server i
r	Thermal capacity of air flowing out of a server
$T_i^{\text{in}}(t)$	Inlet temperature of server i at time t
$T^{\text{sup}}(t)$	Temperature supply of CRAC at time t
$T_i^{\text{out}}(t)$	Outlet temperature of server i at time t
$P_i(t)$	Computation power generated by server i at time t

Equation (11.10) formulates the input heat to server server i in a time interval of dt as the sum of the heat supplied by CRAC to server server i and the heat recirculates from other servers to server i in the time interval, dt.

$$\left(l_i \cdot r_{\text{ac}} + \sum_j a_{ji} \cdot r\right) \cdot T_i^{\text{in}}(t) \cdot \mathrm{d}t = l_i \cdot r_{\text{ac}} \cdot T^{\text{sup}}(t) \cdot \mathrm{d}t + \sum_j a_{ji} \cdot r \cdot T_j^{\text{out}}(t) \cdot \mathrm{d}t. \tag{11.10}$$

Equation (11.11) formulates the outlet heat of server i in a time interval of dt as the sum of the inlet heat and the heat generated by server i during the time dt:

$$\left(l_i \cdot r_{\text{ac}} + \sum_j a_{ji} \cdot r\right) \cdot T_i^{\text{in}}(t) \cdot \text{d}t + P_i(t) \cdot \text{d}t = r \cdot T_j^{\text{out}}(t) \cdot \text{d}t. \quad (11.11)$$

Since Eqs. (11.10) and (11.11) are derived for each server in the data center, the equation set formed by applying these two equations to each server in the data center defines a vectored operation, which is formulated in Eq. (11.12). Since the maximum inlet temperature for a server should not exceed the redline temperature, Eq. (11.12) can be reformulated as in Eq. (11.13):

$$[T_{\text{in}}(t)] = [F] \cdot [T^{\text{sup}}(t)] + [D] \cdot [P(t)], \quad (11.12)$$

$$[F^{-1}] \cdot [T_{\text{redline}}(t)] - [F^{-1}] \cdot [D] \cdot [P(t)] = [T^{\text{sup}}(t)]. \quad (11.13)$$

The maximum value of the supply temperature is constrained to the high thermostat setting ($T_{\text{high}}^{\text{th}}$), total computing power (P_h^{comp}), power extraction of CRAC in the low mode, the thermal capacity of air supplied by CRAC per unit time (r_{ac}), and the switching time of CRAC from low mode to high mode (t_{sw}). In [31], the maximum value of the supply temperature of CRAC is formulated as shown in Eq. (11.14). Thus, substitution of Eq. (11.14) into Eq. (11.13) yields Eq. (11.15), which computes the highest thermostat setting value for the algorithm:

$$T_{\text{max}}^{\text{sup}} = T_{\text{high}}^{\text{th}} + \frac{P_h^{\text{comp}} - p_{\text{ex}}^{\text{low}}}{r_{\text{room}}} \cdot t_{\text{sw}} - \frac{p_{\text{ex}}^{\text{low}}}{r_{\text{ac}}}, \quad (11.14)$$

$$T_{\text{high}^{\text{max}}}^{\text{th}} = [F^{-1}] \cdot [T_{\text{red}}] - [F^{-1}] \cdot [D] \cdot [P_h] - \frac{P_h^{\text{comp}} - p_{\text{ex}}^{\text{low}}}{r_{\text{room}}} \cdot t_{\text{sw}} + \frac{p_{\text{ex}}^{\text{low}}}{r_{\text{ac}}}. \quad (11.15)$$

In [31], the authors have shown that HST leads to significant energy savings (up to 15%) over the conventional thermal-aware workload placement algorithms. Moreover, by turning off the idle servers, further energy savings (up to 9%) are possible under HST. In cloud computing, requests usually last short; hence rather than setting the thermostat value dynamically, integration of decision making schemes with HST would assure both energy savings and service quality requirements. Furthermore, thermal- and cooling-aware workload placement can further be enhanced to address the trade-off between quality of service and energy savings.

11.3.2 Energy-Efficient Storage in Data Centers

Besides processing and short-term transactions, data centers also serve as long-term storage. Furthermore, data center storage demand is increasing by 50–60%

per year [32]. Hence deployment of low-power storage equipment and energy-efficient storage techniques can save significant energy in the data centers.

11.3.2.1 Solid State Disks (SSDs)

A solid state disk is a storage device consisting of NAND flash memory and a controller. Recently, SSDs have appeared as an alternative to conventional hard disk drives (HDDs) due to being lightweight, having a small form factor, having no moving mechanical parts, and lower power consumption [33]. For the sake of energy-efficiency, deployment of SSDs in data centers is advantageous however there are several issues that have to be addressed prior to integrating SSDs in data centers. In [32], write reliability is pointed to as one of these challenges since a single level SSD cell bit introduces a write penalty after 100,000 writes; hence this introduces a drawback when compared to conventional hard disk drives. Another challenge introduced by SSDs is the cost/GB. As of the first quarter of 2011, an SSD costs around $1.80 per GB while an HDD costs approximately $0.11 per GB. Although it is not likely that the SSD costs per GB can be reduced dramatically in short term, as reported in [32], enhancements in high performance and low power can let SSDs be integrated in the data center storage systems.

11.3.2.2 Massive Arrays of Idle Disks (MAIDs)

MAIDs consist of a large amount of hard disk drives that are used for nearline storage. Thus, a hard disk drive [34] spins up whenever an access request arrives for the data stored in it and the rest of the storage consists of a large number of spun down disks. Although MAID can lead to significant power savings due to nearline storage, it also introduces the trade-off between energy-efficiency and performance since spinning up takes more time than data access does. Hence, vendors such as DataDirect Networks Inc., EMC Corp., Fujitsu, Hitachi Data Systems, NEC Corp., and Nexsan Technologies Inc. have introduced multiple levels of power savings to MAID in order to overcome this trade-off. Furthermore, sleep mode support has also been introduced to the MAID system in order to avoid spinning down during peak load hours [35]. Thus, storage systems supporting spin-down and nearline storage seem promising to ensure energy-efficient storage in the data centers [36].

11.3.2.3 Storage Virtualization

Virtualized storage denotes a logical storage pool that is independent of the physical location of the disks [37]. In virtualized storage, unused storage segments can be consolidated in logical storage units increasing the storage efficiency. Enhanced management of the storage pool leads to reduced storage energy since idle physical resources can be spun down and/or put in the standby mode.

Figure 11.7 illustrates a storage area network (SAN), which enables virtualization of storage units. Servers are connected to the physical resources through SAN switches; hence a global storage pool is available to each server. Whenever a storage block requires allocation, a logical unit number is assigned to the allocated virtual space, which acts as a pointer between the physical storage resource and the logical

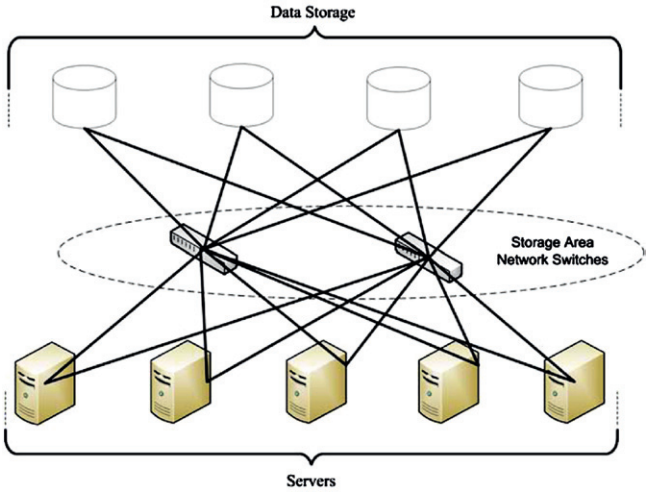

Figure 11.7 Storage area network [37].

storage block. Microsoft's Cluster Server, VMWare's File Server, and IBM's General File System are well-known examples of virtualized storage systems. Besides, IBM System Storage SAN Volume Controller (SVC) is another storage virtualization solution where more active data are moved to SSDs to ensure further energy-efficiency [38].

11.3.3 Monitoring Thermal Activity in Data Centers

As we have seen in the previous subsection, thermal- and cooling-aware workload placement is crucial to meet the desired PUE (and DCE as well) levels. In order to make decisions to increase/decrease the heat supply of the CRAC and place arriving jobs to the servers by avoiding hot spots in the data center, data center operators require online thermal monitoring of the data center. Knowingly, heat distribution in the data center is a function of many parameters including the CRAC temperature setting, workload placement, rack layout design, server types, and so on; hence such a monitoring system would further eliminate the need for on-demand air flow and thermodynamics analysis for dynamic energy management in a data center. Moreover, having the opportunity of thermal monitoring of data centers can also enable the operators to maintain the rack layout of the data centers more efficiently in long term.

The Data Center Genome Project of Microsoft Research and MS Global Foundation Services has proposed using wireless sensor networks (WSNs) for thermal monitoring of the data center [39]. Figure 11.8 illustrates the architecture of a data center monitoring system that mainly consists of three main blocks: data center, data collection/processing/analysis, and the decision-making blocks. Rack layout, cooling and power systems, heat distribution, server performance, and load variations are

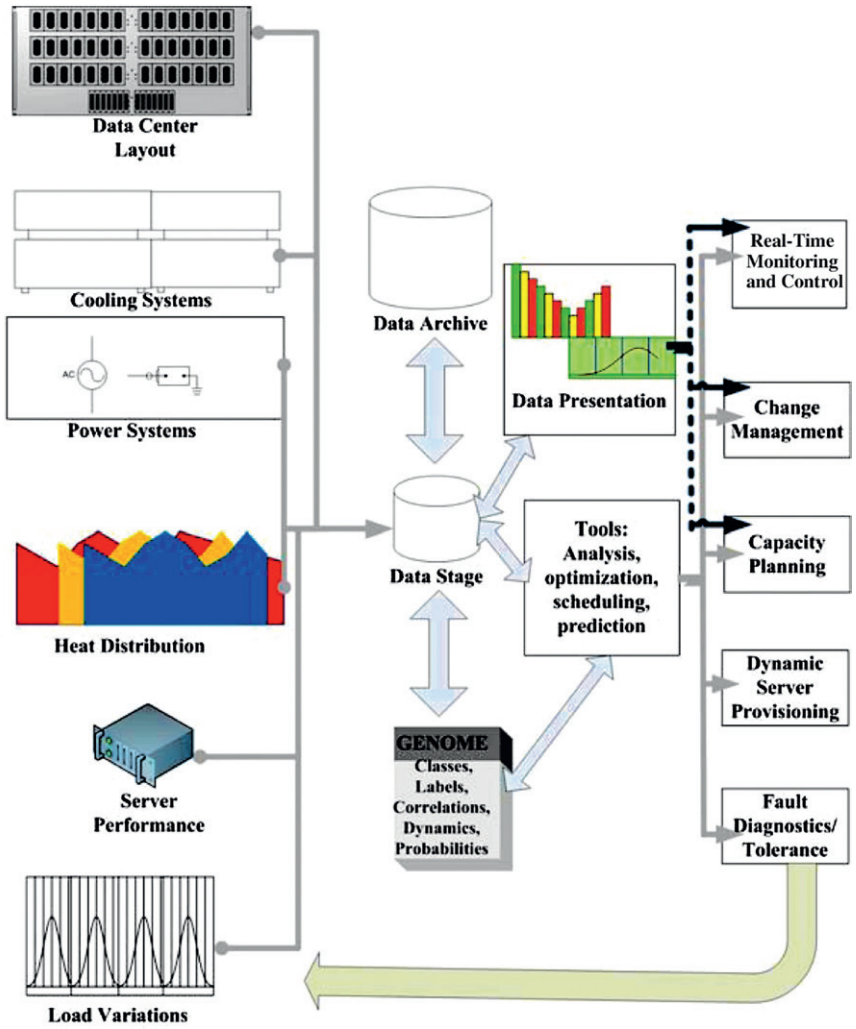

Figure 11.8 Data center monitoring architecture of Genome [39].

the phenomena that are monitored in the data center. The collected data are stored, archived, and used for analysis, optimization, and scheduling purposes. The outputs of the storage/processing block are used for real-time monitoring and control, change management in the data center, capacity planning, dynamic server provisioning, and fault diagnostics or tolerance.

The motivation behind deploying WSNs is explained as WSNs offering a low-cost and nonintrusive technology that has wide coverage and can be repurposed. The Genome Project deploys a sensor network called RACNet in a multimegawatt

(MMW) data center. RACNet consists of approximately 700 sensors. The sensor nodes form a hierarchical topology where master and slave sensor nodes coexist and cooperate. Slave sensors are mounted on the server racks, and they do not have a radio interface. The main function of the slave sensors is collection of the temperature and humidity data and reporting the collected data to the master sensors via serial wired interfaces forming a daisy chain circuit. Each master sensor node has a serial data communication interface and an IEEE 802.15.4 radio interface, which provides the WSN connectivity between the master nodes and the base station.

On the other hand, WSNs face several challenges when deployed in data centers. First of all, the data center introduces a tough radio frequency (RF) environment due to high metal contents of racks and servers, cables, railings, and so on. Besides, since the wireless sensor nodes need to be densely deployed for fast and reliable data collection, packet collision probability is expected to be high. Furthermore, IEEE 802.15.4 radio in RACNet introduces high bit error rate (BER).

In order to overcome the challenges, the Genome project has proposed a reliable Data Collection Protocol (rDCP). The following three key solutions are used by rDCP:

(i) *Channel diversity:* rDCP coordinates multiple base stations among 16 concurrent channels in the 2.4 GHz ISM band without collision. The number of master nodes on a channel is dynamically determined based on link quality measurements.

(ii) *Adaptive bidirectional collection tree:* On each wireless channel, rDCP adaptively builds a bidirectional collection tree. Hence, hop-by-hop connectivity can be met by selecting the high quality links when communicating with the master nodes.

(iii) *Coordinated data retrieval:* Each wireless sensor node (master node) has flash memory on board. Once the master node receives data through the daisy chain of slave nodes, it does not immediately transmit to the base station; instead it stores in the flash memory. Once the base station polls the master node by sending a retrieval request message, the data collected are retrieved over the collection tree in the opposite direction. It is worth noting that rDCP ensures that there is at most one retrieval data stream at a time on a radio channel.

Scalability of the data center monitoring system is a big challenge. For instance, consider that a data center consisting of 1000 racks with 50 servers in each rack required at least 50,000 sensor nodes in order to collect temperature and humidity data. Furthermore, if the humidity and thermal properties collected in different parts of an aisle are highly correlated, some of the sensor nodes will provide redundant data although they can be temporarily put into sleep mode to prolong their battery life. Furthermore, at the sink, processing highly correlated data will consume more processing power than is likely desired. In [40], a self-organizing monitoring system has been proposed for data centers, which consists of scalar sensors mounted on the server racks and thermal cameras mounted on the walls. In the self-organizing data center monitoring system, sensors are grouped as representative (REP) and associate (ASSOC) nodes in order to

eliminate the data redundancy. A sensor node i is said to be a potential ASSOC of a sensor node j if their sensed data is similar and correlated. Sensed data of two sensors are said to be similar if the difference between their mean values is below a certain threshold. The sensed data are said to be correlated if their correlation coefficient is greater than a prespecified threshold value. The correlation coefficient is calculated by using a small finite number of samples taken from the data sensed by corresponding sensor nodes. Similarity and correlation thresholds are set by the data center operator in advance, and the values of these parameters may vary between different parts of the data center. A REP node is selected out of a set of ASSOC nodes.

In order to reduce costs, in [40], thermal cameras are also used with the scalar sensor nodes since thermal cameras have larger fields of view. A thermal camera uses infrared radiation from the objects and generates a 2D thermal image of the data center. Localization of hot spots is possible by exchanging the images between the thermal cameras. However, transmission of raw images between thermal cameras requires high bandwidth; hence compression at the sender and decompression at the receiver enhances bandwidth efficiency. The thermal signature of the data center (i.e., expected thermal map) is computed in advance by considering temperature, air flow rate, and workload placement factors. Thermal cameras and scalar sensors collaborate to obtain the actual thermal map. If the observed thermal map is different than the expected thermal map, the observed data are said to be an *anomaly*. A thermal map that is detected as an anomaly is classified by a supervised neural network since an anomaly can be due to the several reasons such as problems due to the cooling system, increase in cold water temperature, misconfiguration of servers, CPU fan failures, security attacks that overload the servers with illegitimate jobs, and so on. In the case of an anomaly detection, the base station can request the sensed data at the anomaly location in finer granularity, i.e., more sensed data are needed from the anomaly region.

11.4 OPTIMAL DATA CENTER PLACEMENT

The location of data centers has a significant impact on the transport energy of cloud services in the Internet backbone. Furthermore, varying load profile on the backbone nodes due to different time zones (such as the nationwide backbone network in the United States, known as NSFNET) affect the energy consumption throughout the day. Besides, while aiming at minimum power consumption in data center placement, taking advantage of availability of renewable resources such as solar panels and wind farms can lead to further reduction in CO_2 emissions. In [28], a mathematical model has been introduced with the objective of minimizing nonrenewable energy consumption at data centers. Wind farms are assumed to be used to power the data centers while solar panels are employed to power the nodes of the transport network. In this section, we briefly summarize the corresponding optimization model for data center placement.

It is assumed that the backbone employs an IP over WDM network with the optical bypass technology to transport the data center and regular Internet traffic as shown in Figure 11.2a for a public cloud. Figure 11.9 focuses on a backbone link in

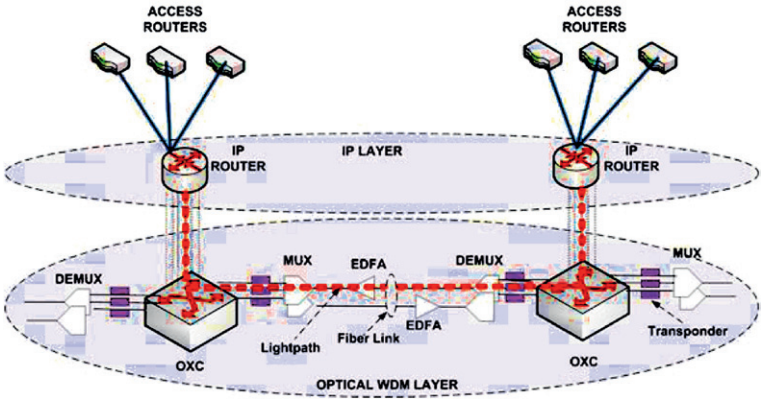

Figure 11.9 Components of an IP over WDM network.

Figure 11.2a. As seen in the figure, the traffic arriving from the access routers are aggregated at the IP routers. The IP layer is bypassed, and the traffic is routed at the optical WDM layer as much as possible. A virtual topology is formed from the physical topology to provision the demands, where a virtual link between two nodes denotes a lightpath between the corresponding nodes in the physical topology. It is not always possible to arrive at the destination node with one virtual hop; hence the traffic is terminated at the optical layer and transmitted to the IP layer where it is routed to the destination over the next virtual hop. This principle is known as *multihop bypass* technology [41]. The notation used in the linear programming (LP) formulation in [28] is shown below:

P_{r}	Power consumption of an IP router port
P_{t}	Power consumption of a transponder
P_{MD}	Power consumption of a (de) multiplexer
DM_i	Number of (de) multiplexers at node i
N_i	Set of neighboring nodes of node i
P_{e}	Power consumption of an erbium-doped fiber amplifier (EDFA)
NEDFA_{mn}	Number of EDFAs deployed in the physical link mn
f_{mn}	Number of fibers deployed in the physical link mn
P_i^{OXC}	Power consumption of the optical switch at node i
$(C_{ij}^t)^{\mathrm{DC}}$	Number of lightpaths that carry data center traffic over the virtual link ij at time t. The corresponding traffic either starts or ends at a data center and it is powered by nonrenewable resources.
$(C_{ij}^t)^{\mathrm{DCW}}$	Number of lightpaths that carry data center traffic over the virtual link ij at time t. The corresponding traffic either starts or ends at a data center and it is powered by wind energy.
$(C_{ij}^t)^{\mathrm{NDC}}$	Number of lightpaths that carry data center traffic over the virtual link ij at time t. The corresponding traffic either starts or ends at a regular node and it is powered by nonrenewable resources.

$(C_{ij}^t)^{\text{NDCW}}$ Number of lightpaths that carry data center traffic over the virtual link ij at time t. The corresponding traffic either starts or ends at a regular node and it is powered by wind energy.

$(C_{ij}^t)^{\text{NDCS}}$ Number of lightpaths that carry data center traffic over the virtual link ij at time t. The corresponding traffic either starts or ends at a regular node and it is powered by solar energy.

$(C_{ij}^t)^{\text{R}}$ Number of lightpaths that regular traffic over the virtual link ij at time t. The corresponding traffic is powered by nonrenewable resources.

$(C_{ij}^t)^{\text{RW}}$ Number of lightpaths that regular traffic over the virtual link ij at time t. The corresponding traffic is powered by wind energy.

$(C_{ij}^t)^{\text{RS}}$ Number of lightpaths that regular traffic over the virtual link ij at time t. The corresponding traffic is powered by solar energy.

Q_i^t Number of aggregation ports at node i at time t that are powered by nonrenewable resources.

$Q_i^{t\,\text{w}}$ Number of aggregation ports at node i at time t that are powered by wind energy.

$Q_i^{t\,\text{s}}$ Number of aggregation ports at node i at time t that are powered by solar energy.

$W_{mn}^{\text{DC}\,t}$ Number of wavelength channels on physical link mn, carrying the data center traffic. The corresponding traffic either starts or ends at a data center, and it is powered by the nonrenewable resources.

$W_{mn}^{\text{DCW}\,t}$ Number of wavelength channels on physical link mn, carrying the data center traffic. The corresponding traffic either starts or ends at a data center, and it is powered by wind energy.

$W_{mn}^{\text{NDC}\,t}$ Number of wavelength channels on physical link mn, carrying the data center traffic. The corresponding traffic either starts or ends at a regular node, and it is powered by the nonrenewable resources.

$W_{mn}^{\text{NDCW}\,t}$ Number of wavelength channels on physical link mn, carrying the data center traffic. The corresponding traffic either starts or ends at a regular node, and it is powered by wind energy.

$W_{mn}^{\text{NDCS}\,t}$ Number of wavelength channels on physical link mn, carrying the data center traffic. The corresponding traffic either starts or ends at a regular node, and it is powered by solar energy.

$W_{mn}^{\text{R}\;\;t}$ Number of wavelength channels on physical link mn, carrying the regular traffic. The corresponding traffic is powered by the nonrenewable resources.

$W_{mn}^{\text{RW}\,t}$ Number of wavelength channels on physical link mn, carrying the regular traffic. The corresponding traffic is powered by wind energy.

$W_{mn}^{\text{RS}\,t}$ Number of wavelength channels on physical link mn, carrying the regular traffic. The corresponding traffic is powered by solar energy.

PWF_k^t Output power of wind farm k at time

S_i^t Output power of solar cells of node i at time t

U_k Available portion of the wind farm k output power to supply data centers

ℓ_k^i Loss of output power of wind farm k when transporting to data center i

δ_k^i Binary variable is 1 if data center i can be powered by wind farm k

The objective of the data center placement problem is to minimize the utilization of nonrenewable resources to power the transport medium. Here, this function is presented by the sum of five components: power consumption at the IP routers (P_{IP}), power consumption by the transponders at the optical layer (P_{optical}), power consumption due to amplification through EDFAs (P_{amp}), and power consumption due to optical switching (P_{sw}) as shown in Eq. (11.16). In the equation, T is the set of time points, and multiplying the power consumptions by time gives the energy consumption of the nonrenewable resources. Each component of the summation in the objective function is explained below:

$$\text{minimize} \quad \sum_{t \in T} P_{\text{IP}} + P_{\text{optical}} + P_{\text{amp}} + P_{\text{sw}}. \tag{11.16}$$

Equation (11.17) formulates the power consumption at the IP router ports. The corresponding IP routers are powered by nonrenewable resources and they carry three types of demands as follows: (i) data center traffic that is either originating or arriving at a data center, (ii) data center traffic that either originates or arrives at a regular node, and (iii) regular Internet traffic:

$$P_{\text{IP}} = \sum_{i \in N} \sum_{j \in N, j \neq i} P_{\text{r}} \cdot \left((C_{ij}^t)^{\text{DC}} + (C_{ij}^t)^{\text{NDC}} + (C_{ij}^t)^{\text{R}} \right) + \sum_{i \in N} P_{\text{r}} \cdot Q_i^t. \tag{11.17}$$

Equation (11.18) formulates the power consumption at the transponders. The corresponding optical nodes are powered by nonrenewable resources and they carry the data center traffic that is either originating or arriving at a data center, data center traffic that either originates or arrives at a regular node, and the regular Internet traffic:

$$P_{\text{optical}} = \sum_{m \in N} \sum_{n \in N_m} P_{\text{t}} \cdot \left(W_{mn}^{\text{DC}^t} + W_{mn}^{\text{NDC}^t} + W_{mn}^{\text{R}\ t} \right). \tag{11.18}$$

Equation (11.19) formulates the power consumption of EDFAs that are powered by nonrenewable resources. Here, it is worth noting that the number of EDFAs deployed in a fiber (NEDFA_{mn}) is calculated as $\lfloor L_{mn}/L_{\text{span}} \rfloor + 1$ where L_{mn} is the length of the fiber between node m and node n and L_{span} is the minimum distance between two EDFAs, which is often taken as 80 km:

$$P_{\text{amp}} = \sum_{m \in N} \sum_{n \in N_m} P_{\text{e}} \cdot \text{NEDFA}_{mn} \cdot f_{mn}. \tag{11.19}$$

The last two components of total transport power consumption are the power consumed by optical switches and the power consumption of (de) multiplexers that are powered by nonrenewable resources and formulated by Eqs. (11.20) and (11.21), respectively:

$$P\text{sw} = \sum_{i \in N} P_i^{\text{OXC}}, \tag{11.20}$$

$$\sum\nolimits_{i \in N} P_{\text{MD}} \cdot \text{DM}_i. \tag{11.21}$$

The constraints mainly denote the link and power capacity, flow conservation, and wavelength continuity constraints as the rest of the optimization model inherits the energy-minimized design of IP over WDM networks [41], and its details can be found in [28]. Here, we focus on the constraints related to renewable energy consumption as seen in Eqs. (11.22)–(11.24). The constraint in Eq. (11.22) ensures that renewable energy consumption at the router ports and transponders as well as the cooling and computing equipments in a data center cannot exceed the power supplied by a single wind farm. In the equation, power losses due to transportation of the wind energy to the data center are also taken into consideration:

$$\begin{aligned} &\sum\nolimits_{j\in N, i\neq j} P_{\mathrm{r}}^{\mathrm{w}} \cdot \left((C_{ij}^{t})^{\mathrm{DCW}} + (C_{ij}^{t})^{\mathrm{NDCW}} + (C_{ij}^{t})^{\mathrm{RW}}\right) \\ &+P_{\mathrm{r}}^{\mathrm{w}} \cdot Q_i^{t\,w} + \sum_{m\in N_i} P_t^{\mathrm{w}} \cdot \left(W_{im}^{\mathrm{DCW}\,t} + W_{im}^{\mathrm{RW}\,t} + W_{im}^{\mathrm{NDCW}\,t}\right) + P_i^{\mathrm{cool}} + P_i^{\mathrm{compute}} \\ &\leqslant \mathrm{sum}_{k\in K}\,\delta_k^i \cdot \mathrm{PWF}_k^t \cdot (1-\ell_i)\cdot U_k, \quad \forall t \in T,\ i \in N. \end{aligned} \tag{11.22}$$

By Eq. (11.23), the solar power that is available to a backbone (regular) node sets an upper bound for the renewable energy consumed by the router ports and the transponders of the corresponding node:

$$\begin{aligned} &\sum_{j\in N, i\neq j} P_{\mathrm{r}}^{\mathrm{s}} \cdot \left((C_{ij}^{t})^{\mathrm{NDCS}} + (C_{ij}^{t})^{\mathrm{RS}}\right) \\ &+P_{\mathrm{r}}^{\mathrm{s}} \cdot Q_i^{t\,s} + \sum_{m\in N_i} P_t^{\mathrm{s}} \cdot \left(W_{im}^{\mathrm{RS}\,t} + W_{im}^{\mathrm{NDCS}\,t}\right) \\ &\leqslant (1 - \mathrm{sum}_{k\in K}\,\delta_k^i)\cdot S_i^t, \quad \forall t \in T,\ i \in N. \end{aligned} \tag{11.23}$$

The left-hand side of Eq. (11.24) formulates the total renewable power consumption by all data centers. Thus, total renewable power consumption of all data centers is constrained to the total power supply of the wind farms:

$$\begin{aligned} \sum_{i\in N} \Bigg\{ (1+\ell_i^k)\cdot \Bigg[&\sum_{j\in N, j\neq i} P_{\mathrm{r}}^{\mathrm{w}} \cdot \left((C_{ij}^{t})^{\mathrm{DCW}} + (C_{ij}^{t})^{\mathrm{NDCW}} + (C_{ij}^{t})^{\mathrm{RW}}\right) + P_{\mathrm{r}}^{\mathrm{w}} \cdot Q_i^{t\,\mathrm{w}} \\ &+ \sum_{m\in N_i} P_t^{\mathrm{w}} \cdot \left(W_{im}^{\mathrm{DCW}\,t} + W_{im}^{\mathrm{RW}\,t} + W_{im}^{\mathrm{NDCW}\,t}\right) \Bigg] + P_i^{\mathrm{cool}} + P_i^{\mathrm{compute}} \Bigg\} \\ &\leq \sum_{k\in K} PWF_k^t \cdot U_k, \quad \forall t \in T,\ i \in N. \end{aligned} \tag{11.24}$$

In [28], optimal data center placement over the NSFNET topology has been tested by using the above LP model. The following three wind farms have been considered: Cedar Creek Wind Farm, Capricorn Ridge Wind Farm, and Twin Groves Wind Farm.

Solar power available to the backbone nodes is affected by the time of the day; e.g., at 23:00 (EST), no solar power is available for the nodes in the Eastern Time Zone while the nodes in the Pacific Time Zone can still have solar power. Without loss of generality, the data centers are assumed to operate with a PUE of 2. According to the LP formulation results, locating the data centers at the center of the NSFNET topology can lead to minimization of nonrenewable power utilization, i.e., minimum CO_2 emissions, as more wind power is available if the data centers are built at the center of the network. Furthermore, the authors have shown that nonrenewable power consumption of the network can be reduced by 20% when compared to a scenario where data centers are randomly located over the NSFNET backbone. Another important result that has been reported is that the optimal placement of data centers by considering renewable resources, and incorporation of multihop bypass technology when routing the demands can lead to reduction in nonrenewable power consumption of up to 77%. Besides, when dynamically provisioning data center requests, in order to ensure fewer hops, replication of data among the data centers based on popularity can also lead to further savings in nonrenewable power consumption as stated by [28].

11.5 ENERGY-EFFICIENT TRANSPORT OF CLOUD SERVICES

11.5.1 From Unicast/Multicast to Anycast/Manycast

Traditionally, connection demands are provisioned with respect to unicast or multicast. Unicast provisioning is denoted by (s, d) where s and d denote the source and destination nodes of the incoming demand, respectively. Multicast provisioning is denoted by (s, D) where s is the source node and D is the set of destination nodes that the messages are required to be delivered to. In distributed computing applications such as cluster computing, grid computing, or cloud computing, one or more destinations are selected out of a group of candidate destinations. Selecting a destination out of a candidate destination set is referred to as *anycast*, and it is denoted as $(s, d_s \in D)$ where d_s is the selected destination out of the set of candidate destinations, D. Selection of a group of destinations out of a candidate destination set is referred as *manycast* provisioning, and it is denoted by $(s, D_s \subseteq D)$ where D_s is a subset of the candidate destinations set, D. Obviously, if $D_s = D$, then, manycast is equivalent to multicast, and if $|D_s| = 1$, then, manycast and anycast are identical.

In this section, we study energy-efficient anycast and manycast provisioning of user demands over the Internet since cloud computing services are provisioned with respect to either an anycast or manycast paradigm.

11.5.2 Energy-Efficient Anycast

11.5.2.1 Anycast with Dynamic Sleep Cycles

In [42], Bathula and Elmirghani have proposed an energy-efficient anycast algorithm for the provisioning of jobs submitted to a computational grid. Despite the differences

in programming, business, computation, and data models, cloud and grid computing have common features in architecture, vision, and technology [43]. Hence, anycasting can also be considered for demand provisioning for cloud computing.

In the referred study, the network is represented by a clustered architecture where these computing clusters are interconnected through their boundary nodes. Dynamic sleep cycles are proposed in the clusters in order to save energy. Thus, based on the traffic load profile, the nodes in a cluster are switched to the *off* state, and then, switched back to the *on* state. Routing the jobs submitted to the grid is realized by energy-efficient anycast, i.e., anycast among the *on* clusters without leading to Quality of service (QoS) degradation. For each link j, a network element vector (NEV_i) is defined as follows:

$$\text{NEV}_i = [\eta_i \quad \tau_i]^T, \tag{11.25}$$

where η_i and τ_i are the noise factor and the propagation delay of link i, respectively. For a set of links denoting a route from node s to node d, end-to-end (E2E) noise factor would be the product of noise factors of the links along the path while the E2E propagation delay is the sum of the propagation delays of all links along the path. Thus, NEV for a route R can be obtained as shown in Eq. (11.26):

$$\text{NEV}_R = [\eta_R \quad \tau_R]^T = \left[\prod_{k=s}^{d} \eta_k \quad \sum_{k=s}^{d} \tau_k\right]^T. \tag{11.26}$$

Since jobs are submitted with their service level agreements (SLAs), the provisioning solution must not violate the SLA. Hence, for each job, θ, a threshold NEV is predetermined as shown in Eq. (11.27), and $\text{NEV}_R \leqslant \text{NEV}_{\text{TH}}^{\theta}$ is the aim:

$$\text{NEV}_{\text{TH}}^{\theta} = [\eta_{\text{th}} \quad \tau_{\text{th}}]^T. \tag{11.27}$$

Anycast provisioning of a submitted job ($\theta = (n, D_n)$) is an iterative procedure where an iteration step is illustrated in the flowchart in Figure 11.10. An iteration step works as follows. The initial value of NEV is set to $[1 \quad 0]^T$. The iterations continue until a destination out of the candidate destinations set, D_n is reached. The destinations are sorted in increasing order with respect to their shortest path distances to the source node, n. The destination with the minimum hop distance is selected as the potential destination node (d'). The next hop node to node d' is looked up in the routing table of node n. n_k is the next hop to the potential destination node d'. If node n_k is the boundary node of a cluster which is in the *off* state, then, the candidate destinations set is updated by removing the nodes that are in the same cluster with n_k. Otherwise, NEV is updated as follows:

$$\text{NEV}\,[n-1 \quad n_k] \leftarrow \text{NEV}\,[n-1 \quad n] \circ \text{NEV}\,[n \quad n_k], \tag{11.28}$$

where the Boolean operator $\circ$ denotes multiplication on the noise factor and summation on the propagation delay. If the SLA requirements are met, i.e.,

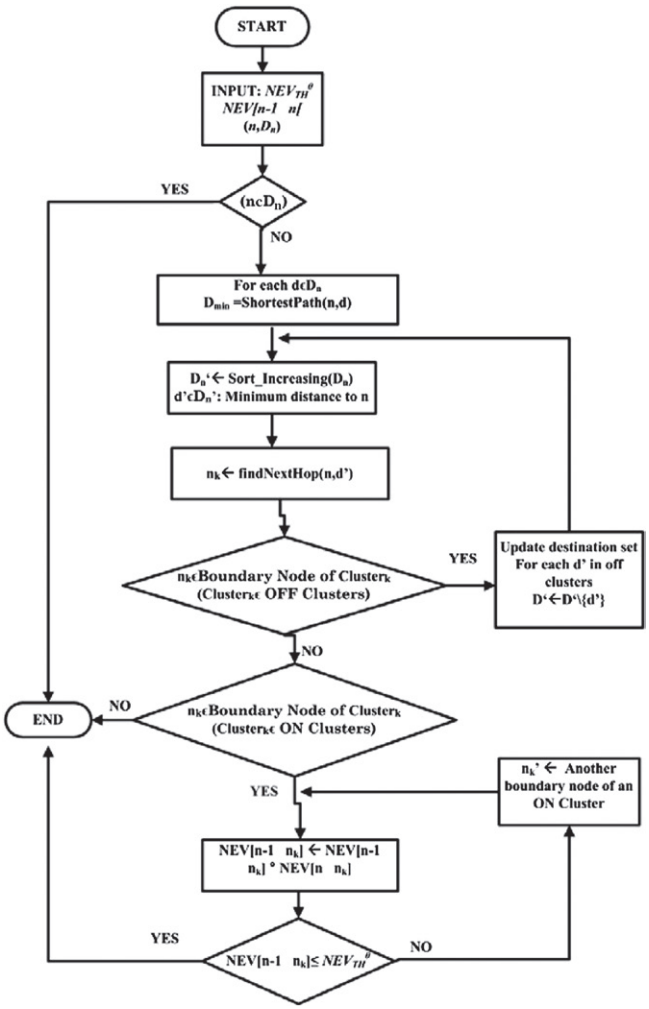

Figure 11.10 An iteration step of an energy-efficient anycast algorithm [42].

$\text{NEV}\,[n-1 \quad n_k] \leqslant \text{NEV}_{\text{TH}}^{\theta}$, the anycast function is called for n_k and the candidate destination set. Otherwise, another candidate destination is selected from another cluster and the above steps are run for the next hop node to the selected cluster.

The proposed anycast scheme is promising in terms of energy-efficiency and SLA guarantee however, dynamic sleep cycle management is crucial to gain maximum benefit in terms of energy-efficiency without SLA violation. Furthermore, the proposed algorithm is based on the assumption that there are at least three clusters in the *on* state in order to provision an incoming request which introduces further requirements in dynamic management of sleep cycles of the clusters.

11.5.2.2 Anycast among Data Center Replicates

In [28], Dong et al. propose the Energy-Delay Optimal Routing (EDOR) algorithm in a network where a few data centers are placed at the backbone nodes and data are replicated in those data centers based on popularity. The algorithm runs on an IP over WDM network, and aims at mapping the physical topology onto a virtual topology by the employment of optical bypass technology as discussed in Section 11.4. Static upstream data center demands are provisioned as follows: The demands are sorted in decreasing order. The first demand in the sorted list is retrieved along with the list of data centers where the desired data are replicated. The algorithm computes all available paths to the retrieved data centers on the virtual topology. If there is sufficient capacity on the virtual paths, the demand is routed based on the shortest path paradigm, and the virtual link capacities are updated to continue with the next demand in the list. Otherwise, a new virtual link is built between the source node and one of the data centers leading to the minimum number of physical hops. Upon updating the virtual topology, the algorithm continues with the next demand in the sorted list.

In [28], the authors show that under the optimal placement of data centers in the NSFNET topology, the EDOR algorithm with multihop bypass technology can ensure an average of 4.5% power savings in the IP over WDM network over non-bypass provisioning. Furthermore, it has also been shown that if the demands are provisioned with respect to the unicast paradigm and with the objective of minimized energy, power savings of the IP over WDM network are limited to 3.7% on average. Thus, anycasting among the replicated data centers lets more demands share bandwidth on the virtual links leading to more power savings. The EDOR algorithm leads to an increase in the propagation delay up to 8% when compared to the shortest distance routing. The EDOR algorithm can be extended by considering SLA requirements such as noise factors and propagation delay. Furthermore, it can be easily modified to provision the demands with respect to the manycast paradigm as well.

11.5.3 Energy-Efficient Manycast

The manycast problem aims to select to reach at a subset of the candidate destinations set originating at the source node where the demand is requested. Considering the employment of optical WDM networks in the Internet backbone, the manycast problem can be defined as the light-tree selection problem [44]. In [45], Kantarci and Mouftah formulated an energy-efficient light-tree selection problem to provision cloud services over optical transport networks and proposed a heuristic for the solution of the corresponding problem.

In Figure 11.11, a cloud over a wavelength routed network is illustrated over the NSFNET topology. Cloud resources are distributed and interconnected through the wavelength routing (WR) backbone nodes. The architecture of a WR node is illustrated at node 3. The main components in a WR switch are the transmitters and receivers where the add and drop traffic is managed, the wavelength switches where the pass-through traffic is transported, and (de) multiplexers where incoming/outgoing traffic is routed among the wavelength switch inlets/outlets or the transceivers. The partial sleep cycle

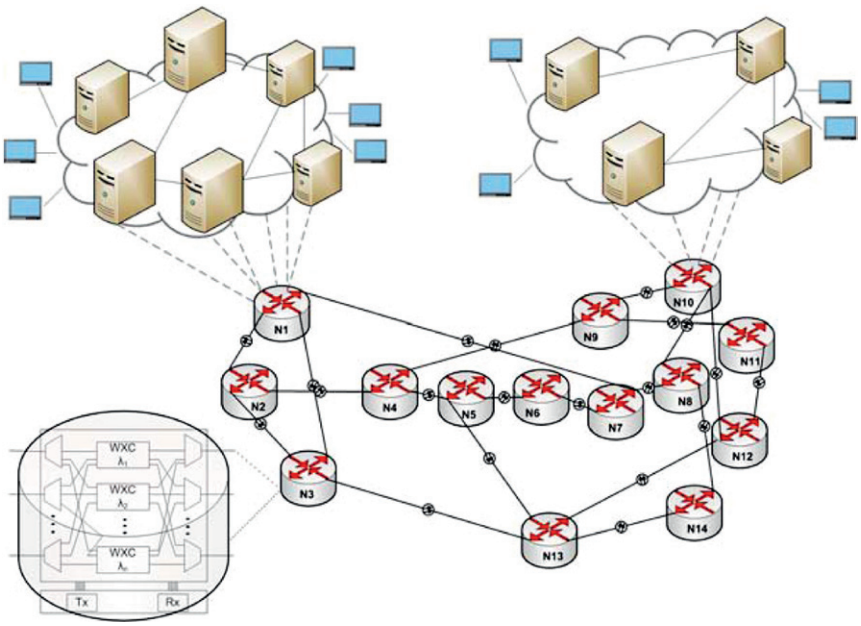

Figure 11.11 Cloud over WR optical network in the NSFNET topology.

idea in [46, 47] is adopted here as follows: A WR node enters the power saving (*off*) mode and the traffic routed via the corresponding node is cut and routed through other nodes that are in the *on* state. To add or drop connections is possible in the *off* mode but pass-through traffic is not routed via a node in the *off* mode. The notation used for optimized provisioning of manycast demands over the wavelength routed network is as follows:

L_{mn} Length of fiber between node m and node n

λ_w^{ij} Number of active wavelength channels on the link ij

χ_i A binary variable, and it is 1 if node is in the power saving mode ($\overline{\chi_i}$ is the one's complement of χ_i)

$\{i, j\}$ Directional link from node i to node j

$u_{i,j}^{x,w}$ A binary variable, and it is 1 if demand x is utilizing the wavelength w on link i, j

$\Phi_{i,j}$ A binary variable, and it is 1 if neither node i nor node j is in the power saving mode

δ_x Lower bound for the number of manycast destinations that have to be reached by demand x

N Number of nodes in the network

O_i^x Addition order of node i to the manycast tree of demand x

Λ_w^x A binary variable, and it is 1 if demand x is utilizing wavelength w on the links of its manycast tree

D_x Set of candidate manycast destinations of demand x

s_x Source node of demand x

Three main components contribute to the power consumption of the optical transport network as follows:

(i) *Power consumption of erbium-doped fiber amplifiers* (E_{EDFA})*:* The number of EDFAs in link ij is a function of the fiber length between node i and node j (dist (i, j)) and the length of a fiber span ($\lfloor L_{mn}/L_{\text{span}} \rfloor + 1$).

(ii) *Power consumption due to switching equipment* (E_{MEMS})*:* MEMS equipment and the wavelength converters in the WR nodes are the major contributors of the switching energy consumption (E_{MEMS}).

(iii) *Idle power consumption*(E_{ON})*:* This denotes the power consumption of a WR node when it is in the *ON* state. Based on the factors mentioned above, total energy consumption in the network can be formulated as shown in Eq. (11.29) where β denotes the average ratio of the time that a WR node spends in the *on* mode. Thus, $(1 - \beta)$ of a WR node's idle power can be saved if the corresponding node spends β of its time in the *on* mode. If the partial sleep cycle idea is adopted, the ratio between the pass-through traffic and the add/drop traffic can be used to determine the value of β. Research by Pramod and Mouftah reports that the add/drop traffic and the pass-through traffic demonstrate a proportion of 3:7 [48]; hence setting β to 0.3 is reasonable as proposed in [45]:

$$\text{Energy} = \sum_i \sum_j (\lfloor L_{mn}/L_{\text{span}} \rfloor + 1) \cdot E_{\text{EDFA}} + \sum_i \sum_j \sum_w \lambda_w^{i,j} \cdot E_{\text{MEMS}} + \sum_i \beta_i \cdot E_{\text{ON}}. \tag{11.29}$$

An optimization model for energy-efficient manycast provisioning is presented below in Eqs. (11.30)–(11.47). The objective function in Eq. (11.30) aims at maximizing the number of nodes that are in the *off* mode. Hereafter, we use the *sleep* and *off* modes interchangeably. The constraint set consists of two parts where the former [Eqs. (11.31)–(11.37)] denotes the energy-efficiency assurance while the latter [Eqs. (11.38)–(11.47)] is the manycast constraint set. Equation (11.31) ensures that a demand x can utilize a channel on link ij if neither node i nor node j is in sleep mode. It is worth noting that it is assumed that the upstream data center traffic traverses at least two hops in the backbone. The next constraint in Eq. (11.32) denotes that the next hop after the source node of a demand has to be in the *on* mode in order to utilize the link between the source node and the corresponding node. In the next constraint [Eq. (11.33)], it is ensured that if node i is one hop before one of the destinations, the light-tree can traverse the corresponding node if and only if it is in the *on* mode. Equation (11.34) ensures that a node can be in either one of the *on* or *off* modes. The next three constraints in Eqs. (11.35)–(11.37) formulates a linear approximation of the product of χ_i and χ_j denoting if two nodes are concurrently in the *off* mode.

$$\text{Maximize} \sum_i \chi_i, \tag{11.30}$$

$$\sum\nolimits_w u_{i,j}^{x,w} \leqslant \Phi_{i,j}, \quad \forall x, i, j \ (i \neq s_x, \{j\} \nsubseteq D_x), \tag{11.31}$$

$$\sum\nolimits_w u_{s_x,j}^{x,w} \leqslant \overline{\chi}_j, \quad \forall x, j\ (\{j\} \nsubseteq D_x), \tag{11.32}$$

$$\sum\nolimits_w u_{i,j}^{x,w} \leqslant \overline{\chi}_i, \quad \forall x, i, j\ (i \neq s_x, \{j\} \subseteq D_x), \tag{11.33}$$

$$\chi_i + \overline{\chi}_i = 1, \quad \forall i, \tag{11.34}$$

$$\Phi_{i,j} - \overline{\chi}_i \leqslant 0, \quad \forall i, j, \tag{11.35}$$

$$\Phi_{i,j} - \overline{\chi}_j \leqslant 0, \quad \forall i, j, \tag{11.36}$$

$$\overline{\chi}_i + \overline{\chi}_j - \Phi_{i,j} \leqslant 1, \quad \forall i, j. \tag{11.37}$$

The remaining of the mathematical model given in Eqs. (11.38)–(11.47) represents the manycast constraints that have initially been presented in [44]. Equation (11.38) denotes that a wavelength on a bidirectional link can be used at most in one direction. Reaching a satisfying number of destinations is guaranteed by Eq. (11.39). The source node must have at least one outgoing wavelength but no incoming wavelengths as formulated by Eqs. (11.40) and (11.41). By Eqs. (11.42)and (11.43), demand x is allowed to utilize at most one wavelength on a bidirectional link. Equations (11.44) and (11.45) stand for the flow conservation constraints while Eq. (11.46) guarantees that loops are avoided. Demand x can utilize one and only one wavelength in its manycast tree as shown in Eq. (11.47).

$$\sum_x (u_{i,j}^{x,w} + u_{j,i}^{x,w}) \leqslant 1, \quad \forall i, j, w, \tag{11.38}$$

$$\sum_i \sum_{j \in D_x} \sum_w u_{i,j}^{x,w} \geqslant \delta_x, \quad \forall x, \tag{11.39}$$

$$\sum_j \sum_w u_{s_x,j}^{x,w} \geqslant 1, \quad \forall x, \tag{11.40}$$

$$\sum_i \sum_w u_{i,s_x}^{x,w} = 0, \quad \forall x, \tag{11.41}$$

$$\sum_i \sum_w u_{i,j}^{x,w} \leqslant 1, \quad \forall x, j \neq s_x, \tag{11.42}$$

$$u_{i,j}^{x,w} + u_{j,i}^{x,w} \leqslant \Lambda^{x,w}, \quad \forall x, w, j, i\ (i < j), \tag{11.43}$$

$$\sum_j (u_{i,j}^{x,w} - N \cdot u_{j,i}^{x,w}) \leqslant 0, \quad \forall x, w, i \neq s_x, \tag{11.44}$$

$$\sum_i (u_{i,j}^{x,w} - u_{j,i}^{x,w}) \leqslant 0, \quad \forall x, w, \{j\} \nsubseteq D_x, \tag{11.45}$$

$$O_i^x - O_j^x + N \cdot u_{i,j}^{x,w} \leqslant N - 1, \quad \forall i, j, x, w, \tag{11.46}$$

$$\sum\nolimits_w \Lambda^{x,w} = 1, \quad \forall x. \tag{11.47}$$

Solution of an optimization model can take a long time, which may not be feasible to make decisions, even if the demand profile can be predicted a few hours

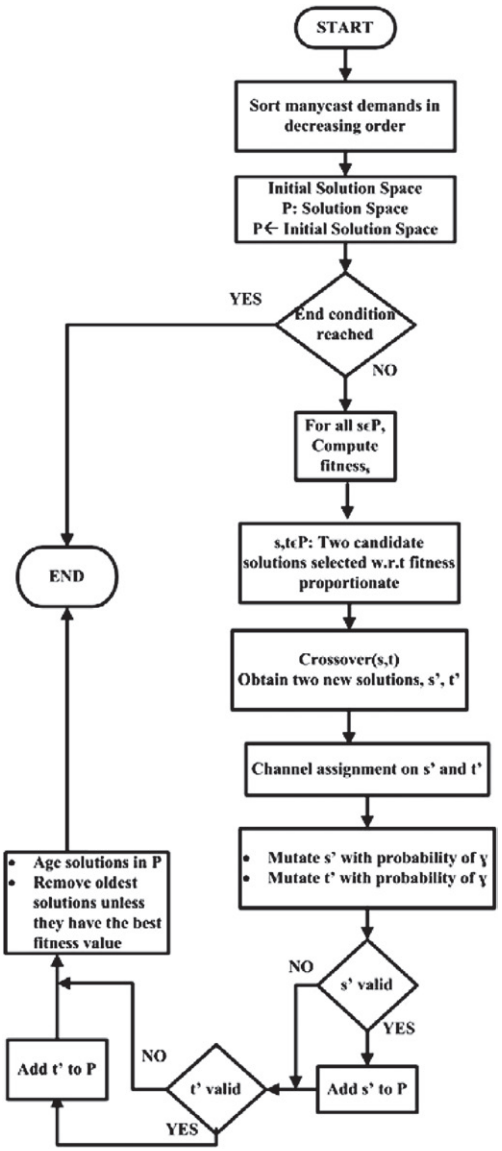

Figure 11.12 Flowchart of the Evolutionary Algorithm for Green Light-tree Establishment (EAGLE) [45].

ahead. Hence, Kantarci, and Mouftah have proposed a heuristic called *Evolutionary Algorithm for Green Light-tree Establishment (EAGLE)*[45]. Figure 11.12 illustrates the flowchart of EAGLE.

As seen in Figure 11.12, EAGLE starts with an initial solution pool which is updated by a number of iterations. The algorithm works as follows: At the beginning

of each iteration, a fitness value is calculated for each individual solution. The fitness function can be defined in various ways. In [45], the authors define three different fitness functions as follows:

(i) $F_{\text{Max-Sleep}}$: aims at maximizing the number of nodes in the *off* mode [Eq. (11.48)].

(ii) $F_{\text{Min-Energy}}$: aims at minimizing the total power consumption in the network [Eq. (11.49)].

(iii) $F_{\text{Min-Channel}}$: aims at minimizing the wavelength channel consumption on each link [Eq. (11.50)].

$$F_{\text{Max-Sleep}} = \sum_i \chi_i, \tag{11.48}$$

$$F_{\text{Min-Energy}} = \frac{1}{\text{Energy}}, \tag{11.49}$$

$$F_{\text{Min-Channel}} = \frac{1}{\max\{\sum_w \lambda_w^{i,j}\}_{i,j}}. \tag{11.50}$$

Upon obtaining the fitness values of the individuals, two individual solutions are selected with respect to the fitness proportionate fashion, i.e., each candidate individual can be selected with the probability p, which is the ratio of its fitness to the sum of the fitness values of all individuals. An individual denotes a solution (tree) prior to wavelength assignment. Two selected individuals are used as the inputs of a crossover function in order to obtain two new solutions with a probability of Θ. The crossover function works as follows: upon finding a common link between the branches of two trees, the rest of the branches are exchanged between the new solutions. The crossover is followed by wavelength assignment to the new solutions in first-fit fashion. For each solution, EAGLE employs an aging counter that is incremented by one at the end of each iteration. During the iteration steps, if the number of the solutions in the solution space exceeds the population size (P), the m oldest solutions are removed from the solution space unless any of them has the highest fitness value in the current solution space.

In [45], the performance of EAGLE has been evaluated by assuming each node has access to at least one data center, and considering the upstream data center demand profile as in Figure11.13a. Four time zones, EST, CST, MST, and PST, are assumed as in [28] to assure heterogeneity of the network traffic, and one day is represented by eight slots where each slot corresponds to a three-hour duration. Energy consumption of EDFAs, MEMs equipment, and the idle power consumption are taken as 8, 1.8, and 150 Wh, respectively. The WR transport network offers 10 Gbps capacity on each wavelength channel while at least two destinations are required to be reached out of either three or four destination candidates. EAGLE has been run with a crossover probability of 0.20 and a mutation probability of 0.01. Figure 11.13 illustrates the cumulative energy consumption of the transport network throughout the day when EAGLE runs under the three fitness functions. As seen in the figure,

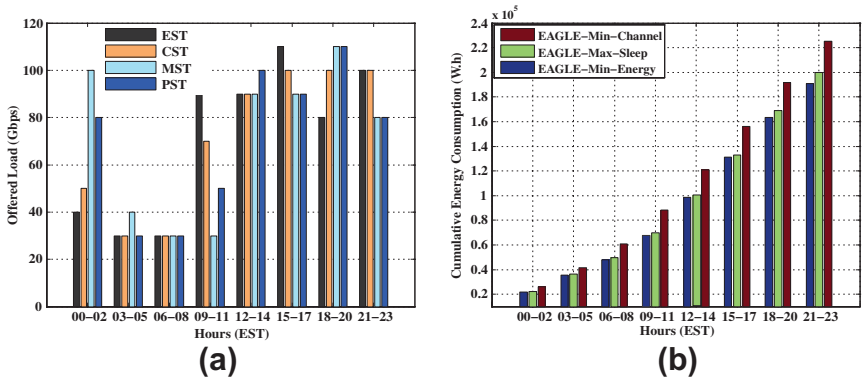

Figure 11.13 (a) Offered load profile in four time zones during the day. (b) Cumulative energy consumption of EAGLE throughout the day with the three fitness functions.

$F_{\text{Min-Channel}}$ (EAGLE-Min-Channel) leads to the highest energy consumption since setting the objective as minimizing the maximum number of wavelength channels per fiber can lead to longer routes during the evolution process. On the other hand, the other fitness functions, namely $F_{\text{Max-Sleep}}$ (EAGLE-Max-Sleep) and $F_{\text{Min-Energy}}$ (EAGLE-Min-Energy) introduce significant energy savings. Moreover, instead of only aiming to avoid the maximum number of pass-through nodes, provisioning the demands by considering the overall energy consumption in the transport network leads to more energy savings at the end of the day.

Since it is based on an evolutionary heuristic, EAGLE provides a suboptimal solution for the energy-efficient provisioning of the cloud services. Hence, further improvements are possible in manycast provisioning by considering more efficient fitness functions. Furthermore, faster heuristics running in dynamic environments are emergent to eliminate the need for lookahead demand profiles since demands are provisioned and released dynamically in cloud computing.

11.6 SUMMARY AND CHALLENGES

By bringing many parallel and distributed system concepts together, cloud computing is expected to serve many business areas such as health [49], education [50], scientific computation [51], multimedia service delivery [52], and so on. Migration of local resources to a shared pool of resources distributed over the Internet introduces many advantages but at the same time increases the communication energy, and consequently increases GHG emissions. Furthermore, data centers that are the main hosts of cloud computing environment emit significantly large amount of GHGs. Hence, energy-efficient and green migration of IT services by cloud computing is emergent.

Table 11.1 Energy Management Solutions in Cloud Computing

Energy Management Solution	Processing	Storage	Transport
Thermal-aware workload placement in data centers	j		
Cooling management in data centers	j		
Dynamic component deactivation	j		
Dynamic voltage and frequency sampling	j		
Data center monitoring	j	j	
Massive array of idle disks (MAIDs)		j	
Solid-state disks (SSDs)		j	
Virtualization by storage area networks		j	
Optimal data center placement	j	j	j
Energy-efficient anycast			j
Energy-efficient manycast			j

This chapter has provided an overview of energy-efficiency issues and the existing solutions. Table 11.1 summarizes the energy management techniques in cloud computing. Energy-efficiency of processing and storage in cloud computing has been studied in the sense of energy management in data centers. Thermal- and cooling-aware solutions for efficient workload placement problems are required to improve data center PUE values. Furthermore, thermal monitoring of data centers is crucial to assist cooling management and workload placement processes. The effect of data center locations on the energy-efficiency of cloud computing has also been reviewed with consideration of the associated transport network energy. Building the data centers close to renewable resources such as wind farms and use of solar panels at the transport network nodes can minimize the dependency on nonrenewable resources and lead to further reduction in GHG emissions of processing, storage, and transport energies of cloud computing. Energy-efficient provisioning of the cloud services over the transport network has been studied by introducing energy-efficient anycast and manycast techniques. Sleep-mode support at the backbone nodes can introduce significant savings.

Despite the existing techniques, there are still open issues for the researchers in this field. Although the existing cloud service providers offer 99.995% availability, this value has to be enhanced for the mission-critical applications of large enterprizes [53]. Enhancement of reliability is possible by replication of content

and redundancy of transport resources, which introduce an increase in network energy consumption. Efficient solutions are needed to address the trade-off between service level agreement (SLA) assurance and energy-efficiency. Furthermore, reliability assurance of data center monitoring systems is another emergent subject in energy-efficient cloud computing. The more reliable the monitoring system, the more precise the decisions made for the workload consolidation among the servers. Dynamic, fast, and scalable anycast and manycast algorithms are required for efficient utilization of the network resources while provisioning the cloud services. Last but not least, energy-efficient design of storage area networks (SANs) is another direction for researchers in this field.

REFERENCES

[1] IBM Autonomic Computing Manifesto, 2001 (online). <http://www.research.ibm.com/autonomic.manifesto/>.
[2] U. Schwiegelshohn et al., Perspectives on grid computing, Future Generation Computer Systems 26 (8) (2010) 1104–1115.
[3] R. Schollmeier, A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications, in: Proceedings of First International Conference on Peer-to-Peer Computing, August 2001, pp. 101–102.
[4] J.W. Ross, G. Westerman, Preparing for utility computing: the role of IT architecture and relationship management, IBM Systems Journal 43 (1) (2004) 5–19.
[5] M. Armbrust et al., Above the Clouds: A Berkeley View of Cloud Computing, UCB/EECS-2009-28, EECS Department, University of California, Berkeley, February 2009 (online). <http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html>.
[6] Q. Zhang, L. Cheng, R. Boutaba, Cloud computing: state-of-the-art and research challenges, Journal of Internet Services and Applications 1, 7–18.
[7] L.M. Vaquero et al., A break in the clouds: towards a cloud definition, ACM SIGCOMM Computer Communication Review 39 (1) (2009) 50–55.
[8] P. Mell, T. Grance, The NIST Definition of Cloud Computing, 800–145, National Institute of Standards and Technology, January 2011 (online). <http://csrc.nist.gov/publications/>.
[9] D. Kondo et al., Cost-benefit analysis of cloud computing versus desktop grids, in: IEEE International Symposium on Parallel Distributed Processing, May 2009, pp. 1–12.
[10] Amazon EC2 Service Level Agreement, 2008 (online). <http://aws.amazon.com/ec2-sla/>.
[11] B. Hayes, Cloud computing, Communications of the ACM 51 (2008) 9–11.
[12] Vic J.R. Winkler, Chapter 2—cloud computing architecture, in: Securing the Cloud, Syngress, Boston, 2011, pp. 29–53 (online). <http://www.sciencedirect.com/science/article/pii/B9781597495929000026>.
[13] J. Baliga et al., Photonics switching and the energy bottleneck, in: OSA Conference on Photonics in Switching, 2007, pp. 125–126.
[14] K. Hinton et al., Power consumption and energy efficiency in the Internet, IEEE Network 25 (2) (2009)
[15] J. Baliga et al., Architectures for energy-efficient IPTV networks, in: Conference on Optical Fiber Communication (OFC), March 2009, pp. 1–3.

[16] J. Baliga et al., Green cloud computing: balancing energy in processing, storage, and transport, Proceedings of the IEEE 99 (1) (2011) 149–167.
[17] M. Gupta, S. Singh, Greening of the Internet, in: ACM SIGCOMM'03, August 2003, pp. 19–26.
[18] P. Leisching, P. Pickavet, Energy footprint of ICTs: forecasts and network solutions, in: OFC/NFOEC, Workshop on Energy Footprint of ICT: Forecast and Network Solutions, 2009.
[19] Y. Zhang et al., Energy efficiency in telecom optical networks, IEEE Communications Surveys and Tutorials 12 (4) (2010) 441–458.
[20] H.T. Mouftah and B. Kantarci, Energy-Aware Systems and Networking for Sustainable Initiatives, in: W-C. Hu, N. Kaabouch (Eds), Greening the Survivable Optical Networks: Solutions and Challenges for the Backbone and Access, IGI Global, pp. 256–286, 2012.
[21] K. Kant, Data center evolution: A tutorial on state of the art, issues, and challenges, Computer Networks 53 (2009) 2939–2965.
[22] H.S. Sun, S.E. Lee, Case study of data centers energy performance, Energy and Buildings 38 (5) (2006) 522–533.
[23] US Environmental Protection Agency ENERGY STAR Program, Report to Congress on Server and Data Center Energy Efficiency, ENERGY STAR, August 2007 (online). <http://www.energystar.gov>.
[24] C. Belady, The green grid data center power efficiency metrics: PUE and DCiE, Whitepaper, 2008 (online). <http://www.thegreengrid.org/>.
[25] S. Greenberg, W. Tshudi, J. Weale, Self-Benchmarking Guide for Data Center Energy Performance, Lawrence Berkley National Laboratory, (Online) <http://www.lbnl.gov/>, 2006.
[26] Data Center Knowledge (online). <http://www.datacenterknowledge.com>, accessed in July 2012.
[27] Google Data Center Efficiency (online). <http://www.google.com/about/datacenters/inside/efficiency/powerusage.html> (accessed October 2011).
[28] X. Dong, T. El-Gorashi, J.M.H. Elmirghani, Green IP over WDM with data centers, IEEE/OSA Journal of Lightwave Technology 29 (12) (2011) 1861–1880.
[29] A. Beloglazov, A taxonomy and survey of energy-efficient data centers and cloud computing systems, Advance in Computers vol. 82, Elsevier.
[30] J. Moore et al., Making scheduling cool: temperature-aware workload placement in data centers, in: Usenix Ann. Technical Conf., 2005, pp. 61–74.
[31] A. Banerjee et al., Sustainable Computing: Informatics and Systems 1 (2) (2011) 134–150.
[32] D. Reinsel and J. Janukowicz, Data center SSDs: solid footing for growth, Whitepaper, 2008 (online). <http://www.samsung.com/global/business/semiconductor/products/SSD/downloads/datacenter ssds.pdf>.
[33] D. Kim et al., Architecture exploration of high-performance PCs with a solid-state disk, IEEE Transactions on Computers 59 (2010) 878–890.
[34] D. Colarelli, D. Grunwald, Massive arrays of idle disks for storage archives, in: SC Conference, 2002, pp. 47–52.
[35] The State of MAID in Data Centers, April 2009 (online). <http://searchstorage.techtarget.com/report/The-state-of-MAID-in-data-centers>.
[36] X. Zhang et al., Key technologies for green data center, in: Third International Symposium on Information Processing (ISIP), October 2010, pp. 477–480.

[37] D. Barrett, G. Kipper, Virtualization challenges, in: Virtualization and Forensics, Syngress, Boston, 2010, pp. 175–195.
[38] IBM System Storage SAN Volume Controller (online). <http://www-03.ibm.com/systems/storage/software/virtualization/svc/index.html>.
[39] J. Liu et al., Project genome: wireless sensor network for data center cooling, The Architecture Journal Microsoft 18 (2008) 28–34.
[40] H. Viswanathan, E.K. Lee, D. Pompili, Self-organizing sensing infrastructure for autonomic management of green datacenters, IEEE Network 25 (4) (2011) 34–40.
[41] G. Shen, R.S. Tucker, Energy-minimized design for IP over WDM networks, IEEE/OSA Journal of Optical Communications and Networking 1 (2009) 176–186.
[42] B.G. Bathula, J.M.H. Elmirghani, Green networks: energy efficient design for optical networks, in: International Conference on Wireless and Optical Communications Networks (WOCN), April 2009.
[43] I. Foster et al., Cloud computing and grid computing 360-degree compared, in: Grid Computing Environments Workshop (GCE), November 2008, pp. 1–10.
[44] N. Charbonneau, V.M. Vokkarane, Routing and wavelength assignment of static manycast demands over all-optical wavelength-routed WDM networks, Journal of Optical Communications and Networking 2 (7) (2010) 442–455.
[45] B. Kantarci, H.T. Mouftah, Energy-efficient cloud services over wavelength-routed optical transport networks, in: Proceedings of IEEE GLOBECOM, December 2011, pp. SAC06.6.1–SAC06.6.5.
[46] B.G. Bathula, M. Alresheedi, J.M.H. Elmirghani, Energy efficient architectures for optical networks, in: London Communications Symposium (LCS), 2009.
[47] B.G. Bathula, J.M.H. Elmirghani, Energy efficient optical burst switched (OBS) networks, in: IEEE Globecom Workshops, 2009.
[48] S.R. Pramod, H.T. Mouftah, A sharable architecture for efficient utilization of transmitter–receiver ports on an optical switch, in: SPIE Photonics, December 2004.
[49] A. Rosenthal et al., Cloud computing: a new business paradigm for biomedical information sharing, Journal of Biomedical Informatics 43 (2) (2010) 342–353.
[50] N. Sultan, Cloud computing for education: a new dawn? International Journal of Information Management 30 (2) (2010) 109–116.
[51] S.N. Srirama, P. Jakovits, E. Vainikko, Adapting scientific computing problems to clouds using MapReduce, Future Generation Computer Systems 28 (1) (2012) 184–192.
[52] W. Shi et al., Sharc: a scalable 3d graphics virtual appliance delivery framework in cloud, Journal of Network and Computer Applications 34 (4) (2011) 1078–1087.
[53] S. Marston et al., Cloud computing the business perspective, Decision Support Systems 51 (1) (2011) 176–189.