NJIT New Jersey's Science & Technology University

RENEWABLE ENERGY-AWARE INTER-DATACENTER VIRTUAL MACHINE MIGRATION OVER ELASTIC OPTICAL NETWORKS

LIANG ZHANG

TAO HAN

NIRWAN ANSARI

TR-ANL-2015-005

AUGUST 26, 2015

ADVANCED NETWORKING LABORATORY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

NEW JERSY INSTITUTE OF TECHNOLOGY

Renewable Energy-Aware Inter-datacenter Virtual Machine Migration over Elastic Optical Networks

Liang Zhang, Tao Han, *Member, IEEE*, and Nirwan Ansari, *Fellow, IEEE*

Abstract

Datacenters (*DC*s) are deployed in a large scale to support the ever increasing demand for data processing to support various applications. The energy consumption of DCs becomes a critical issue. Powering DCs with renewable energy can effectively reduce the brown energy consumption and thus alleviates the energy consumption problem. Owing to geographical deployments of DCs, the renewable energy generation and the data processing demands usually vary in different DCs. Migrating virtual machines (*VM*s) among DCs according to the availability of renewable energy helps match the energy demands and the renewable energy generation in DCs, and thus maximizes the utilization of renewable energy. Since migrating VMs incurs additional traffic in the network, the VM migration is constrained by the network capacity. The inter-datacenter (*inter-DC*) VM migration with network capacity constraints is an NP-hard problem. In this paper, we propose two heuristic algorithms that approximate the optimal VM migration solution. Through extensive simulations, we show that the proposed algorithms, by migrating VM among DCs, can reduce up to 31% of brown energy consumption.

Index Terms

Manycast, Cloud Computing, Elastic Optical Networks.

Liang Zhang and Nirwan Ansari are with the Advanced Networking Laboratory, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, 07102, USA. (Email: {lz284, nirwan.ansari}@njit.edu).

Tao Han was with the Advanced Networking Laboratory, Department of Electrical and Computer Engineering, New Jersey Institute of Technology. He is now with the Department of Electrical and Computer Engineering, University of North Carolina at Charlotte, Charlotte, NC, 28223, USA. (Email: than3@uncc.edu).

I. Introduction

Cloud infrastructures are widely deployed to support various emerging applications such as: Google App Engine, Microsoft Window Live Service, IBM Blue Cloud, and Apple Mobile Me [1]. Large-scale data centers (*DC*s), which are the fundamental engines for data processing, are the essential elements in cloud computing [2], [3]. Information and Communication Technology (*ICT*) is estimated to be responsible for about 14% of the worldwide energy consumption by 2020 [4]. The energy consumption of DCs accounts for nearly 25% of the total ICT energy consumption [4]. Hence, the energy consumption of DCs becomes an imperative problem.

Renewable energy, which includes solar energy and wind power, produces 12.7% domestic electricity of the United States in 2011 [5]. Renewable energy will be widely adopted to reduce the brown energy consumption of ICT [6]. For example, Parasol is a solar-powered DC [7]. In Parasol, GreenSwitch, a management system, is designed to manage the work loads and the power supplies [7]. The availability of renewable energy varies in different areas and changes over time. The work loads of DCs also vary in different areas and at different time. As a result, the renewable energy availability and energy demands in DCs usually mismatch with each other. This mismatch leads to inefficient renewable energy usage in DCs. To solve this problem, it is desirable to balance the work loads among DCs according to their green energy availability. Although the current cloud computing solutions such as cloud bursting [8], VMware and F5 [9] support inter-datacenter (*inter-DC*) virtual machine (*VM*) migration, it is not clear how to migrate VM among renewable energy powered DCs to minimize their brown energy consumption.

Elastic Optical Networks (*EONs*), by employing orthogonal frequency division multiplexing (*OFDM*) techniques, not only provide a high network capacity but also enhance the spectrum efficiency because of the low spectrum granularity [10]. The granularity in EONs can be 12.5 GHz or even much smaller [11]. Therefore, EONs are one of the promising networking technologies for inter-DC networks [12].

Powering DCs with renewable energy can effectively reduce the brown energy consumption, and thus alleviate green house gas emissions. DCs are usually co-located with the renewable energy generation facilities such as solar and wind farms [13]. Since transmitting renewable energy via the power grid may introduce a significant power loss, it is desirable to maximize the utilization of renewable energy in the DC rather than transmitting the energy back to the power grid. In this paper, we investigate the *r*enewable *e*nergy-*a*ware *i*nter-DC VM *m*igration (*RE-AIM*) problem that optimizes the renewable energy utilization by migrating VMs among DCs. Fig. 1 shows the architecture of an inter-DC network. The vertices in the graph stand for the optical switches in EONs. DCs are connected to the optical switches via IP routers [1]. These DCs are powered by hybrid energy including brown energy, solar energy, and wind energy. In migrating VMs among DCs, the background traffic from other applications are also considered in the network. For example, assume that DC 1 lacks renewable energy while DC 2 and DC 3 have superfluous renewable energy. Some VMs can be migrated out of DC 1 in order to save brown energy. Because of the background traffic and the limited network resource, migrating VMs using different paths (Path 1 or Path 2) has different impacts on the network in terms of the probability of congesting the network. It is desirable to select a migration path with minimal impact on the network.

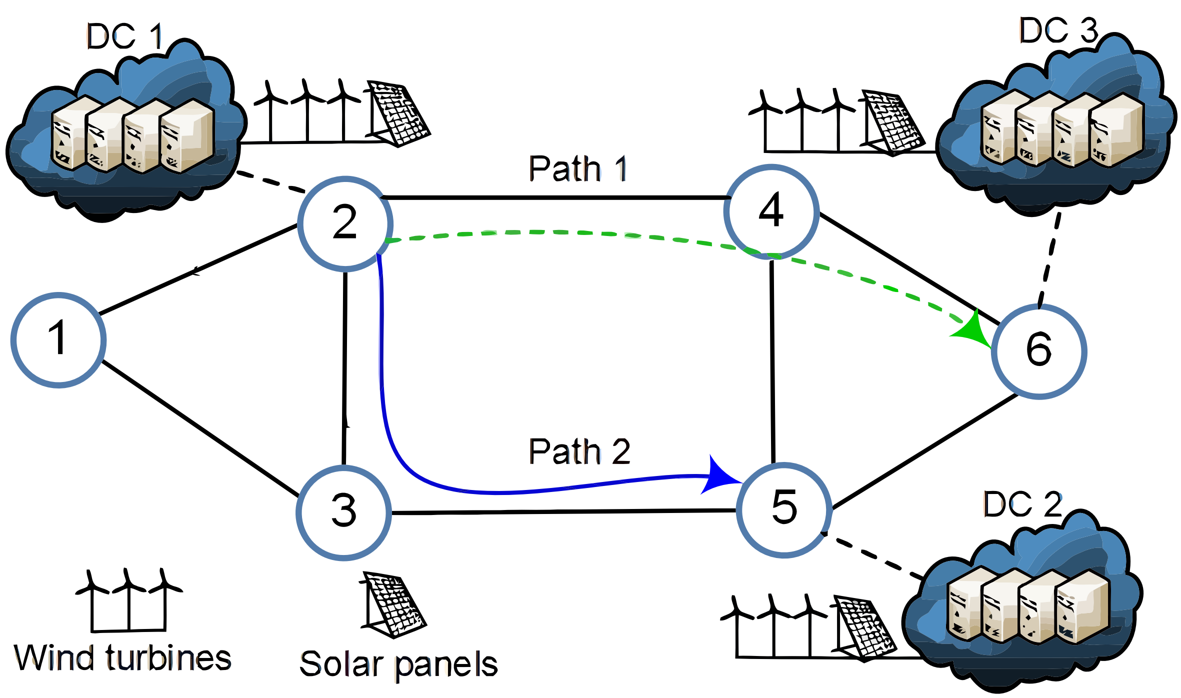

Fig. 1. Inter-DC architecture.

The rest of this paper is organized as follows. Section II describes the related work. Section

[1]In this paper, we focus on the EONs. The design and optimization of the IP networks are beyond the scope of this paper.

III formulates the RE-AIM problem. Section IV briefly analyzes the property of the RE-AIM problem and proposes two heuristic algorithms to solve the problem. Section V demonstrates the viability of the proposed algorithms via extensive simulation results. Section VI concludes the paper.

II. Related Work

Owing to the energy demands of DCs, many techniques and algorithms have been proposed to minimize the energy consumption of DCs [14].

Fang *et al.* [15] presented a novel power management strategy for the DCs, and their target was to minimize the energy consumption of switches in a DC. Cavdar and Alagoz [16] surveyed the energy consumption of server and network devices of intra-DC networks, and showed that both computing resources and network elements should be designed with energy proportionality. In other words, it is better if the computing and networking devices can be designed with multiple sleeping states. A few green metrics are also provided by this survey, such as Power Usage Effectiveness (PUE) and Carbon Usage Effectiveness (CUE).

Deng *et al.* [17] presented five aspects of applying renewable energy in the DCs: the renewable energy generation model, the renewable energy prediction model, the planning of green DCs (i.e., various renewable options, avalabity of energy sources, different energy storage devices), the intra-DC work loads scheduling, and the inter-DC load balancing. They also discussed the research challenges of powering DCs with renewable energy. Ghamkhari and Mohsenian-Rad [14] developed a mathematical model to capture the trade-off between the energy consumption of a data center and its revenue of offering Internet services. They proposed an algorithm to maximize the revenue of a DC by adapting the number of active servers according to the traffic profile. Gattulli *et al.* [18] proposed algorithms to reduce CO_2 emissions in DCs by balancing the loads according to the renewable energy generation. These algorithms optimize renewable energy utilization while maintaining a relatively low blocking probability.

Mandal *et al.* [5] studied green energy aware VM migration techniques to reduce the energy consumption of DCs. They proposed an algorithm to enhance the green energy utilization by migrating VMs according to the available green energy in DCs. However, they did not consider the network constraints while migrating VMs among DCs. In the optical networks, the available spectrum is limited. The large amount of traffic generated by the VM migration may congest the optical networks and increase the blocking rate of the network. Therefore, it is important to consider the network constraints in migrating VMs. In this paper, we propose algorithms to solve the green energy aware inter-DC VM migration problem with network constraints.

III. Problem Formulation

In this section, we present the network model, the energy model, and the formulation of the RE-AIM problem. The key notations are summarized in Table I.

TABLE I

The Important Notations

Symbol	Definiton
c_e	The capacity of a link $e \in E$ in terms of spectrum slots.
c_s	The capacity of a spectrum slot.
c_m	The maximum number of servers in the mth DC.
ς	The maximum number of VMs can be supported in a server.
Φ_m	The amount of renewable energy in the mth DC.
Θ_m	The number of VMs in the mth DC.
α_m	Per unit energy cost for the mth DC.
$\zeta_{m,k}$	The required bandwidth for migrating the kth VM in the mth DC.
$\mathcal{R}$	The set of the migration requests.
$\mathcal{Q}_r$	The set of VMs migrated in the rth migration.
κ	The migration granularity.
$w_p^{r,m}$	The used spectrum slot ratio of the pth path in the rth migration from the mth DC.
w_B	The maximum network congestion ratio.
p_s	The maximum energy consumption of a server.
η	The power usage efficiency.

A. Network Model

We model the inter-DC network by a graph, $\mathcal{G}(V, E, B)$. Here, V, E and B are the node set, the link set and the spectrum slot set, respectively. The set of DC nodes is denoted as $\mathcal{D}$. We assume that all DCs are powered by hybrid energy. We denote $\mathcal{D}_s$ as the set of DCs that does not have sufficient renewable energy to support their work loads and $\mathcal{D}_d$ as the set of DCs that has surplus renewable energy. During the migration, $\mathcal{D}_s$ and $\mathcal{D}_d$ correspond to the two sets of DCs acting as the sources and destinations, respectively. We define κ as the migration granularity, which determines the maximum routing resource that can be used in one migration to each DC.

B. Energy Model

We assume that there are c_m servers in the mth DC and each server can support up to ς VMs. The energy consumption of a server is p_s when it is active. A server is active as long as it hosts at least one active VM; otherwise, the server is in the idle state. Here, we assume that an idling server will be turned off and its energy consumption is zero. Then, $\lceil \varTheta_m/\varsigma \rceil$ is the number of active servers required in the mth DC [5]. We denote η as the power usage effectiveness, which is defined as the ratio of a DC's total energy consumption (which includes the facility energy consumption for cooling, lighting, etc. [19]) to that of the servers in the DC. Given η, a DC's total energy consumption is $\eta \cdot p_s \cdot \varTheta_m/\varsigma$. We denote $\varUpsilon_m$ as the brown energy consumption in the mth DC. Then,

$$\varUpsilon_m = \max(0, \eta \cdot p_s \cdot \lceil \varTheta_m/\varsigma \rceil - \varPhi_m) \tag{1}$$

C. Problem Formulation

In the problem formulation, $\chi_{p,f}^{m,k}$ is a binary variable. $\chi_{p,f}^{m,k} = 1$ indicates that the kth VM in the mth DC is migrated using the p path with the fth spectrum slot as the starting spectrum slot. The objective of the RE-AIM problem is to minimize the total brown energy cost in all DCs with the VM service constraints and the network resource constraints. The problem is formulated as:

$$\min_{\chi_{p,f}^{m,k}} \quad \sum_{m} \alpha_m \cdot \Upsilon_m \tag{2}$$

$$s.t.:$$

$$VM\ service\ constraints:$$

$$\sum_{m}\sum_{k}\sum_{p}\sum_{f} \chi_{p,f}^{m,k} = \sum_{m} \Theta_m \tag{3}$$

$$\sum_{k}\sum_{p}\sum_{f} \chi_{p,f}^{m,k} \leq c_m, \forall m \in \mathcal{D}_s \tag{4}$$

$$\sum_{m' \in \mathcal{D}_s}\sum_{k}\sum_{p}\sum_{f} \chi_{p,f}^{m',k} + \sum_{k}\sum_{p}\sum_{f} \chi_{p,f}^{m,k} \leq c_m, \forall m \in \mathcal{D}_d \tag{5}$$

$$Network\ resource\ constraints:$$

$$w_p^{r,m} + \frac{\Gamma_{p,f}^{r,m}}{c_e} \leq w_B, \quad \forall m \in \mathcal{D}_s, r \in \mathcal{R} \tag{6}$$

$$f(\chi_{p,f}^{m,k}) + b(\chi_{p,f}^{m,k}) \leq c_e \tag{7}$$

$$f(\chi_{p,f}^{m,k}) + b(\chi_{p,f}^{m,k}) - f(\chi_{p,f}^{m,k+1}) \leq 0 \tag{8}$$

$$f(i) + b(i) - f(j) \leq [2 - \delta_{i,j} - y(i,j)] \cdot F_{max}, \quad \forall i \neq j \tag{9}$$

$$f(j) + b(j) - f(i) \leq [1 + \delta_{i,j} - y(i,j)] \cdot F_{max}, \quad \forall i \neq j \tag{10}$$

Here, Eqs. (3)-(5) are the VM service constraints. Eq. (3) constrains that all the VMs should be hosted in the DCs, while Eqs. (4)-(5) constrain that the total number of VMs in a DC should not exceed the DCs' capacity. The network resource constraints are shown in Eqs. (6)-(10).

Eq.(6) constrains the network congestion ratio to be less than w_B, which is the maximum network congestion ratio allowed for routing in the network. In Eq. (6), $w_p^{r,m}$ is the spectrum slot ratio of the pth path in the rth migration from the mth DC, which is defined as the ratio of the number of occupied spectrum slots in the pth path to the total number of spectrum slots of this path. $\Gamma_{p,f}^{r,m}$ is defined as the number of spectrum slots used in the pth path for the rth migration from the mth DC. Eq.(7) is a link capacity constraint of the network; it constrains the bandwidth used in migrating VMs not to exceed the capacity of the network resource. Here, $b(\cdot)$ is the bandwidth requirement in terms of spectrum slots, and $f(\cdot)$ is the index of the starting spectrum slot of a path. For example, $f(\chi_{p,f}^{m,k})$ represents the starting spectrum slot index of the path, which is used by $\chi_{p,f}^{m,k}$. Eq.(8) is the spectrum non-overlapping constraint of a path used by two different VMs in one migration. This constraint must be met for each VM in every migration; if two VMs use the same spectrum slot in one migration, the total bandwidth allocated to the two VMs should not exceed the capacity of a spectrum slot; otherwise, each VM must use a unique spectrum slot. In the migration, the VMs are sorted in ascending order based on their bandwidth requirement. We assume the VMs are migrated according to an ascending order; for example, the $(k+1)$th VM is moved after the kth VM is migrated.

Eqs. (9)-(10) are the spectrum non-overlapping and the continuity constraints [20]. This spectrum non-overlapping constraint is used for different paths. In these constraints, i and j represent two different paths used in the migration. Here, F_{max} is the upper bound of the total bandwidth requirement in terms of spectrum slots. $\delta_{i,j}$ $(\forall i \neq j)$ is a Boolean variable defined in Eq. (11), which equals 1 if the starting spectrum slot index of the ith path is smaller than that of the jth path; otherwise, it is 0. We define $y(i,j)$ $(\forall i \neq j)$ as a Boolean indicator, which equals 1 if the ith path and the jth path in the migration have at least one common link; otherwise, it is 0. We give an example to illustrate these equations. If $y(i,j) = 1$ and $\delta_{i,j} = 1$, Eq. (9) becomes Eq. (12), which ensures the bandwidth non-overlapping constraint. Eq. (10) is automatically satisfied in this case.

$$\delta_{i,j} = \begin{cases} 1, & f(i) < f(j) \\ 0, & f(i) \geq f(j) \end{cases} \tag{11}$$

$$f(i) + b(i) \leq f(j) \tag{12}$$

When we provision spectrum slots for requests in the EONs, the path continuity constraint, spectrum continuity constraint and non-overlapping constraint must be considered. For the path continuity constraint, a lightpath must use the same subcarriers in the whole path for a request. For the spectrum continuity constraint, the used subcarriers must be continuous if a request needs more than one subcarriers. For the non-overlapping constraint, two different lightpaths must be assigned with different subcarriers if they have one or more common links. Since we use a path based method to formulate the RE-AIM problem, the path continuity constraint of the network is already taken into account.

The main contribution of this paper is considering the network influence on the migration when we minimize the brown energy consumption of the DCs. In other words, we want to impose a controllable effect on the network in the migration that leads to less network congestion.

IV. Problem Analysis and Heuristic Algorithms

A. *Problem Analysis*

To solve the RE-AIM problem, both the energy costs in DCs and the network resource required for the migration should be considered. For example, when a DC consumes brown energy, it is desirable to migrate some VMs to other DCs. The VM migration will introduce additional traffic to the network. To avoid congesting the network, we have to optimize the number of VMs that will be migrated and select the routing path for the migration. Therefore, it is challenging to solve the RE-AIM, which is proven to be NP-hard.

Lemma 1. *The RE-AIM problem is NP-hard*

Proof: We prove that the RE-AIM problem is NP-hard by reducing any instance of the multi-processor scheduling problem (*MPS*) into the RE-AIM problem. ■

In the RE-AIM problem, without considering the network constraints, the optimal number of VMs hosted in the DCs can be derived according to the availability of the renewable energy. However, with the consideration of the network constraints and the background traffic, it is difficult and impossible to solve the RE-AIM problem online. For the RE-AIM problem, many VMs are migrated from a set of DCs (source DCs) to another set of DCs (destination DCs). Therefore, we can model the VM migration problem as a manycast problem. Since the RE-AIM problem is NP-hard, we propose heuristic algorithms to solve this problem. These algorithms determine which VM should be migrated to which DC and select a proper routing path in the network to avoid congesting the network. We consider two network scenarios. The first one is a network with light traffic load. Under this network scenario, we design Manycast with Shortest Path Routing (*Manycast-SPR*) algorithm for VM migrations. The second network scenario is a network with heavy traffic load. In this case, we propose Manycast Least-Weight Path Routing (*Manycast-LPR*) for migrating VMs among DCs.

B. Heuristic Algorithms For Light Work Loads

When the network load is light, there are more available spectrum slots. It is easy to find a path with available spectrum slots for the migration requests. Then, a lower computing complexity algorithm is preferred. Manycast-SPR only uses the shortest path, and thus it is a very simple algorithm. Hence, Manycast-SPR is expected to provision the inter-DC VM migration requests in a network with light work loads.

The Manycast-SPR algorithm, as shown in Alg. 1, is to find the shortest routing path that satisfies the VM migration requirement and the network resource constraints. In the beginning, we input $\mathcal{G}(V, E, B)$, $\varTheta_m$ and $\varPhi_m$, and then calculate the optimal work loads distribution. Afterward, we get $\mathcal{D}$, $\mathcal{D}_s$ and $\mathcal{D}_d$. Then, we collect the migration requests R. Here, our algorithm splits the manycast requests into many anycast requests $r \in R$. Now, we start to find a source DC s and a destination DC d for the request r. The migration will try to use the shortest path p from s to d; the request r is carried out if the network congestion constraint is satisfied; otherwise, the request is denied. Then, we update $\mathcal{D}_s$ and $\mathcal{D}_d$ for the next request. After many rounds of migration, if $\mathcal{D}_s$ or $\mathcal{D}_d$ is empty, or Eq. (6) is not satisfied, the migration is completed.

Details of the Manycast-SPR algorithm is described in *Algorithm* 1. Here, $p(\cdot)$ is a function which targets to get the path for the migration. The complexity of Manycast-SPR is $O(|B||E|^2|\mathcal{R}||Q_r||\mathcal{D}|^2 c_m \varsigma)$. Here, $O(|\mathcal{D}|^2 c_m \varsigma)$ is the complexity to determine the optimal work loads, $O(|B|)$ is the complexity to determine f, and $O(|\mathcal{R}||Q_r|)$ is the complexity in building the VM set for the migration. $O(|E|^2)$ is the complexity of determining the path p for Manycast-SPR.

C. Heuristic Algorithms For Heavy Work Loads

When the work load of the network is heavy, the number of available spectrum slots in the network is limited. Since Manycast-SPR only uses the shortest path (one path) for routing, it is impossible for Manycast-SPR to find an available path and spectrum slots in this scenario. Then, Manycast-SPR may block the migration request, and leads to high brown energy consumption of DCs. Hence, we propose another algorithm Manycast-LPR to achieve better routing performance, that results in low brown energy consumption. Manycast-LPR checks K-shortest paths from the source node to the destination node, and picks up the idlest path to serve the requests. The requests will be provisioned with a higher probability by Manycast-LPR as compared to Manycast-SPR. In summary, Manycast-LPR is expected to provision the inter-DC VM migration requests under a heavy work load. It targets to find a path with more available spectrum slots at the expense of a higher complexity.

Manycast-LPR, as shown in Alg. 2, is to find the least weight routing path that satisfies the VM migration requirement and the network resource constraints. The main difference between Manycast-LPR and Manycast-SPR is using different ways to find a path. For Manycast-SPR, it first determines the source node and the destination node. Manycast-LPR, however, finds the path first, then uses the path to find the source node and the destination node. The other steps are almost the same. Since Manycast-LPR should calculate the weights for all node pairs to find a path, it increases the complexity.

Details of the Manycast-LPR algorithms is described in *Algorithm* 2. The complexity for Manycast-LPR is $O(K|B||E|^2|\mathcal{R}||Q_r||\mathcal{D}|^3 c_m \varsigma)$. Here, $p(\cdot)$ is a function which targets to get the path for the migration. $O(|\mathcal{D}|^2 c_m \varsigma)$ is the complexity to determine the optimal work loads, $O(|B|)$ is the complexity to determine f, and $O(|\mathcal{R}||Q_r|)$ is the complexity in building the VM set for the migration. $O(K|E|^2|\mathcal{D}|)$ is the complexity of determining the path p for Manycast-LPR. The most complex part is to determine the set of VMs for the migration.

V. Performance Evaluations

We evaluate the proposed algorithms for the RE-AIM problem in this section. In order to make the RE-AIM problem simple, we assume migratory VMs can be completed in one time slot. The NSFNET topology, shown in Fig. 2, is used for the simulation. There are 14 nodes, and the DCs are located at $\mathcal{D} = \{3, 5, 8, 10, 12\}$ [21], [22]. The DCs are assumed to be equipped with wind turbines and solar panels, which provide the DCs with renewable energy, as shown in Fig 2. The constant α is randomly generated from $[1.6, 3.2]$ and represents the varying price of the electric grid. The capacity of a spectrum slot b is set to 12.5Gps. The maximum number of slots c_e is set to 300; 300 spectrum slots are available when the network is empty. Assume ς equals to 10; 10 VMs can be run in one server. K is set to 3, i.e., the maximum number of shortest paths that can be used in Manycast-LPR is 3. Without losing generality, the average energy consumption of a VM is assumed to be 1 unit, implying that p_s equals to 10 units. The

Algorithm 1: Manycast with Shortest Path Routing

Input : $\mathcal{G}(V, E, B)$, Θ_m and Φ_m;

Output: $\mathcal{D}$, $\mathcal{Q}_r$, $p(\mathcal{Q}_r)$, $f(\mathcal{Q}_r)$ and $\Gamma^r_{p,f}$, $r \in R$;

Build $\mathcal{D}$, $\mathcal{D}_s$ and $\mathcal{D}_d$ by the the optimal work loads allocation;
Collect manycast requests R;
while *$\mathcal{D}_s$ and $\mathcal{D}_d$ are not empty* **do**
    calculate the network congestion ratio for all p, and get w_B;
    for *all nodes $s \in \mathcal{D}_s$* **do**
        find s with the max migratory VMs as the source node;
    for *all nodes $d \in \mathcal{D}_d$* **do**
        find d with the max available renewable energy as the destination node;
    get the shortest path $p(\mathcal{Q}_r)$ from s to d for the rth migration;
    build $\mathcal{Q}_r$ for the rth migration according to $p(\mathcal{Q}_r)$ and Eqs. (9)-(10);
    if *Eq. (6) is satisfied* **then**
        path $p(\mathcal{Q}_r)$ is used to migrate;
        find the start spectrum slot index $f(\mathcal{Q}_r)$ in B ;
        get the allocated bandwidth $\Gamma^r_{p,f}$;
        update $\mathcal{D}_s$ and $\mathcal{D}_d$;
    else
        return;

VM bandwidth requirement $\zeta_{m,k}$ is randomly selected from $[1, 14]$, which is convenient for the simulation. The migration requests are generated by the optimal work loads distribution which is calculated based on Θ_m and Φ_m. The background traffic is randomly generated between node pairs in the network. The background traffic load is counted as an average of $\frac{\lambda}{\mu}$, where λ is an average arrival rate of the requests and $\frac{1}{\mu}$ is the holding period of each request [21]. Here, the background traffic arriving process is a poisson process, and the holding time is a negative exponential distribution. Parameters which are used for the evaluation are summarized in Table

Algorithm 2: Manycast with Least-weight Path Routing

Input : $\mathcal{G}(V, E, B)$, Θ_m and Φ_m;

Output: $\mathcal{D}$, $\mathcal{Q}_r$, $p(\mathcal{Q}_r)$, $f(\mathcal{Q}_r)$ and $\Gamma^r_{p,f}$, $r \in R$;

Build $\mathcal{D}$, $\mathcal{D}_s$ and $\mathcal{D}_d$ by the the optimal work loads allocation;
Collect manycast requests R;
while *$\mathcal{D}_s$ and $\mathcal{D}_d$ are not empty* **do**
    calculate the network congestion ratio for all p, and get w_B;
    for *all nodes $s \in \mathcal{D}_s$* **do**
        for *all nodes $d \in \mathcal{D}_d$* **do**
            build K-shortest path set $\mathcal{P}$;
            for *path $p \in \mathcal{P}$* **do**
                get path $p(\mathcal{Q}_r)$ with the lowest congestion ratio for the rth migration;
    build $\mathcal{Q}_r$ for the rth migration according to $p(\mathcal{Q}_r)$ and Eqs. (9)-(10);
    if *Eq. (6) is satisfied* **then**
        path $p(\mathcal{Q}_r)$ is used to migrate;
        find the start spectrum slot index $f(\mathcal{Q}_r)$ in B;
        get the allocated bandwidth $\Gamma^r_{p,f}$;
        update $\mathcal{D}_s$ and $\mathcal{D}_d$;
    else
        return;

II.

We run the simulation for 150 times, and exclude the scenario with empty VM requests traffic load ($\mathcal{D}_s \neq \varnothing \quad \& \quad \mathcal{D}_d \neq \varnothing$). Fig. 3 shows the total cost of brown energy consumption of the strategy without using renewable energy, Manycast-SPR $(\kappa = 2)$ and Manycast-LPR $(\kappa = 2)$. Apparently, Manycast-SPR and Manycast-LPR can save brown energy substantially. Manycast-SPR saves about 15% cost of brown energy as compared with the strategy without migration. Manycast-LPR reduces up to 31% cost of brown energy as compared with the strategy

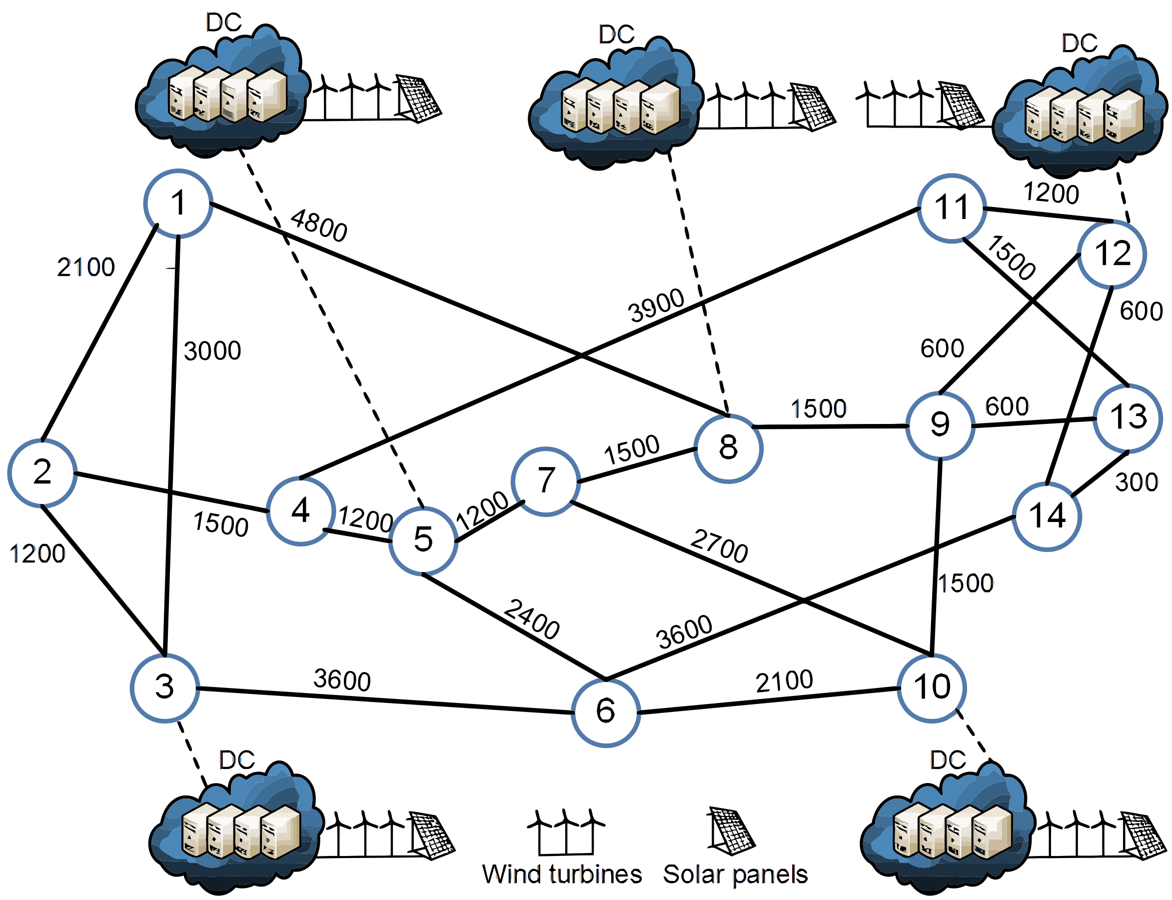

Fig. 2. NSFNET topology with renewable energy DCs.

without migration. Manycast-LPR has better performance because Manycast-LPR employs the least weight path p of all node pairs for routing, while Manycast-LPR engages only the short path p of one node pair.

In order to obtain a better analysis, the running time of Manycast-SPR $(\kappa = 2)$ and Manycast-LPR $(\kappa = 2)$ are shown in Fig. 4. Manycast-SPR spends less time than Manycast-LPR, implying that Manycast-SPR has a lower complexity and Manycast-LPR has a higher computing complexity. It also illustrates that the time and the final cost value is a trade-off in the evaluation. Manycast-LPR is more complex and hence incurs a lower brown energy cost.

The results of Manycast-SPR for various κ are described in Fig. 5. The cost of brown energy consumption keeps increasing when the background traffic increases, because high background traffic tends to congest the network links and leads to more migration failures. Apparently, a

TABLE II
SIMULATION PARAMETERS

Network topology	NSFNET
$\mathcal{D}$	$\{3, 5, 8, 10, 12\}$
ς	10 VMs
c_m	1000 servers
$\alpha = \{\alpha_1, \alpha_2, ..., \alpha_m\}$	$\{2.1, 2.5, 1.9, 2.8, 2\}$
Θ_m	$[0, 8000]$
Φ_m	$[1000, 9000]$
p_s	10 units, 1 unit for 1 VM in average
$\zeta_{m,k}$	$[1, 14]$ Gb/s
c_e	300 spectrum slots
c_s	12.5 Gbps
κ	$\{2, 4, 8, 16\}$ spectrum slots
$\frac{\lambda}{\mu}$	$\{40, 80, 120, 160, 200, 240, 280, 320\}$

small κ brings more benefits than a big κ in reducing the energy cost.

Fig. 6 shows the results of Manycast-LPR for various κ, almost the same results as shown in Fig. 5, but the cost of the brown energy consumption is much less than that in Fig. 5, because Manycast-LPR can easily find a path which has available bandwidth for migration. Obviously, Manycast-LPR with $\kappa = 2$ achieves the best result with the lowest cost of consumed brown energy. All these results illustrate that a small κ leads to a lower cost of the brown energy consumption and a big κ induces a higher cost of the brown energy consumption. This is because it is difficult to find a path with enough bandwidth for a big κ, when the network has background traffic. A smaller κ achieves a lower energy cost at the cost of higher complexity.

Figs. 7 and 8 show the running time of Manycast-SPR and that of Manycast-LPR with different κ, respectively. We can observe that the computing time is decreased when the traffic load increases. For the same κ with a given background traffic load, Manycast-SPR consumes more time than Manycast-LPR does. For either of the two algorithms under a specific background traffic load, we can see that the running time is nearly halved when κ is doubled. Hence, a smaller κ brings a better performance but takes longer time, and a larger κ has worse performance with a shorter running time.

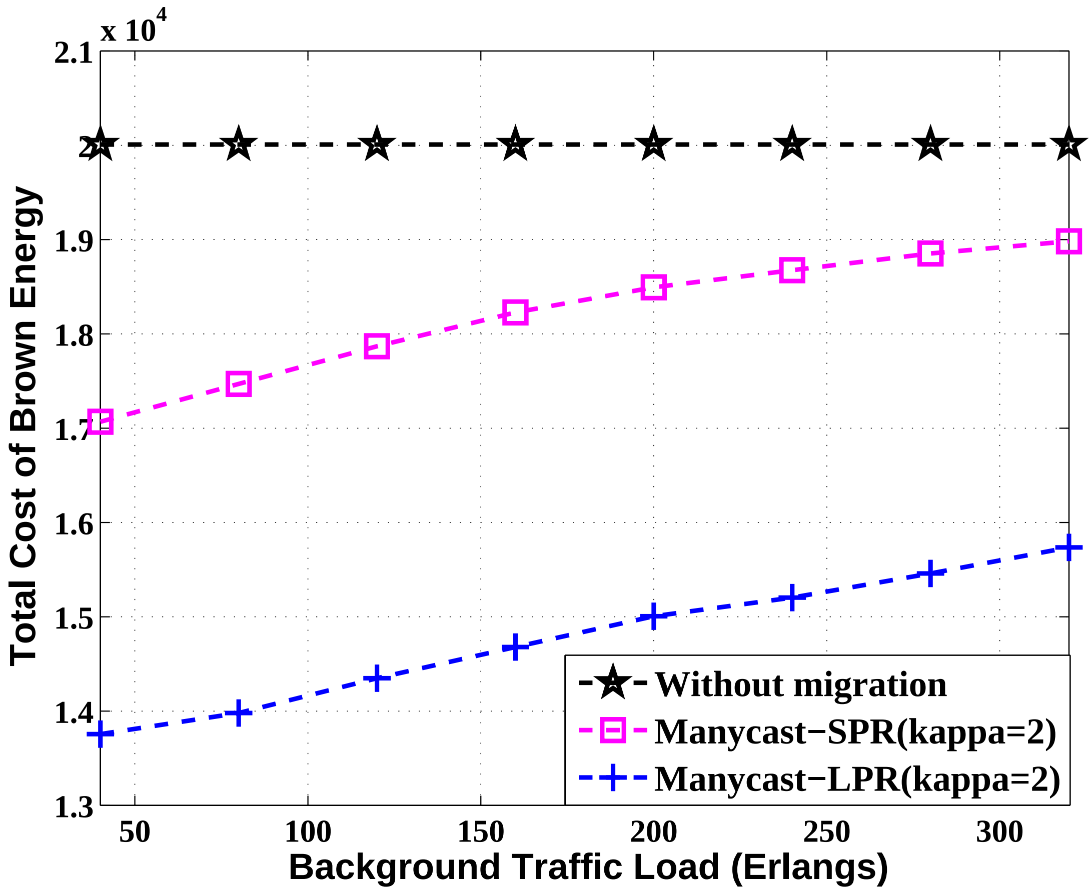

Fig. 3. Total brown energy cost comparison.

VI. Conclusion

Datacenters are widely deployed for the increasing demands of data processing and cloud computing. The energy consumption of DCs will take up 25% of the total ICT energy consumption by 2020. Powering DCs with renewable energy can help save brown energy. However, the availability of renewable energy varies by locations and changes over time, and DCs' work loads demands also vary by locations and time, thus leading to the mismatch between the renewable energy supplies and the work loads demands in DCs. Inter-DC VM migration brings additional traffic to the network, and the VM mitigation is constrained by the network capacity, rendering inter-DC VM migration a great challenge.

This paper addresses the emerging renewable energy-aware inter-DC VM migration problem.

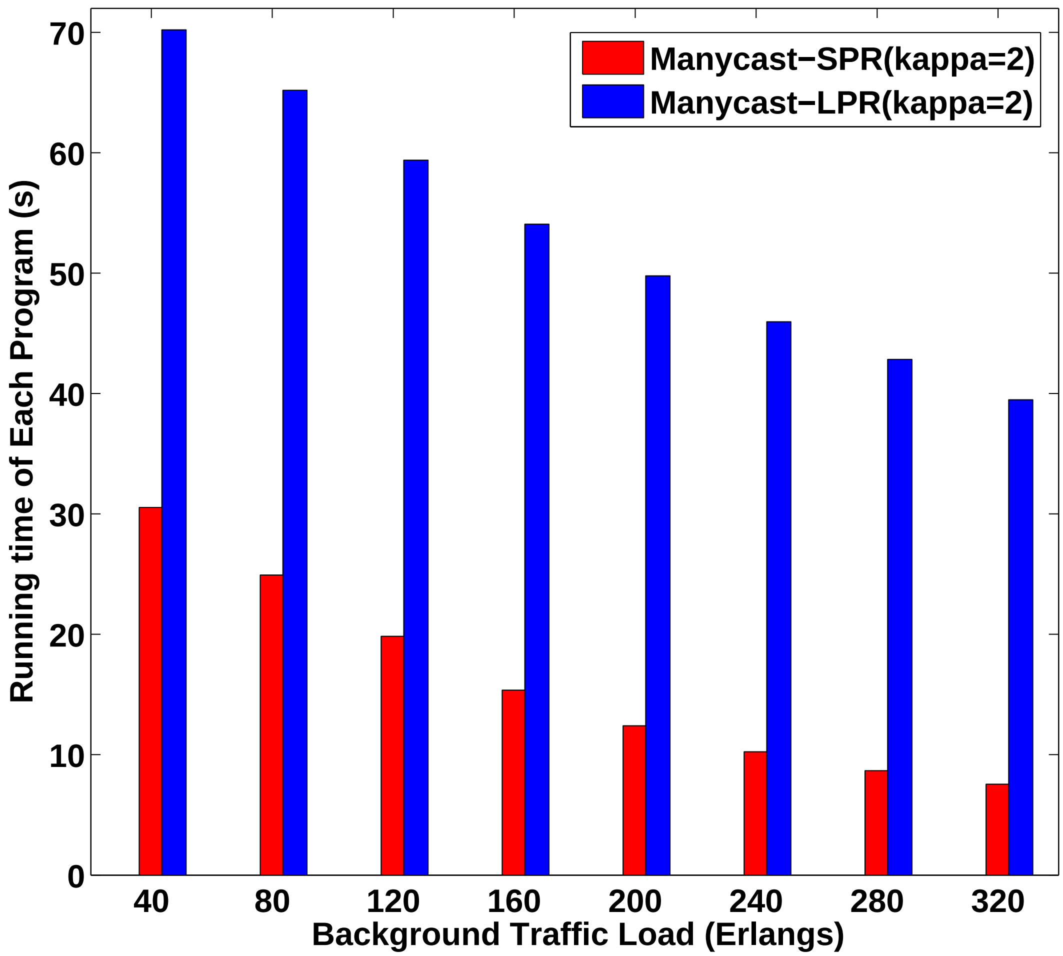

Fig. 4. Running time comparison.

The main contribution of this paper is to reduce the network influence on the migration while minimizing the brown energy consumption of the DCs. The RE-AIM problem is formulated and proven to be NP-hard. Two heuristic algorithms, Manycast-SPR and Manycast-LPR, have been proposed to solve the RE-AIM problem. Our results show that Manycast-SPR saves about 15% cost of brown energy as compared with the strategy without migration, while Manycast-LPR saves about 31% cost of brown energy as compared with the strategy without migration. The computing time of Manycast-LPR is longer than that of Manycast-SPR because the complexity of Manycast-LPR is higher than Manycast-SPR. In conclusion, we have demonstrated the viability

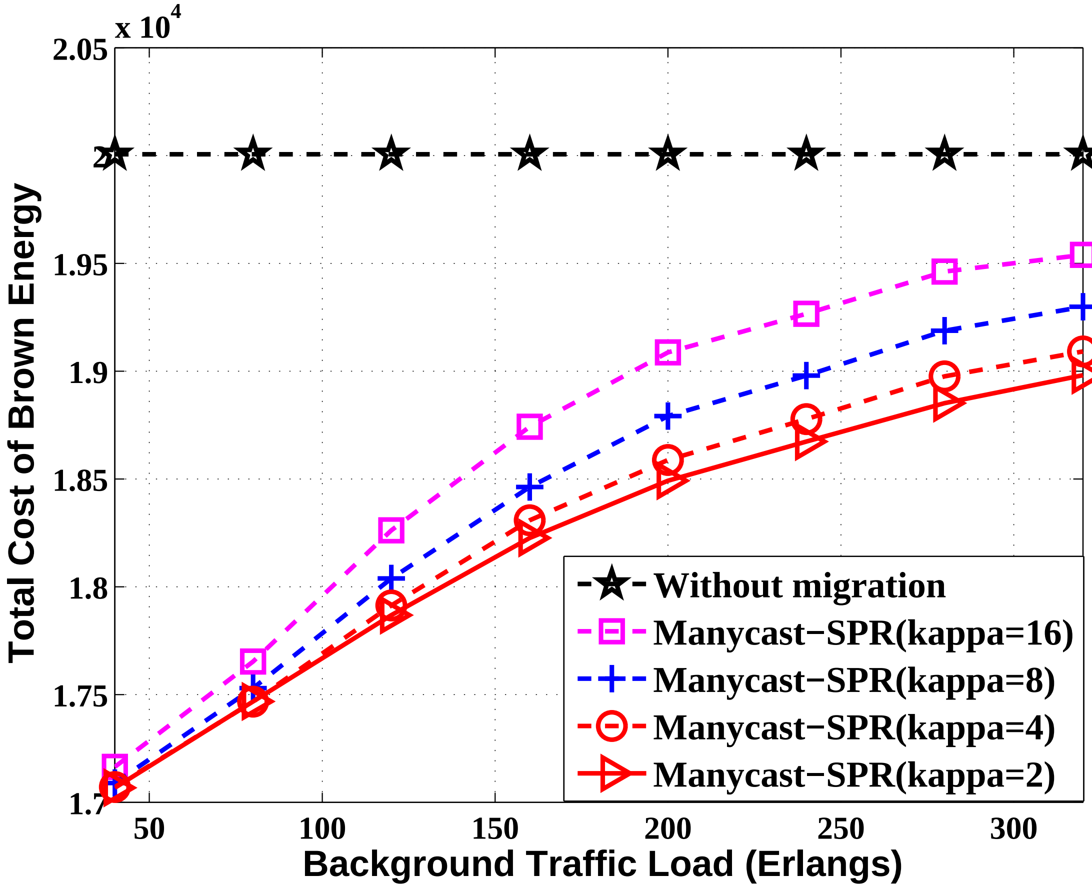

Fig. 5. Total brown energy cost of Manycast-SPR.

of the proposed algorithms in minimizing brown energy consumption in inter-DC migration without congesting the network.

REFERENCES

[1] M. Sadiku, S. Musa, and O. Momoh, "Cloud computing: Opportunities and challenges", *IEEE Potentials*, vol. 33, no. 1, pp. 34-36, Jan. 2014.

[2] Y. Zhang and N. Ansari, "HERO: Hierarchical energy optimization for data center networks", *IEEE Systems Journal*, vol. 2, no. 9, pp. 406-415, Jun. 2015.

[3] Y. Zhang and N. Ansari, "On architecture design, congestion notification, TCP incast and power consumption in data centers, *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 39-64, Jan. 2013.

[4] M. Pickavet, *et al.*, "Worldwide energy needs for ICT: The rise of poweraware networking", in Proc. *ANTS 2008*, pp. 1-3, Dec. 2008.

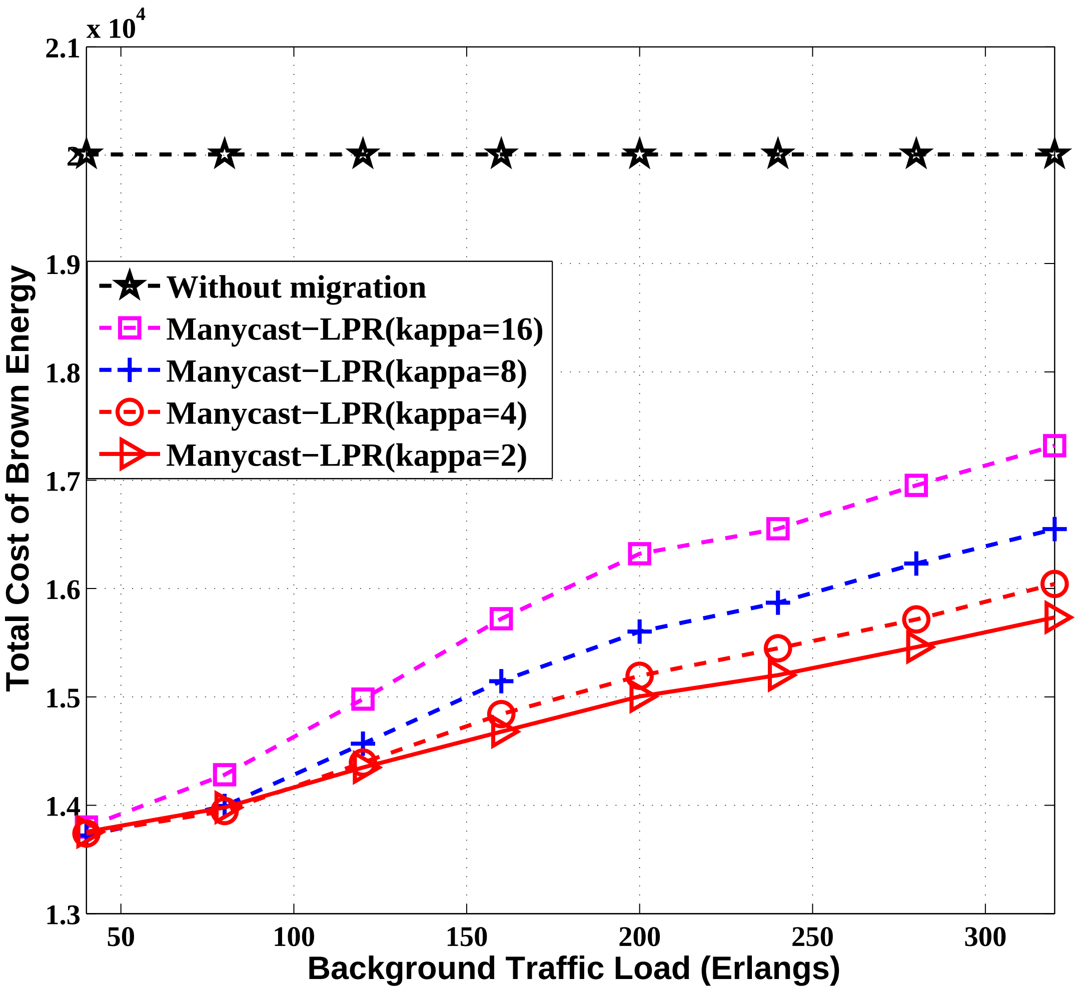

Fig. 6. Total brown energy cost of Manycast-LPR.

[5] U. Mandal, *et al.*, ”Greening the cloud using renewable-energy-aware service migration”, *IEEE Network*, vol. 27, no. 6, pp. 36-43, Nov. 2013.

[6] T. Han and N. Ansari, ”Powering mobile networks with green energy”, *IEEE Wireless Communications*, vol. 21, no. 1, pp. 90-96, Feb. 2014.

[7] I. Goiri, *et al.*, ”Designing and managing data centers powered by renewable energy”, *IEEE Micro*, vol. 34, no. 3, pp. 8-16, May 2014.

[8] T. Wood, *et al.*, ”CloudNet: Dynamic pooling of cloud resources by live WAN migration of virtual machines”, *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1-16, Aug. 2014.

[9] Enabling long distance live migration with f5 and VMware vMotion.[Online]. Available: https://f5.com/resources/white-papers/enablinglong-distance-live-migration-with-f5-and-vmware-vmotion.

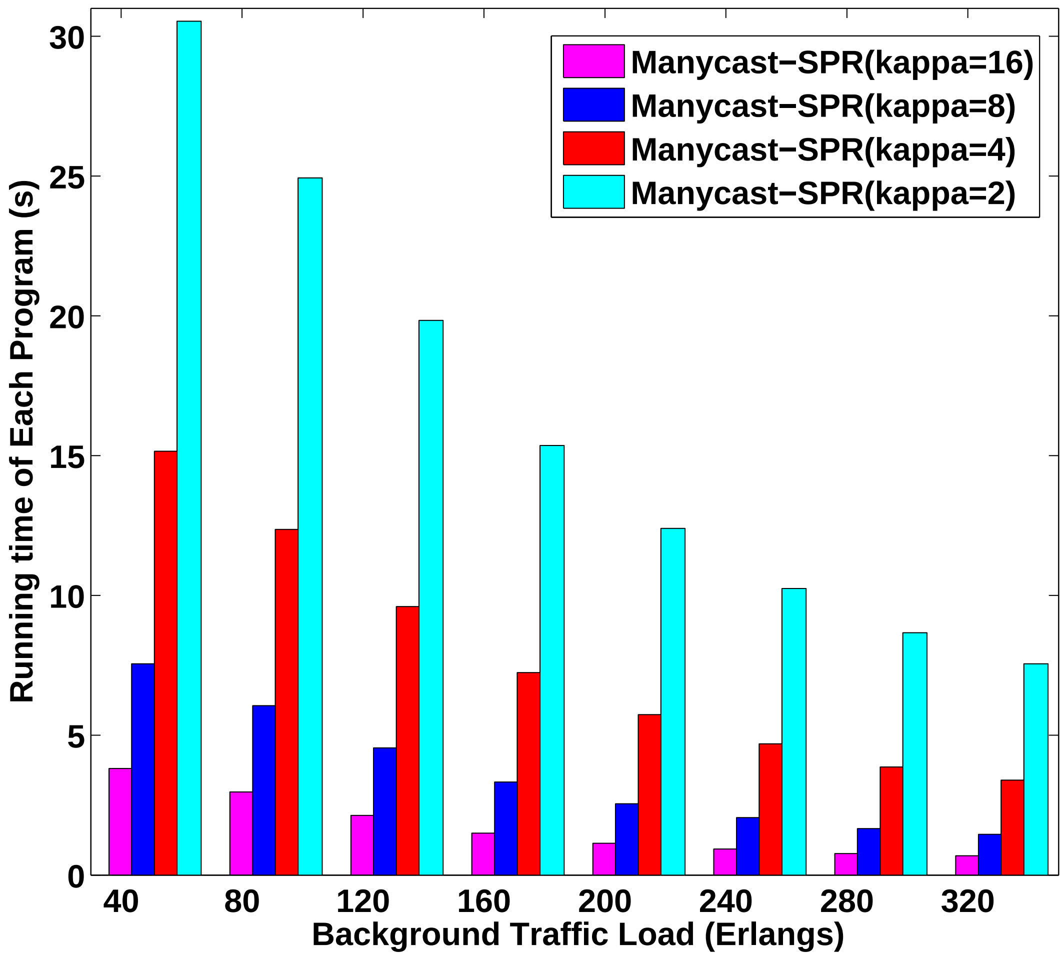

Fig. 7. Running time of Manycast-SPR.

[10] W. Shieh, X. Yi, and Y. Tang, "Transmission experiment of multi-gigabit coherent optical OFDM systems over 1000km ssmf fibre", *Electronics Letters*, vol. 43, no. 3, pp. 183-184, Feb. 2007.

[11] J. Armstrong, "OFDM for optical communications", *J. Lightw. Technol.*, vol. 27, pp. 189-204, Feb. 2009.

[12] C. Develder, *et al.*, "Optical networks for grid and cloud computing applications", *Proceedings of the IEEE*, vol. 100, pp. 1149-1167, May 2012.

[13] S. Figuerola, *et al.*, "Converged optical network infrastructures in support of future internet and grid services using IaaS to reduce GHG emissions", Journal of Lightwave Technology, vol. 27, no. 12, pp. 1941-1946, Jun. 2009.

[14] M. Ghamkhari and H. Mohsenian-Rad, "Energy and performance management of green data centers: A profit maximization approach", *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 1017-1025, Jun. 2013.

[15] S. Fang, *et al.*, "Energy optimizations for data center network: Formulation and its solution", in Proc. *GLOBECOM*, pp.

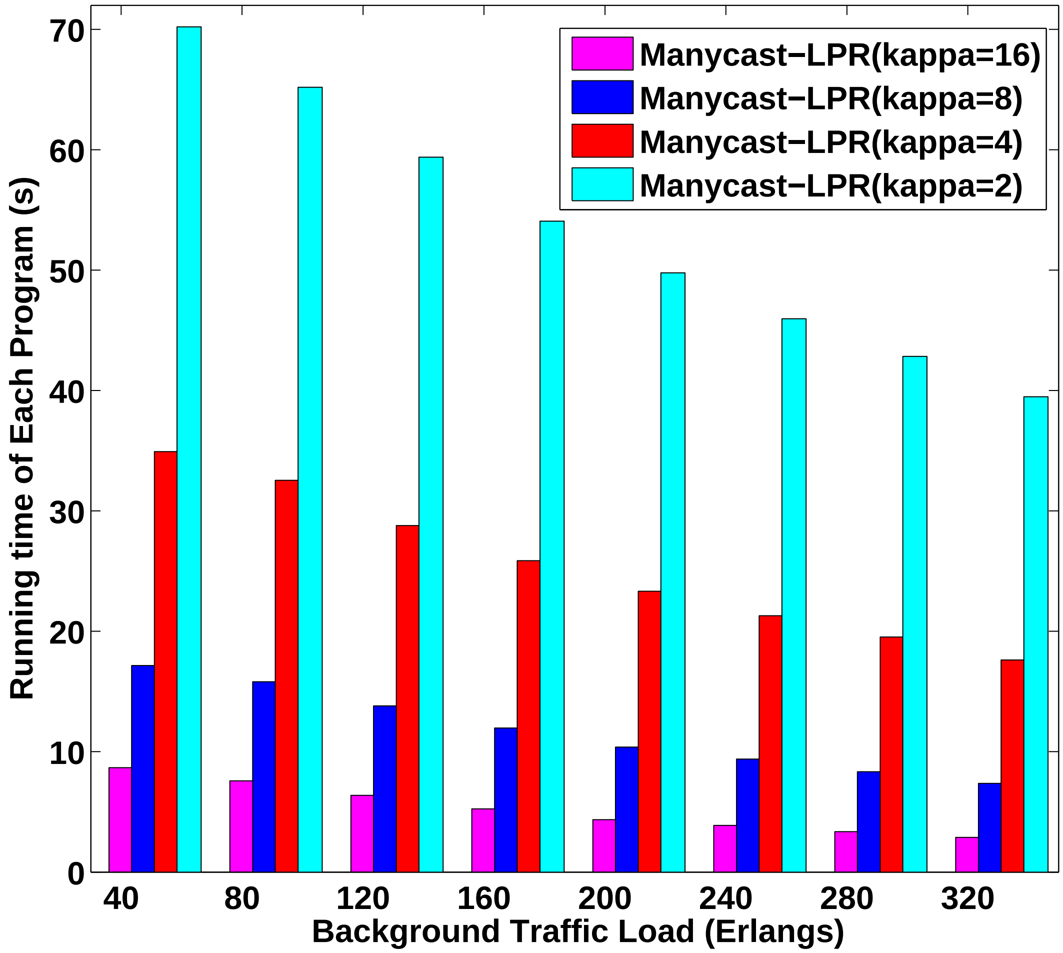

Fig. 8. Running time of Manycast-LPR.

3256-3261, Dec. 2012.

[16] D. Cavdar and F. Alagoz, "A survey of research on greening data centers", in Proc. *GLOBECOM*, pp. 3237-3242, Dec. 2012.

[17] W. Deng, *et al.*, "Harnessing renewable energy in cloud datacenters: opportunities and challenges", *IEEE Network*, vol. 28, no. 1, pp. 48-55, Jan. 2014

[18] M. Gattulli, M. Tornatore, R. Fiandra, and A. Pattavina, "Low-carbon routing algorithms for cloud computing services in IP-over-WDM networks", in Proc. *ICC*, pp. 2999-3003, Jun. 2012.

[19] A. Kiani and N. Ansari, "Toward low-cost workload distribution for integrated green data centers", *IEEE Communications Letters*, vol. 19, no. 1, pp. 26-29, Jan. 2015.

[20] K. Christodoulopoulos, I. Tomkos, and E. Varvarigos, "Elastic bandwidth allocation in flexible OFDM-based optical

networks”, *J. Lightw. Technol.*, vol. 29, pp. 1354-1366, May 2011.

[21] L. Zhang and Z. Zhu, ”Dynamic anycast in inter-datacenter networks over elastic optical infrastructure”, in Proc. *ICNC*, pp. 491-495, Feb. 2014.

[22] L. Zhang and Z. Zhu, ”Spectrum-efficient anycast in elastic optical inter-datacenter networks”, *Elsevier Optical Switching and Networking (OSN)*, vol. 14, no. 3, pp. 250-259, Aug. 2014.