20th IEEE Symposium on Computers and Communication (ISCC)

Minimizing Signaling Cost in Green Routing for Software Defined Networks

Hadi Alizadeh[1], Didem Gözüpek[2], Seyed M. Buhari[3], Aysegul Yayimli[1]
[1]Department of Computer Engineering, Istanbul Technical University, Istanbul, Turkey
[2]Department of Computer Engineering, Gebze Technical University, Kocaeli, Turkey
[3]Faculty of Computing and Information Technology, King Abdulaziz University, Saudi Arabia
Email: alizadehhadi@itu.edu.tr, didem.gozupek@gtu.edu.tr, mesbukary@kau.edu.sa, gencata@itu.edu.tr

***Abstract*—Research studies show that energy consumption in communication networks is mainly related to active network elements such as communication links. Based on this approach, several energy management techniques, generally known as green techniques, have been proposed. Their main goal is to minimize energy consumption by routing network traffic through a set of network resources and powering off the remaining unused resources. However, this approach imposes a signaling overhead on the routing system due to selectively powering off/on the network resources. In this work, we investigate the trade-off between energy efficiency and signaling overhead in a software defined network (SDN) domain with a single controller. To this end, we formulate an integer linear programming (ILP) problem whose objective function is to minimize the control overhead by taking into account the total energy consumption of the network. We then propose two polynomial-time heuristic algorithms to find near-optimal solutions for the problem. We evaluate the performance of our heuristic algorithms by comparison with the results obtained from our ILP formulation using optimization software CPLEX.**

Index Terms—Green networks, software defined networks, signaling overhead, routing update cost.

I. INTRODUCTION

Energy conservation is an important issue for countries due to its contribution to the economy as well as its environmental influences. Among economy sectors, the information and communication technologies (ICT) sector has paid much more attention to energy efficiency over the past years. According to [1], the annual electricity consumption of networking devices in the U.S. is 6.06 Terra Watt hours, which is equal to nearly 1 billion US dollars per year. Energy management has received attention in wired networks very recently although it has been a critical issue in wireless communications for a long time [2].

Researchers have hitherto proposed many techniques to bring energy-awareness into network elements and processes. These techniques are generally called green techniques [3]. Fisher et al. [4] categorized the existing techniques into network-wide and local (device-level) approaches.

Device-level approaches aim to improve the energy efficiency of individual network devices by advanced technological hardware. In these approaches, the networking device is put into sleep or a lower energy mode based on local decisions. [5] presented a power measurement study of a variety of networking devices such as hubs, edge switches, core switches, routers and wireless access points. The authors proposed link rate adaptation technique for low traffic periods and sleeping for no traffic periods. Mahadevan et al. [6] estimated the power consumption of a switch using a linear model. In this model, the energy saving is obtained from adapting the rate of a port, disabling a port as well as completely switching off a device.

Network-wide schemes aim to reduce the total network power consumption from routing perspective. In this type of routing called *green routing*, the objective is to aggregate demands over a subset of network nodes and links and allow other links and devices to be switched off [7].

Chabarek et al. [8] formulated a green routing problem as a multiple commodity network flow problem and used mixed-integer programming techniques to investigate power consumption and network performance in both static and dynamic network design. [3] formulated a minimum energy routing problem as an optimization problem, analyzed a green routing algorithm and solved it numerically considering a core-network scenario. It is shown in [8] that the energy consumption of network devices is largely independent of their load. Thus, the power consumption in networks is mainly due to active devices. In line with this fact, the authors in [9] aim to find a routing that minimizes the number of active links. For this purpose, they modeled the routing problem as an integer linear program (ILP) and then proposed a heuristic algorithm. [10] proposed an energy-aware routing model and compared it with an energy-agnostic routing model. The objective in [10] is to minimize the total power consumption of the network taking into consideration a user delay constraint.

Since switching from one path to another requires time and additional signaling overhead, an energy-aware mechanism should in general avoid rerouting too often. [11] proposed an ILP formulation for traffic engineering including constraints on the changes of the device states and routing paths to limit the impact on quality of service and the signaling overhead.

Making network-wide decisions for green routing is a problematic task in the current OSPF or IBGP networks as there is a need for a centralized decision making mechanism to determine which devices to switch off/on in green routing. In the current protocols, routers are responsible for establishing routes for demands. Thus, a green routing approach for the current networks has to collect link state data from the entire network, create a database and distribute it to all routers [1].

Recently, the idea of software defined networking (SDN) [12] has attracted considerable attention. SDN is a new networking paradigm in which the forwarding hardware is decoupled from control decisions. In this paradigm, the network intelligence is (logically) centralized in the controller and network devices are simple forwarding devices that can be programmed by the controller. SDN provides a centralized management as well as a fast reshaping of the traffic through modifications of the flow tables in the forwarding devices. The aforementioned properties make SDN a proper environment for developing green routing techniques. Using SDN technology, [13] explored a new solution to energy-aware flow scheduling in data centers.

An SDN controller periodically collects port, flow, and table level statistics from switches in its domain using control messages [14]. Implementing energy-aware algorithms in an SDN controller increases the number of these control messages as well. Considering that a controller usually has a limited resource capacity, the energy efficiency and control overhead are conflicting parameters in a green routing scheme.

The research studies about routing in green networks [3], [8], [9], [10], [11] and resource management in SDN [13], [14] mainly focus on minimizing total energy consumption and ignore the impact of control overhead such as routing update and sleep messages. To the best of our knowledge, this paper is the first one addressing the trade-off between energy efficiency of green routing and the control overhead it generates in an SDN domain with a single controller.

In this paper, we formulate a green routing problem as a multiple commodity network flow problem. In particular, we propose an ILP problem whose objective function is to minimize the control overhead subject to an energy constraint that limits the total energy consumption of the network. We then propose two polynomial-time heuristic algorithms to find near-optimal solutions for the problem. We evaluate the performance of our heuristic algorithms by comparison with the results obtained from our ILP formulation using optimization software CPLEX.

The remainder of this paper is organized as follows. In Section II, we present the ILP formulation for our green routing problem. In Section III, our proposed heuristic algorithms are described. A set of numerical results obtained from different network instances and with different values of the model parameters are shown and discussed in Section IV. Finally, Section V concludes the paper.

II. PROBLEM FORMULATION

A. Description

We represent the considered SDN network as a graph $G(V, E)$, where the set of nodes V denotes interconnection devices and the set of edges E represents communication links. In the considered SDN system, controller is responsible for performing green routing. To this end, the controller generates two types of control messages: *routing update* and *sleep* messages. Routing update messages, which contain new flow table entries, are sent to all nodes along calculated paths and sleep messages are sent to the nodes at both ends of the links for which a sleep decision has been made.

An important aspect in our modeling is the temporal nature of the green routing problem. Arrival of new demands into the system or the release of a number of existing connections necessitates rerouting by which the system transits into a new state from the viewpoint of energy consumption. We will refer to these states as *configuration* in our modeling. In other words, a configuration is a network state in a particular time period, during which the set of traffic demands which is given as an input in the form of source and destination pairs to our ILP, is static. This set has cardinality $|P|$. Our ILP aims to establish a path for each demand $p \in P$. Results of the previous configuration are fed as input to the current configuration.

We describe the input parameters and the decision variables in our ILP formulation as follows:

Parameters

g_{ij}	It represents the network topology. g_{ij} is equal to 1, if a link between node i and node j (link $(i, j) \in E$) exists in the topology.
l_{ij}	It is equal to 1, if link (i, j) is used by some path in the previous configuration.
l_{ij}^{p}	It is equal to 1 if link (i, j) is used by path p in the previous configuration.
c_i	Cost of a routing table update message sent to node i.
d_i	Cost of a sleep message sent to node i.
w_{ij}	Energy consumption of link (i, j).
W_T	The upper limit for total energy consumption of the network.
m_{ij}	It shows the capacity of link (i, j). m_{ij} is the number of paths that link (i, j) can accommodate.

Decision Variables

x_{ij}	A binary variable that is equal to 1 if any path uses link (i, j) in the direction from i to j in the current configuration.
x_{ij}^{p}	A binary variable that is equal to 1 if link (i, j) is used by path p in the direction from i to j in the current configuration.
r_i	An integer variable that denotes the number of routing update messages sent to node i.
s_i	An integer variable that denotes the number of sleep messages sent to node i.

B. Formulation

In our ILP formulation, the objective function is to minimize the signaling cost of green routing. The objective function is comprised of two terms: the first one is the total cost of routing update messages and the second one is the total cost of sleep messages generated by SDN controller.

Objective Function:

$$min \quad \sum_{i=1}^{|V|} c_i \cdot r_i + \sum_{i=1}^{|V|} d_i \cdot s_i \tag{1}$$

x_{ij} represents active (used) links in the new configuration. New configuration should belong to the given graph $G(V, E)$:

$$x_{ij} \leq g_{ij}; \qquad \forall i, j \in V \tag{2}$$

Considering the current configuration, a link (i, j) is active if some path p uses it:

$$x_{ij}^{p} \leq x_{ij}; \qquad \forall i, j \in V, \forall p \in P \tag{3}$$

The link (i, j) is inactive if no path uses it:

$$x_{ij} \leq \sum_{p=1}^{|P|} x_{ij}^{p}; \qquad \forall i, j \in V \tag{4}$$

A routing update message is sent to node i if the link (i, j) is used by path p in the current configuration (x_{ij}^{p}=1) but not used by the same path p in the previous configuration (l_{ij}^{p}=0). The following constraint models this behavior, where r_i denotes the number of routing update messages sent to node i:

$$r_i = \sum_{j=1}^{|V|} \sum_{p=1}^{|P|} x_{ij}^{p} \cdot (1 - l_{ij}^{p}); \qquad \forall i \in V \tag{5}$$

A sleep message is sent to node i if link (i, j) is active in the previous configuration ($l_{ij} = 1$), but not active in the current configuration ($x_{ij} = 0$). This situation implies that the link has to be put into sleep mode, which can be accomplished when nodes at both ends of the link put their respective ports into sleep. The following constraint models this behavior, where s_i denotes the number of sleep messages sent to node i:

$$s_i = \sum_{j=1}^{|V|} \Big(l_{ij} \cdot (1 - x_{ij}) + l_{ji} \cdot (1 - x_{ji}) \Big); \qquad \forall i \in V \tag{6}$$

Flow conservation constraint:

The following constraints guarantee that each demand p is allocated a path from its source to its destination.

$$\forall i \in V, \forall p \in P: \tag{7}$$

$$\sum_{j'=1}^{|V|} x_{ij'}^{p} - \sum_{j'=1}^{|V|} x_{j'i}^{p} = \begin{cases} 1; & \text{if } i \text{ is source of p} \\ -1; & \text{if } i \text{ is destination of p} \\ 0; & \text{otherwise} \end{cases}$$

Energy consumption constraint:

Total energy consumption depends on the number of active links. The total energy consumption of all active links should not exceed the given threshold W_T:

$$\sum_{i=1}^{|V|} \sum_{j=1}^{|V|} w_{ij} \cdot x_{ij} \leq W_T; \tag{8}$$

Link capacity constraint:

The following constraint ensures that the maximum number of paths sharing link (i, j) does not exceed m_{ij}:

$$\sum_{p=1}^{|P|} x_{ij}^{p} \leq m_{ij}; \qquad \forall i, j \in V \tag{9}$$

Decision Variables:

$$x_{ij}, x_{ij}^{p} \in \{0, 1\} \tag{10}$$

$$r_i, s_i \in Z^{+} \cup \{0\} \tag{11}$$

III. PROPOSED ALGORITHM

We have proved that our formulated problem in Equations (1) to (11) is NP-Hard by a reduction from the Steiner Forest problem [15]; however, we do not mention it here due to space constraints. In this section, we propose two heuristics for our proposed green routing problem. These algorithms are called *Fully Greedy Heuristic* (FGH) and *Stepwise Greedy Heuristic* (SGH). Table I shows the input parameters of our heuristic algorithms, whereas Algorithm 1 and Algorithm 2 present the pseudocode of these heuristics.

At each configuration, both heuristics first deal with the terminated demands (F_r) and then accommodate the arriving demands (F_a). At the beginning of the first configuration, ϵ is set to zero because it is assumed that there is no demand in the system and all network links are inactive. The value of ϵ increases by accommodating demands and decreases by releasing the terminated demands.

Recall that routing update messages are sent to all nodes along a calculated path. It is thus necessary to allocate the possible shortest path for each demand to minimize the number of routing update messages required to establish a connection for that demand. In both heuristics, Yen's K-shortest path algorithm [16] is used to find a set containing k shortest paths (in terms of the number of hops) for each demand. This set is sorted in increasing order according to the length of the paths.

TABLE I
INPUT PARAMETERS OF THE HEURISTICS

G	Network topology
F_a	Set of arriving demands
F_r	Set of terminated demands
C	Capacity of each link
W	Energy consumption of each link
W_T	Energy threshold of the network
W_{temp}	Varying energy threshold
ϵ	Current energy consumption of the network
k	A parameter for Yen's algorithm
δ	Step value for varying threshold

Algorithm 1: Fully Greedy Heuristic (FGH)

```
for each demand $f \in F_r$ do
    Release(f);
    Update(C,$\epsilon$);
end
for each demand $f \in F_a$ do
    $S \leftarrow$ Yen_K_ShortestPath(G,f,k);
    $p \leftarrow$ Select_Best_Path(S);
    if ($p \neq \emptyset$) then
        Accommodate(p);
        Update(C,$\epsilon$);
    else
        return $InfeasibleSolution$;
    end
end
return $TotalControlMessages$;
```

Algorithm 2: Stepwise Greedy Heuristic (SGH)

```
for each demand $f \in F_r$ do
    Release(f);
    Update(C,$\epsilon$);
    if ($\epsilon \leq W_{temp} - \delta$) then
        $W_{temp} \leftarrow W_{temp} - \delta$;
    end
end
for each demand $f \in F_a$ do
    flag $\leftarrow$ true;
    while (flag= true) do
        $S \leftarrow$ Yen_K_ShortestPath(G,f,k);
        $p \leftarrow$ Select_Best_Path(S);
        if ($p \neq \emptyset$) then
            Accommodate(p);
            Update(C,$\epsilon$);
            $flag \leftarrow false$;
        else
            if ($W_{temp} < W_T$) do
                $W_{temp} \leftarrow W_{temp} + \delta$ ;
            else
                return $InfeasibleSolution$;
            end
        end
    end
end
return $TotalControlMessages$;
```

The function *Select_Best_Path()* chooses in this set the first path which satisfies both energy threshold and network capacity constraints.

If all demands in a configuration are accommodated, the heuristics calculate routing update and sleep messages for that configuration. However, failure of the algorithms in finding a path for even one of the demands in a configuration yields an infeasible solution for that configuration and for the experiment, which is performed for a certain value of the energy threshold of the network (W_T).

The heuristics differ only in the way they accommodate the demands. FGH greedily activates new links and assigns shortest paths for the demands as long as ϵ is below W_T. If $\epsilon = W_T$, since activating new links violates the energy threshold, FGH checks if the previously activated links can be used to find a path for the following demands.

SGH is also a greedy algorithm; however, it is more flexible than FGH. In addition to W_T, SGH employs a varying threshold W_{temp}, which dynamically changes according to the arrival of new demands and release of the existing demands. W_{temp} varies in a range between an initialized value and W_T by using an input parameter δ. W_{temp} is initially set to the value of δ at the beginning of the first configuration. SGH then increases or decreases the value of W_{temp} by δ depending on the situation.

Taking into consideration the value of W_{temp} as the upper limit for energy consumption at each time, SGH starts assigning shortest paths to the demands in a greedy manner. By accommodating the demands, the value of ϵ reaches the value of W_{temp}. In this situation, SGH checks if it can find a path for the demands through the previously activated links. The paths found in this situation are not necessarily the possible shortest paths for the demands. However, if accommodation of new demands with the current value of W_{temp} is not possible and if increasing the value of W_{temp} by δ does not cause it to exceed the value of W_T, SGH increases the value of W_{temp} by δ and checks whether the demands can be accommodated in the new situation. This process is repeated until W_{temp} reaches W_T. At this point, if finding a path for a demand is not possible, SGH returns infeasible solution.

We have also proved that the computational complexity of our proposed heuristic algorithms is polynomial; however, we have not included them here due to space constraints.

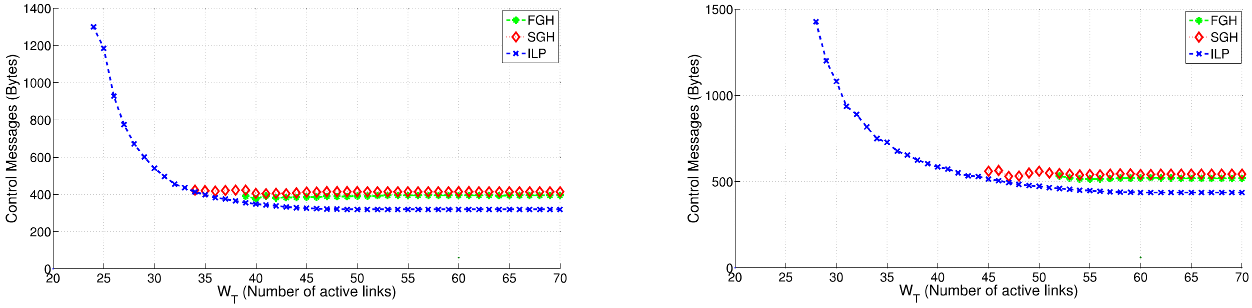
IV. NUMERICAL EVALUATION

In this section we present the numerical results of our ILP formulation and our proposed heuristics on different network instances. In our experiments, we investigated the impact of energy threshold of the network and network density on the amount of control overhead generated by our green routing mechanisms. We evaluate the performance of our heuristics by comparing them with the results obtained from our ILP formulation.

A. Simulation Setup

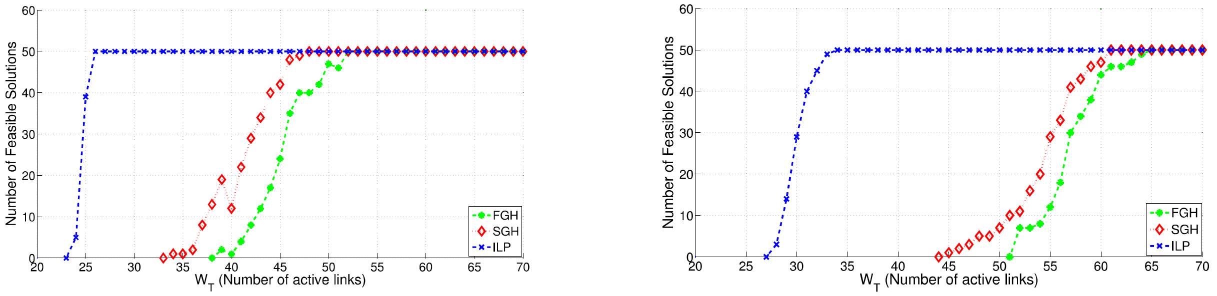
We used randomly generated topologies in our experiments. For the first test, we considered a network with 30 nodes where each node is of degree 4. We generated 50 different instances of this network. In the second experiment, we considered a network with 30 nodes and varied the node degrees from 3 to 7. For each value of node degree, we generated 50 different instances.

For our simulations, a network model that has high variation in its traffic demands is required while the available realistic traffic models are not applicable in our work as they are mainly intended for backbone networks, which have demands with low variations. For this purpose, we generated two traffic



(a) Control message with respect to W_T for traffic load with 30 demands. (b) Control message with respect to W_T for traffic load with 60 demands.

Fig. 1. Impact of W_T on the amount of control messages.



(a) Number of feasible solutions with respect to W_T for traffic load with 30 demands. (b) Number of feasible solutions with respect to W_T for traffic load with 60 demands.

Fig. 2. Impact of W_T on the number of feasible solutions.

loads with 30 and 60 demands. The sources and destinations of each demand are randomly chosen. The arrival rate and the duration of the demands follow Poisson distribution with average interarrival time of 500 ms and exponential distribution with average duration of 10 s respectively. With these parameters, we evaluated the total amount of control messages in 20 configurations which corresponds to a total simulation time of 20s. The number of demands on each configuration depends on the Poisson and exponential distributions.

As in [9], we assume that all links have the same value of energy consumption; therefore, W_T corresponds to the maximum number of links that can be active in routing the demands. Hence, the unit of W_T in Figs. 1 and 2 is the number of active links. We also assume that the links have the same capacity and each link can accommodate at most 25 demands.

The cost of control overhead is stated in bytes. In our green routing mechanism, sleep messages contain the identity of the interface to be switched off on a specific node. We thus consider a sleep message to have a size of one byte. We also consider three bytes for a routing update message, which contains a routing table entry consisting of the source and destination of the demand as well as the next hop. In all experiments the value of the parameter of Yen's K-shortest path algorithm (k) is set to 15 and the step value for varying threshold in SGH (δ) is set to be $0.1 \times W_T$.

B. *Impact of Network Energy Threshold*

We first investigate the impact of W_T on the amount of control messages considering two traffic loads with 30 and 60 demands. Fig. 1 shows the results of this experiment. The network instances used for this test have 120 links. We have started the experiment by setting W_T to 120, which means that the algorithm is allowed to use all the links in routing. We have then repeated the test for the smaller values of W_T.

We observe that ILP and both heuristics have a similar behavior when W_T decreases from 120 to 45 in Fig. 1(a). The same behavior is also observed when the value of W_T decreases from 120 to 55 in Fig. 1(b). In these ranges of W_T, the solutions found by our heuristics are close to those of ILP. The heuristics generate nearly 30 and 50 bytes more control messages than ILP in both traffic loads.

In the case of ILP, the number of control messages increases dramatically as W_T decreases below 45 in Fig. 1(a) and below 55 in Fig. 1(b). Our ILP formulation provides a lower bound for the amount of control messages. The heuristics would generate higher amounts of control messages than this bound even if they can reach a feasible solution. This trend shows that there is a trade-off between energy efficiency and control overhead for values of W_T smaller than a certain value. In other words, in order to obtain more energy efficiency, more control messages have to be generated.

We also observe that the value of W_T at which the trade-off starts, depends on the traffic load. As the number of demands increases from 30 to 60, this value also increases from 45 to 55. As to the performance of our heuristics, FGH outperforms SGH with a small difference of about 9 bytes in Fig. 1(a) and 15 bytes in Fig. 1(b) for the ranges of W_T where both heuristics reach a feasible solution.

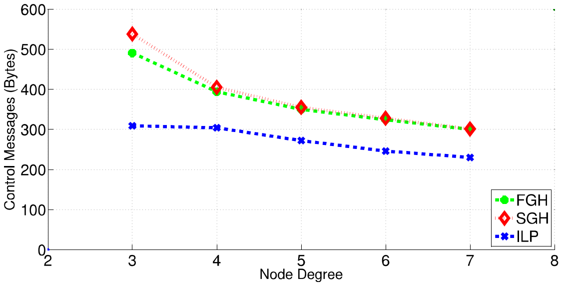


Fig. 3. Amount of control messages with respect to different node degrees for $W_T = 60$

Fig. 2 shows the performance of the algorithms in finding feasible solutions with respect to the values of W_T for two traffic loads with 30 and 60 demands. According to Fig. 2(a). FGH can find feasible solutions in all 50 instances for values of W_T greater than 59. However, SGH outperforms FGH and can find feasible solutions for W_T ranging from 54 to 120. From link sparing point of view, the results denote that FGH and SGH can guarantee reducing link usage in routing up to 50% and 55%, respectively, in the case of the traffic load with 30 demands. However, ILP is able to find feasible routing for the values of W_T equal to or greater than 26; hence, it guarantees 78% reduction in link usage. In the case of the traffic load with 60 demands, Fig. 2(b) shows that the guaranteed link sparing drops to 45%, 49%, and 75% for FGH, SGH, and ILP, respectively.

C. Impact of Network Topology Density

In the second set of simulations, we investigate the impact of network density on the amount of control overhead with respect to a fixed value of W_T. The considered network instances for this experiment have 30 nodes and their node degrees range from 3 to 7. We have used the traffic pattern with 30 demands in this experiment. We set the value of W_T to be 60, at which point all three methods yield feasible solutions for all values of node degrees in all network instances. Fig. 3 shows the amount of control messages with respect to different node degree values. The value of each point in the figure is the average of 50 different network instances.

The results show that there is a considerable difference between the performance of our heuristics and that of our ILP when node degree is 3. For this value of node degree, both FGH and SGH generate nearly 200 bytes more than the optimal amount achieved by ILP. The behavior, however, improves as the node degree increases. As the density of the instances increases, the performance of the heuristics approach the one of ILP. The difference in the amount of overhead generated by the heuristics and the ILP method reaches only 50 bytes when node degree is 7. Although FGH outperforms SGH when node degree is 3, both heuristics show a similar performance as node degree increases.

V. CONCLUSION

In this work we have investigated the tradeoff between signaling overhead and energy efficiency in green networks. For this purpose, we have formulated an ILP problem, which minimizes the control overhead by taking into account the total energy consumption of the network. We have also proposed two heuristic algorithms. Our simulation results show that our proposed methods achieve substantial energy savings while minimizing the signaling overhead.

As a future work, we plan to perform experiments on realistic network topologies and develop better heuristic algorithms which produce solutions much closer to those of our ILP. We also plan to design approximation algorithms, which have theoretically provable performance guarantee, to address the signaling overhead in green routing.

ACKNOWLEDGMENTS

This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under grant no.113E567.

REFERENCES

[1] M. Gupta and S. Singh, “Greening of the internet,” *ACM SIGCOMM*, pp. 19–26, 2003.

[2] X. Wang, A. Vasilakos, M. Chen, Y. Liu, and T. Kwon, “A survey of green mobile networks: opportunities and challenges,” *Mobile Networks and Applications*, vol. 17, no. 1, pp. 4–20, February 2012.

[3] A. Bianzino, C. Chaudet, F. Larroca, D. Rossi, and J. Rougier, “Energy-aware routing: A reality check,” in *GLOBECOM Workshops (GC Wkshps), 2010 IEEE*, Dec 2010, pp. 1422–1427.

[4] W. Fisher, M. Suchara, and J. Rexford, “Greening backbone networks: reducing energy consumption by shutting off cables in bundled links,” in *ACMSIGCOMM Workshop on Green networking*, 2010, pp. 29–34.

[5] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, “A power benchmarking framework for network devices,” in *IFIP Networking*, May 2009, pp. 795–808.

[6] ——, “Energy aware network operations,” in *INFOCOM Workshops, IEEE*, April 2009.

[7] A. Bianzino, C. Chaudet, D. Rossi, and J. Rougier, “A survey of green networking research,” *Communications Surveys Tutorials, IEEE*, vol. 14, no. 1, pp. 3–20, First 2012.

[8] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, “Power awareness in network design and routing,” in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, April 2008, pp. 457–465.

[9] F. Giroire, D. Mazauric, J. Moulierac, and B. Onfroy, “Minimizing routing energy consumption: From theoretical to practical results,” in *Green Computing and Communications (GreenCom), 2010 IEEE/ACM Int'l Conference on Int'l Conference on Cyber, Physical and Social Computing (CPSCom)*, Dec 2010, pp. 252–259.

[10] L. Chiaraviglio and I. Matta, “Greencoop: Cooperative green routing with energy-efficient servers,” 2010, pp. 191–194.

[11] B. Addis, A. Capone, G. Carello, L. Gianoli, and B. Sanso, “Energy management through optimized routing and device powering for greener communication networks,” *IEEE/ACM Transactions on Networking*, vol. 22, no. 1, pp. 313–325, Feb 2014.

[12] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, “A survey of software-defined networking: Past, present, and future of programmable networks,” *Communications Surveys Tutorials, IEEE*, vol. 16, no. 3, pp. 1617–1634, Third 2014.

[13] D. Li, Y. Shang, and C. Chen, “Software defined green data center network with exclusive routing,” in *INFOCOM, 2014 Proceedings IEEE*, April 2014, pp. 1743–1751.

[14] M. Bari, A. Roy, S. Chowdhury, Q. Zhang, M. Zhani, R. Ahmed, and R. Boutaba, “Dynamic controller provisioning in software defined networks,” vol. 18, pp. 18–25, October 2013.

[15] C.-L. Li, S. T. McCormick, and D. Simchi-Levi, “The point-to-point delivery and connection problems: complexity and algorithms.” *Discrete Applied Mathematics*, vol. 36, no. 3, pp. 267–292, 1992.

[16] J. Yen, “Finding the k shortest loopless paths in a network,” *management Science*, pp. 712–716, 1971.